بِسۡمِ اللّٰہِ الرَّحۡمٰنِ الرَّحِیۡمِ

اللہ کے نام سے شروع جو بڑا مہربان نہایت رحم فرمانے والا ہے۔

کمپیوٹر سائنس
اینڈ
انٹرپرینیورشپ

10

نیشنل کریکولم آف پاکستان 2023ء (نظرِ ثانی شدہ) کے مطابق

PECTAA
Transformation, Innovation and Excellence

پنجاب ایجوکیشن، کریکولم، ٹریننگ اینڈ اسسمنٹ اتھارٹی

مصنّفین

- **پروفیسر ڈاکٹر محمد عاطف چٹھہ**
  ڈین فیکلٹی آف کمپیوٹر سائنس
  لاہور گیریژن یونیورسٹی لاہور
- **پروفیسر ڈاکٹر ارشد علی**
  فاسٹ سکول آف کمپیوٹنگ، یونیورسٹی آف کمپیوٹر
  اینڈ امرجنگ سائنسز (NUCES) لاہور کیمپس
- **ڈاکٹر عبدالستار**
  ڈیپارٹمنٹ آف کمپیوٹر سائنس
  لاہور گیریژن یونیورسٹی لاہور
- **پروفیسر ڈاکٹر اشفاق احمد**
  لاہور گیریژن یونیورسٹی، لاہور
- **پروفیسر ثاقب محمود**
  لاہور گیریژن یونیورسٹی، لاہور
- **ذوالفقار علی کلاچی**
  ہیڈ ماسٹر، گورنمنٹ ہائی سکول، کوٹ ہیبت، ڈی جی خان

ایڈیٹرز

- **پروفیسر ڈاکٹر مدثر نصیر**
  ڈیپارٹمنٹ آف کمپیوٹر سائنس اینڈ آئی ٹی، یونیورسٹی آف لاہور، لاہور
- **پروفیسر ثاقب عبید**
  انسٹیٹیوٹ آف کمپیوٹر سائنس، خواجہ فرید یونیورسٹی آف انجنیئرنگ اینڈ آئی ٹی، رحیم یار خان

مترجم

- **کاشف شہزاد چودھری** گورنمنٹ مسلم ہائی سکول نمبر 2، سول لائنز، لاہور

ریویو کمیٹی

- **پروفیسر محمد فہیم**
  گورنمنٹ گریجویٹ کالج، گلبرگ لاہور
- **پروفیسر مہوش پونم**
  گورنمنٹ ایسوسی ایٹ کالج برائے خواتین، گوالمنڈی لاہور
- **انجم حسین**
  اسسٹنٹ ایجوکیشن آفیسر، مرکز خان پور، تحصیل کوٹ چھٹہ
- **پروفیسر محمود احمد چودھری**
  دی کریسنٹ کالج، شادمان، لاہور

ریویو کمیٹی برائے اُردو میڈیم

- **اعجاز حسین** SST(CS/IT)
  گورنمنٹ بوائز ہائی سکول، گوالمنڈی لاہور
- **صائمہ** SST(CS/IT)
  گورنمنٹ ماڈل گرلز ہائی سکول، سمن آباد لاہور
- **محمد نوید الرحمٰن** SST(CS/IT)
  گورنمنٹ ہائی سکول چوہنگ، ملتان روڈ، لاہور
- **سید مدثر حسین** SST(CS/IT)
  گورنمنٹ اسلامیہ ہائی سکول، سنت نگر، لاہور

نگرانِ طباعت

- **جہاں زیب خان**، اسسٹنٹ ڈائریکٹر، پنجاب ایجوکیشن، کریکولم، ٹریننگ اینڈ اسسمنٹ اتھارٹی، لاہور

ڈائریکٹر (C&C)	عامر ریاض
ڈپٹی ڈائریکٹر (C-S)	سیّد صغیر الحسنین ترمذی
انچارج آرٹ سیل	عائشہ صادق
لے آؤٹ اینڈ ڈیزائننگ	طاہر سرفراز بٹ
السٹریٹرز	آیت اللہ، طاہر سرفراز بٹ

تجرباتی ایڈیشن

ناشر: پنجاب ایجوکیشن کریکولم ٹریننگ اینڈ اسسمنٹ اتھارٹی، لاہور۔
مطبع: سلور کلاسک پرنٹنگ اینڈ پیکیجنگ، لاہور

DATE OF PRINTING	PMIU DANISH SCHOOL	PEF	MLWC	PWWF	TOTAL QUANTITY
Feb, 2026	187,921	15,285	127	94	203,427

یہ کتاب پاکستان کے نظرثانی شدہ قومی نصاب 2023ء پر مبنی ہے۔ جملہ حقوق بحق پنجاب ایجوکیشن، کریکولم، ٹریننگ اینڈ اسسمنٹ اتھارٹی محفوظ ہیں۔ اس کتاب کا کوئی حصہ نقل یا ترجمہ نہیں کیا جاسکتا نہ ہی ٹیسٹ پیپر، گائیڈ بکس، خلاصہ جات، نوٹس یا امدادی کتب کی تیاری میں استعمال کیا جاسکتا ہے۔

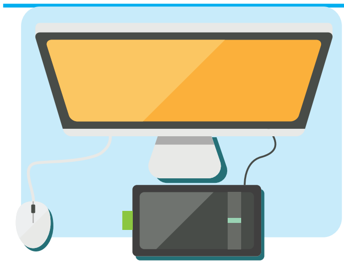
فہرست

یونٹ	عنوانات	صفحہ نمبر
1	آپریٹنگ سسٹمز ساخت اور خدمات	1
2	سسٹم ریکوری اینڈ مینٹیننس	19
3	تعارف برائے پائتھن پروگرامنگ	40
4	پائتھن میں کنٹرول اسٹرکچر+	57
5	ڈیٹا سائنس کا تعارف	73
6	مصنوعی ذہانت (AI) اور مشین لرننگ (ML) کا تعارف	95
7	مصنوعی ذہانت (AI) کے اطلاقات	109
8	ڈیجیٹل انٹرپرینیور شپ	125
	سوالیہ پرچے کی تیاری بارے ہدایات	
	ماڈل سوالیہ پرچہ	

یونٹ

1

آپریٹنگ سسٹمز
ساخت اور خدمات



حاصلاتِ تعلم: اس باب کے اختتام تک، آپ اس قابل ہو جائیں گے کہ:

- آپریٹنگ سسٹم کی تعریف کر سکیں اور اس کے کردار کی وضاحت کر سکیں کہ یہ صارف اور ہارڈویئر کے درمیان بطور رابطہ (Interface) کیسے کام کرتا ہے۔
- متعدد صارف اکاؤنٹس کو سنبھالنے میں آپریٹنگ سسٹم کی ذمہ داریوں کی شناخت کر سکیں۔
- کرنل (Kernel) اور شیل (Shell) کے درمیان فرق بتا سکیں، بشمول ان کے مخصوص افعال۔
- آپریٹنگ سسٹم کی تہوں (OS Layers) اور ہر تہہ ایک دوسرے کے ساتھ کیسے باہمی رابطہ کرتی ہے، کی وضاحت کر سکیں۔
- آپریٹنگ سسٹم کے کردار میں سسٹم لائبریریوں اور ڈیوائس ڈرائیورز کے کردار کی شناخت کر سکیں۔
- پروسیس لائف سائیکل کے مراحل (تخلیق، عملدرآمد اختتام) کی مثالوں کے ساتھ وضاحت کر سکیں اور فرسٹ-کم، فرسٹ-سروڈ (FCFS) پروسیس شیڈولنگ کو لاگو کر سکیں۔
- ملٹی ٹاسکنگ اور ہم آہنگی (Concurrency) کی وضاحت کر سکیں، اور ایک حقیقی دنیا کی مثال دے سکیں۔
- پرائمری میموری (RAM) اور ورچوئل میموری کے درمیان فرق کر سکیں۔
- پروسیس اور تھریڈ کی تعریف کر سکیں، اور وضاحت کریں کہ تھریڈز ایک پروسیس کے اندر وسائل کو کیسے شیئر کرتے ہیں۔
- مثالیں دیں کہ کس طرح ملٹی تھریڈنگ (Multithreading) ایپلیکیشنز میں کارکردگی کو بہتر بناتی ہے۔
- سسٹم کال (System Call) کی تعریف کر سکیں اور اس کا مقصد اور اقسام کی وضاحت کر سکیں۔
- فائل سسٹم، فائلیں، فولڈرز، اور میٹاڈیٹا کی تعریف کر سکیں، اور ڈیٹا کو منظم کرنے میں ان کے کردار کی وضاحت کر سکیں۔
- یہ اجزاء مل کر کس طرح صارف کی درخواست کو ڈیوائس سے سرور تک اور پھر واپس پراسیس کرتے ہیں، کی وضاحت کر سکیں۔

تعارف

اس باب کے اختتام تک، آپ سمجھ جائیں گے کہ ایک آپریٹنگ سسٹم کس طرح درپردہ کام کرتا ہے تاکہ وسائل کا انتظام کرے، کارکردگی کو برقرار رکھے، ڈیٹا کو محفوظ رکھے، اور موثر اور کارآمد سسٹم کی کارکردگی کو یقینی بنانے کے لیے ضروری پس منظر کے افعال انجام دے۔ اس باب میں ہم دیکھیں گے:

- آپریٹنگ سسٹم کا آرکیٹیکچر اور کمپیوٹر کے مرکزی کنٹرولر کے طور پر اس کا کردار۔ پراسس مینجمنٹ کا تصور جیسا کہ پراسس لائف سائیکل، ملٹی ٹاسکنگ اور شیڈولنگ۔
- میموری مینجمنٹ، ملٹی تھریڈنگ، اور ہارڈویئر وسائل تک محفوظ طریقے سے رسائی کے لیے سسٹم کالز کا استعمال۔
- فائل سسٹم کی تنظیم اور آپریٹنگ سسٹم کی اہم اقسام، بشمول ریئل ٹائم، ایمبیڈڈ، نیٹ ورک، اور موبائل آپریٹنگ سسٹم۔

1.1 آپریٹنگ سسٹم (OS) کا تعارف

آپریٹنگ سسٹم (OS) ایک قسم کا سسٹم سافٹ ویئر ہے جو کمپیوٹر کے ہارڈویئر کو منظم کرتا ہے، ایپلیکیشن سافٹ ویئر کو چلاتا ہے، اور صارف (User) کو انٹرفیس فراہم کرتا ہے۔ اس کی مثالیں Windows, macOS, Linux, Android اور iOS ہیں، جن کے اپنے بنیادی کام ہوتے ہیں۔ یہ باب اس بات پر توجہ دیتا ہے کہ آپریٹنگ سسٹم کس طرح کمپیوٹر سسٹم کے مرکزی کنٹرولر کے طور پر کام کرتا ہے اور صارف اور ہارڈویئر کے درمیان ہموار اور محفوظ رابطہ قائم رکھتا ہے، خاص طور پر جب ایک سے زیادہ صارفین (multi-user) سسٹم استعمال کر رہے ہوں۔

آپریٹنگ سسٹم بطور مرکزی سسٹم کنٹرولر (OS as the Central System Controller)

ایک آپریٹنگ سسٹم کمپیوٹر کے لیے ٹریفک کنٹرولر کی طرح کام کرتا ہے۔ یہ فیصلہ کرتا ہے کہ کون سا کام پہلے کیا جانا چاہیے، کمپیوٹر کی میموری کس طرح استعمال ہوتی ہے، اور ایک مخصوص وقت میں کون سی ڈیوائسز (جیسے پرنٹر یا سپیکر) فعال ہونی چاہئیں۔ مختصراً، ہم کہہ سکتے ہیں کہ OS اس بات کو یقینی بناتا ہے کہ کمپیوٹر آسانی سے کام کرتا رہے، یہاں تک کہ جب ایک ہی وقت میں متعدد کام چل رہے ہوں۔

صارف اور ہارڈویئر کے درمیان باہمی رابطہ کا کردار (Role in User-Hardware Interaction)

کمپیوٹر ہارڈویئر براہ راست انسانی زبان کو نہیں سمجھ سکتا۔ آپریٹنگ سسٹم صارف اور ہارڈویئر سسٹم کے درمیان ایک مترجم کے طور پر کام کرتا ہے۔

مثال: جب آپ کسی آئیکن پر کلک کرتے ہیں یا کچھ ٹائپ کرتے ہیں، تو OS ان افعال کو ان ہدایات میں بدل دیتا ہے جنھیں ہارڈویئر سمجھ سکتا ہے۔ یہ لوگوں کو ہارڈویئر کے کام کرنے کا طریقہ سیکھے بغیر کمپیوٹر کو آسانی سے استعمال کرنے کی اجازت دیتا ہے۔

کیا آپ جانتے ہیں؟

انسان اردو یا انگریزی جیسی زبانیں بولتے ہیں، جبکہ کمپیوٹر صرف بائنری زبان یا آن اور آف سوئچ یعنی 10 اور 1 کو سمجھتے ہیں۔

مثال: جب آپ اپنے کی بورڈ پر حرف 'A' دباتے ہیں، تو آپریٹنگ سسٹم اسے ایک خاص بائنری کوڈ میں تبدیل کرتا ہے:

بائنری میں حرف 'A' 01000001_2 ہے۔

یہ کوڈ پھر ہارڈویئر کو بھیجا جاتا ہے، جو اسکرین پر 'A' کو ظاہر کرتا ہے۔ یہ عمل اتنی تیزی سے ہوتا ہے کہ صارف کو فوری دکھائی دیتا ہے۔

ملٹی یوزر ماحول میں ذمہ داریاں

اسکولوں، دفاتر، یا آن لائن سسٹمز جیسی جگہوں پر، بہت سے لوگ ایک ہی کمپیوٹر سسٹم استعمال کر سکتے ہیں۔ آپریٹنگ سسٹم اس بات کو یقینی بناتا ہے کہ:

- الگ الگ یوزر اکاؤنٹس بناتا ہے اور ان کا انتظام کرتا ہے۔
- ہر صارف کی فائلوں اور معلومات (Files and Information) کو محفوظ اور نجی (Private) رکھتا ہے۔

- کمپیوٹر کے وسائل کو منصفانہ طور پر تقسیم کرتا ہے تاکہ کوئی ایک صارف سسٹم کی کارکردگی (System Functionality) کو متاثر نہ کرے۔
- مشترکہ وسائل کو غیر مجاز رسائی سے محفوظ رکھا جائے۔

یوزر اکاؤنٹس بنانا اور انہیں منظم کرنا (Creating and Managing User Accounts)

آپریٹنگ سسٹم مختلف صارفین کے لیے الگ اکاؤنٹس کی اجازت دیتا ہے، ہر ایک یوزر کے اپنے ڈیسک ٹاپ، فائلوں، سیٹنگز، اور پاس ورڈز کے ساتھ۔

مثال: کمپیوٹر لیب میں، طلباء اسکول کی طرف سے فراہم کردہ یوزر نیم اور پاس ورڈ کے ساتھ لاگ ان ہوتے ہیں۔ یہ ہر طالب علم کے کام کو الگ اور نجی رکھتا ہے۔ ونڈوز جیسے زیادہ ترجدید OS میں، Settings یا Control Panel کے ذریعے ایک نیا اکاؤنٹ بنایا جاسکتا ہے، جہاں User Accounts کو منتخب کرکے، Add New User کا انتخاب کیا جاتا ہے، اور یوزر نیم، پاس ورڈ، اور اکاؤنٹ کی قسم (Standard, Administrator, Guest) جیسی تفصیلات درج کی جاتی ہیں۔

1.2 آپریٹنگ سسٹم کا آرکیٹیکچر

ایک آپریٹنگ سسٹم کا آرکیٹیکچر وہ طریقہ ہے جس سے اس کے حصے منظم ہوتے ہیں اور وہ ایک ساتھ کام کرتے ہیں۔ ہر حصے کا ایک خاص کردار ہوتا ہے، اور وہ تمام مل کر کمپیوٹر کو آسانی سے کام کرنے کے قابل بناتے ہیں۔ یہ بالکل ایک اسکول کی طرح ہے جس میں مختلف شعبے مخصوص فرائض انجام دیتے ہیں لیکن ادارے کے ہموار کام کے لیے مل کر کام کرتے ہیں۔

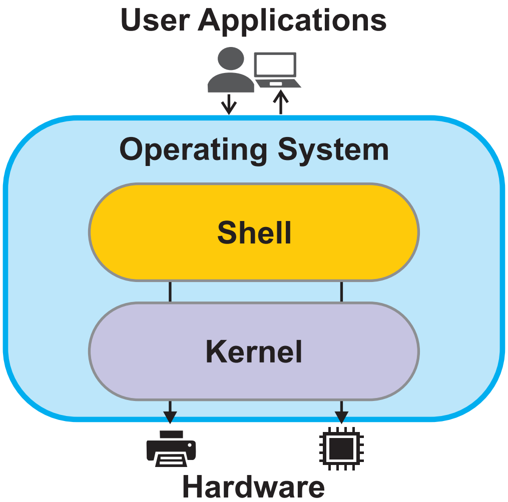
کرنل بمقابلہ شیل (Kernel vs. Shell)

کرنل (Kernel): کرنل آپریٹنگ سسٹم کا بنیادی حصہ ہے جو براہ راست کمپیوٹر کے ہارڈ ویئر اور سسٹم کے وسائل، جیسے CPU، میموری، اور ڈیوائسز کو کنٹرول کرتا ہے جیسا کہ شکل 1.1 میں دکھایا گیا ہے۔ یہ فیصلہ کرتا ہے کہ مختلف پروگرام ان وسائل کو کب اور کیسے استعمال کرسکتے ہیں۔

مثال: جب آپ کوئی فائل کھولتے ہیں، تو کرنل اسے ہارڈ ڈرائیو سے پڑھنے اور اسکرین پر بھیجنے کے عمل کا انتظام کرتا ہے۔
کرنل کا کام کار کے انجن جیسا ہے، جبکہ شیل، ڈیش بورڈ جیسا ہے اور اسیسریز اسٹیئرنگ وہیل جیسی ہیں۔

شیل (Shell): شیل OS کا بیرونی حصہ ہے جو صارف کے ساتھ تعامل کرتا ہے۔ یہ صارف سے کمانڈز وصول کرتا ہے اور انہیں کرنل تک پہنچاتا ہے۔ شیلز کی دو اقسام ہیں:

- گرافیکل شیلز (Graphical shells): (جیسے ونڈوز ڈیسک ٹاپ) جہاں آپ آئیکنز پر کلک کرتے ہیں اور مینیوز کا استعمال کرتے ہیں۔
- کمانڈ-لائن شیلز (Command-line shells): (جیسے کمانڈ پرامپٹ یا ٹرمینل) جہاں آپ ہدایات ٹائپ کرتے ہیں۔



شکل 1.1: آپریٹنگ سسٹم میں صارف، شیل، اور کرنل کے درمیان تعلق کو واضح کرنے والی ڈایاگرام۔

OSI کی لیئرز اور ماڈیولر ڈیزائن

آپریٹنگ سسٹمز میں، ڈیزائن کو لیئرز میں تقسیم کیا جاتا ہے، جہاں ہر لیئر کا ایک مخصوص کام ہوتا ہے:

- **لور لیئرز (Lower layers):** براہ راست ہارڈ ویئر ڈیوائسز، جیسے CPU، میموری، اور اسٹوریج اور ہارڈ ڈرائیو کے ساتھ کام کرتی ہیں۔
- **مڈل لیئرز (Middle layers):** وسائل کا انتظام کرتی ہیں اور یقینی بناتی ہیں کہ ضرورت پڑنے پر پروگرام انہیں استعمال کر سکیں۔
- **اَپر لیئرز (Upper layers):** ایپلی کیشنز کو چلاتی ہیں اور وہ انٹرفیس فراہم کرتی ہیں جسے صارف اسکرین پر دیکھتا ہے۔
- ہر لیئر اپنے نیچے والی لیئر پر منحصر ہوتی ہے۔ یہ ڈیزائن آپریٹنگ سسٹم کے پورے سسٹم کو تبدیل کیے بغیر انتظام، مرمت، اور بہتر بنانے میں آسانی فراہم کرتا ہے۔

مثال: آپریٹنگ سسٹم کے لیئرڈ آرکیٹیکچر کو سمجھنے کے لیے ایک اسکول سسٹم کی مثال استعمال کی جاسکتی ہے:

- سپورٹ اسٹاف (Support Staff): جیسے چوکیدار اور صفائی کرنے والے نچلی سطح پر کام کرتے ہیں، تاکہ اسکول تیار رہے (جیسے لو لیئر ہارڈ ویئر کے ساتھ کام کر رہی ہیں)۔
- انتظامیہ (Administration): وسائل، شیڈول، اور قوانین کا انتظام کرتی ہے (جیسے مڈل لیئر میموری اور اسٹوریج کا انتظام کر رہی ہیں)۔

اساتذہ اور طلباء (Teachers and Students): ان انتظامات کو پڑھانے اور سیکھنے کے لیے استعمال کرتے ہیں (جیسے اَپر لیئرز ایپلی کیشنز کو

- چلاتی ہیں اور صارفین کے ساتھ تعامل کرتی ہیں)۔

سسٹم لائبریریز اور ڈیوائس ڈرائیورز (System Libraries and Device Drivers)

سسٹم کی لائبریریاں تیار شدہ ہدایات کا مجموعہ ہیں جنہیں پروگرام عام کاموں کو انجام دینے کے لیے استعمال کر سکتے ہیں، جیسے فائلیں کھولنا یا اسکرین پر ٹیکسٹ دکھانا۔

مثال: جب کسی فوٹو ایڈیٹنگ ایپ کو کوئی تصویر کھولنے کی ضرورت ہوتی ہے، تو وہ فائل کو پڑھنے کے لیے آپریٹنگ سسٹم کی لائبریری کا استعمال کرتی ہے۔ اس طرح، ایپ کو تصاویر کھولنے کے لیے اپنا طریقہ اپنانے کی ضرورت نہیں پڑتی۔

ڈیوائس ڈرائیورز (Device Drivers) خاص پروگرام ہیں جو آپریٹنگ سسٹم کو ہارڈویئر ڈیوائسز، جیسے پرنٹرز، کی بورڈز، اور گرافکس کارڈز کے ساتھ بات چیت کرنے کی اجازت دیتے ہیں۔

سرگرمی:

مقصد: اپنے کمپیوٹر کے ذریعے استعمال ہونے والے شیل، ڈیوائس ڈرائیور، اور سسٹم لائبریری کی شناخت کریں۔

طریقہ کار:

1. اپنے کمپیوٹر پر شیل کا پتہ لگائیں (ڈیسک ٹاپ یا کمانڈ-لائن)۔
2. Device Manager (ونڈوز) یا System Information (میک / لینکس) کھولیں۔
3. ایک ہارڈویئر ڈیوائس (مثلاً، پرنٹر، کی بورڈ) تلاش کریں اور اس کے ڈرائیور کا نام نوٹ کریں۔
4. ایک سادہ پروگرام (مثلاً، کیلکولیٹر) کھولیں اور ایک ایسا کام پہچانیں جو یہ انجام دیتا ہے جس میں سسٹم لائبریری کا استعمال ہو سکتا ہے (مثلاً، نمبرز کو ظاہر کرنا)۔
5. اپنی معلومات کو ان کالموں کے ساتھ ایک ٹیبل میں ریکارڈ کریں:
   آپریٹنگ سسٹم کا کمپونٹ — (OS Component) جیسے: شیل، ڈیوائس ڈرائیور، یا سسٹم لائبریری
   آپ کی مثال — (Your Example) مثلاً: ونڈوز ڈیسک ٹاپ، ایچ پی پرنٹر ڈرائیور، کیلکولیٹر ایپ
   مقصد / کردار — (Purpose / Role) یہ نظام میں کیا کردار ادا کرتا ہے یا کیا کام انجام دیتا ہے

1.3 آپریٹنگ سسٹم میں پروسیس مینجمنٹ

آپریٹنگ سسٹم کمپیوٹر پر چلنے والے تمام پروگراموں کو منظم کرنے کا ذمہ دار ہے۔ اس تناظر میں، ان چلنے والے پروگراموں کو پروسیسز (Processes) کہا جاتا ہے۔ پروسیس مینجمنٹ اس بات کو یقینی بناتا ہے کہ ہر پروسیس کو وہ وسائل ملیں جن کی اسے ضرورت ہے، یہاں تک کہ جب متعدد پروسیسز فعال ہوں۔ پروسیس مینجمنٹ ایک آپریٹنگ سسٹم کے سب سے اہم کاموں میں سے ایک ہے کیونکہ یہ یقینی بناتا ہے کہ تمام پروسیسز ایک دوسرے کو متاثر کیے بغیر درست کام کریں۔

پروسیس لائف سائیکل

ایک پروسیس اپنی لائف سائیکل کے دوران کئی مراحل سے گزرتا ہے:

1. تخلیق (Creation): یہ تب ہوتا ہے جب آپ کوئی بھی پروگرام شروع کرتے ہیں (جیسے MS Word)۔

OS پروگرام کو میموری میں لوڈ کرتا ہے اور اسے مطلوبہ وسائل (جیسے CPU کا وقت اور میموری) دیتا ہے۔

2. عمل در آمد (Execution): پروسیس فعال طور پر چل رہا ہے اور مؤثر طریقے سے کام انجام دے رہا ہے۔

3. اختتام (Termination): پروسیس اپنا کام مکمل کرتا ہے اور صارف یا سسٹم کے ذریعہ بند کر دیا جاتا ہے۔ OS وسائل کو آزاد کر دیتا ہے تاکہ انہیں دوسرے پروسیسز استعمال کر سکیں۔

مثال: آپ کے موبائل یا کمپیوٹر پر ویب براوزر کھولنے کا پروسیس لائف سائیکل (جیسے Google Chrome):

- **تخلیق:** جب صارف براوزر آئیکن پر کلک کرتا ہے تو پراسس شروع ہوتا ہے۔ آپریٹنگ سسٹم پروگرام کو میموری میں لوڈ کرتا ہے اور مطلوبہ وسائل مختص کرتا ہے۔
- **عملدرآمد:** براوزر کام انجام دیتا ہے، جیسے ویب پیجز لوڈ کرنا، میڈیا دکھانا، اور صارف کے اعمال کا جواب دینا۔
- **اختتام:** جب براوزر بند ہو جاتا ہے تو اختتام واقع ہوتا ہے۔ آپریٹنگ سسٹم پروسیس کو روکتا ہے اور اس کے وسائل کو دیگر پراسسز کے استعمال کے لیے فارغ کرتا ہے۔

ملٹی ٹاسکنگ اور ہم آہنگی (Multitasking and Concurrence)

جدید آپریٹنگ سسٹمز بہت سے پروسیسز کو اتنے مؤثر طریقے سے منظم کرتے ہیں کہ ایسا لگتا ہے کہ وہ سب ایک ہی وقت میں چل رہے ہیں۔

- ملٹی ٹاسکنگ (Multitasking): آپریٹنگ سسٹم ایک ہی صارف کے ذریعہ ایک ہی وقت میں ایک سے زیادہ پروگراموں کو کھولنے اور استعمال کرنے کی اجازت دیتا ہے۔ آپ ضرورت پڑنے پر آسانی سے ان کے درمیان سوئچ کر سکتے ہیں۔

مثال: آپ موسیقی سن سکتے ہیں، ایک دستاویز کو کھلا رکھ سکتے ہیں، اور انٹرنیٹ براوز کر سکتے ہیں، ضرورت کے مطابق ان کے درمیان سوئچ کر سکتے ہیں۔

- ہم آہنگی (Concurrency): ایک آپریٹنگ سسٹم میں ایک سے زیادہ پروسیسز ایک ہی وقت میں فعال ہوتے ہیں، لیکن CPU انہیں انتہائی تیزی سے دائروی عمل میں ایک کے بعد ایک پروسیس کرتا ہے۔

مثال: جیسے ایک شیف تین پکوان تیار کر رہا ہو، ایک پر تھوڑے وقت کے لیے کام کر رہا ہو، پھر دوسرے پر منتقل ہو رہا ہو، اور اس عمل کو دہراتا ہو، CPU پروسیسز کے درمیان اتنی تیزی سے سوئچ کرتا ہے کہ صارف کو کوئی تاخیر محسوس نہیں ہوتی۔

پروسیس شیڈولنگ کے تصورات (Process Scheduling Concepts)

ہم نے سیکھا کہ بہت سے پروسیسز ایک ہی وقت کی مدت کے دوران چل سکتے ہیں، لیکن CPU ان پر باری باری بہت تیزی سے لیکن ایک وقت میں ایک پر کام کرتا ہے۔ چونکہ CPU ایک ہی وقت میں تمام پروسیسز کو نہیں چلا سکتا، اس لیے آپریٹنگ سسٹم کو فیصلہ کرنا چاہیے کہ:

- کون سا پروسیس پہلے چلنا چاہیے؟

- ہر پروسیس کو کتنی دیر تک چلنا چاہیے؟

یہ فیصلہ سازی کا عمل شیڈولنگ (Scheduling) کہلاتا ہے۔ آپریٹنگ سسٹمز شیڈولنگ کے مختلف طریقے استعمال کرتے ہیں۔ لیکن اس باب میں، ہم صرف ایک سب سے آسان شیڈولنگ تکنیک، فرسٹ کم، فرسٹ سروڈ (First-Come, First-Served - FCFS) کو دیکھیں گے۔

FCFS طریقہ کار میں، CPU کاموں کو ان کی آمد کے عین مطابق ترتیب میں پروسیس کرتا ہے۔ سب سے پہلے آنے والا پروسیس سب سے پہلے مکمل کیا جاتا ہے، اور اگلا پروسیس صرف پچھلے کے ختم ہونے کے بعد ہی شروع ہوتا ہے۔

- مثال: یہ ایک دکان کے کاؤنٹر پر قطار کی طرح ہے، قطار میں موجود پہلا گاہک پہلے سرو کیا جاتا ہے، پھر اگلا، اور اسی طرح۔

عددی مثال (Numerical Example): آئیے یہ دیکھنے کے لیے کچھ سادہ ڈیٹا استعمال کریں کہ FCFS شیڈولنگ عملی طور پر کیسے کام کرتی ہے۔

پروسیس کی آئی-ڈی (Process ID)	آمد کا وقت (Arrival Time)	درکار وقت (Time Needed)
P1	0 سیکنڈ	5 سیکنڈ
P2	1 سیکنڈ	3 سیکنڈ
P3	2 سیکنڈ	2 سیکنڈ

مرحلہ وار وضاحت:

1. 0 سیکنڈ پر:
   - P1 سب سے پہلے پہنچتا ہے، لہٰذا یہ فوری طور پر چلنا شروع کر دیتا ہے۔
   - اسے ختم ہونے کے لیے 5 سیکنڈ درکار ہیں
   - یہ 0s سے 5s تک چلتا ہے۔
2. 1 سیکنڈ پر:
   - P2 پہنچتا ہے جبکہ P1 ابھی بھی چل رہا ہے، لہٰذا P2 انتظار کرتا ہے۔
3. 2 سیکنڈ پر:
   - P3 پہنچتا ہے، لیکن P1 ابھی بھی چل رہا ہے اور P2 پہلے ہی انتظار کر رہا ہے۔
   - FCFS میں، P3 کو بھی P1 اور P2 دونوں کے مکمل ہونے تک انتظار کرنا پڑے گا۔
4. 5 سیکنڈ پر:
   - P1 ختم ہو جاتا ہے۔
   - CPU اب P2 شروع کرتا ہے، جسے مکمل ہونے کے لیے 3 سیکنڈ درکار ہیں۔

- P2 5s سے 8s تک چلتا ہے۔

5. 8 سیکنڈ پر:

- P2 ختم ہو جاتا ہے۔
- CPU اب P3 شروع کرتا ہے، جسے 2 سیکنڈ درکار ہیں۔
- P3 شروع ہوتا ہے اور 8s سے 10s تک چلتا ہے۔

عملدر آمد کی ٹائم لائن:

[P1: 0–5] → [P2: 5–8] → [P3: 8–10]

0	5	8	10
P1	P2	P3	

P1: 0 سیکنڈ سے 5 سیکنڈ تک (اختتام 5 سیکنڈ پر)

P2: 5 سیکنڈ سے 8 سیکنڈ تک (اختتام 8 سیکنڈ پر)

P3: 8 سیکنڈ سے 10 سیکنڈ تک (اختتام 10 سیکنڈ پر)

مشاہدہ (Observation): اگرچہ P3 کی مدت سب سے کم ہے (صرف 2 سیکنڈ)، اسے اب بھی P1 اور P2 کے ختم ہونے تک انتظار کرنا پڑتا ہے کیونکہ وہ پہلے پہنچے تھے۔ یہ FCFS کی اہم خصوصیات میں سے ایک ہے؛ مختصر کام تاخیر کا شکار ہو سکتے ہیں اگر وہ طویل کاموں کے بعد پہنچیں۔

FCFS کے فوائد اور نقصانات (Advantages and Disadvantages of FCFS):

اگرچہ FCFS ایک سادہ اور منصفانہ شیڈولنگ طریقہ ہے، اس کے فوائد اور نقصانات دونوں ہیں:

فوائد (Advantages): سمجھنے اور لاگو کرنے میں آسان، کیونکہ پروسیسز کو ان کی آمد کے عین مطابق ترتیب میں سرو کیا جاتا ہے۔ کوئی بھی بعد میں آنے والا پروسیس پہلے آنے والے پروسس کے چلنے میں رکاوٹ نہیں ڈالتا۔

نقصانات (Disadvantages): مختصر پروسیسز کو طویل پروسیسز کے پیچھے قطار میں کھڑے ہونے کی صورت میں ایک طویل وقت تک انتظار کرنا پڑ سکتا ہے (Convoy Effect)، جو سسٹم کی مجموعی کارکردگی کو متاثر کر سکتا ہے۔

DO YOU KNOW کیا آپ جانتے ہیں؟

آپ کے کمپیوٹر کا CPU ایک سیکنڈ میں لاکھوں بار پروسیسز کے درمیان سوئچ کر سکتا ہے۔ یہ اتنی تیزی سے ہوتا ہے کہ آپ ایک ہی وقت میں موسیقی چلا سکتے ہیں، ویب براؤز کر سکتے ہیں، اور فائلیں ڈاؤن لوڈ کر سکتے ہیں۔

سرگرمی:

مقصد: فرسٹ-کم، فرسٹ-سروڈ (FCFS) طریقہ کار کا استعمال کرتے ہوئے پروسیسز کی ترتیب اور تکمیل کا وقت معلوم کریں۔

طریقہ کار:

1. درج ذیل پروسیسز پر غور کریں:

پروسیس (Process)	آمد کا وقت (Arrival Time)	درکار وقت (Time Needed)
P1	0s	4s
P2	1s	5s
P3	2s	3s

2. FCFS طریقہ کار کے مطابق پروسیسز کو اس ترتیب میں ترتیب دیں جس میں وہ چلیں گے۔
3. ہر پروسیس کے لیے شروع ہونے اور ختم ہونے کے وقت کا حساب لگائیں۔

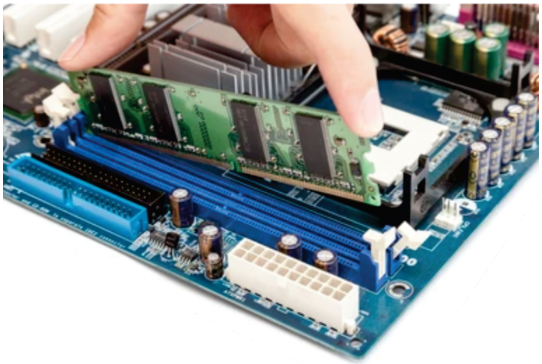
سوالات کے جواب دیں

- o تمام پروسیسز کے ختم ہونے میں کتنا کل وقت لگا؟
- o کس پروسیس کو سب سے زیادہ انتظار کرنا پڑا؟

1.4 میموری (Memory)

میموری کمپیوٹر سسٹم کا ایک بنیادی جزو ہے، جو پروسیسنگ کے لیے درکار ڈیٹا اور ہدایات کو محفوظ یا ذخیرہ کرنے کے لیے استعمال ہوتی ہے۔ یہ یقینی بناتی ہے کہ CPU پروگرام پر عملدر آمد کے دوران معلومات تک تیزی سے رسائی حاصل کر سکے۔

پرائمری میموری (Primary Memory - RAM)



شکل: 1.2 RAM

پرائمری میموری، جسے رینڈم ایکسیس میموری (RAM) بھی کہا جاتا ہے، کمپیوٹر کا بنیادی ورکنگ ایریا ہے۔ یہ ایک تیز رفتار اسٹوریج ایریا ہے جہاں کمپیوٹر ڈیٹا اور ہدایات کو عارضی طور پر محفوظ کرتا ہے۔ جب کمپیوٹر بند کر دیا جاتا ہے تو اس کا ڈیٹا ختم ہو جاتا ہے۔

مثال: جب آپ ایک Word دستاویز کھولتے ہیں، تو کمپیوٹر تیزی سے پروگرام اور دستاویز کو RAM میں رکھ دیتا ہے تاکہ CPU ان پر فوری طور پر کام کر سکے۔

ورچوئل میموری (Virtual Memory)

جب RAM بھر جاتی ہے، تو آپریٹنگ سسٹم کمپیوٹر کی اسٹوریج ڈرائیو (جیسے ہارڈ ڈرائیوز، سالڈ-اسٹیٹ ڈرائیوز (SSD)، اور Non-Volatile Memory (NVMe)) کا کچھ حصہ ورچوئل میموری (Virtual Memory) کے طور پر استعمال کرتا ہے۔ یہ اضافی جگہ عارضی RAM کی طرح کام کرتی ہے، جو زیادہ پروگراموں کو ایک ہی وقت میں چلنے کی اجازت دیتی ہے۔

اسٹوریج ڈرائیوز RAM کے مقابلے میں کم ڈیٹا کی منتقلی کی رفتار اور زیادہ رسائی کے اوقات پر کام کرتی ہیں؛ اسی وجہ سے ورچوئل میموری کا استعمال سسٹم کی کارکردگی میں کمی کا سبب بن سکتا ہے۔ جب کئی پروگرام ایک ساتھ کھلے ہوتے ہیں، تو آپریٹنگ سسٹم کم فعال پروگراموں سے ڈیٹا کو ورچوئل میموری میں منتقل کر سکتا ہے۔ یہ ان پروگرامز کے لیے RAM میں جگہ خالی کرتا ہے جو آپ فی الحال استعمال کر رہے ہیں، لیکن ان کم فعال پروگرامز کو جواب دینے میں سست کر سکتا ہے۔

کیا آپ جانتے ہیں؟

پہلا پرسنل کمپیوٹر، IBM 5150 (1981)، صرف 16 KB RAM سے لیس تھا۔ اس کے برعکس، جدید DDR5 RAM, 50GB فی سیکنڈ سے زیادہ کی رفتار سے ڈیٹا منتقل کر سکتی ہے، جس سے ایک مکمل ہائی-ڈیفینیشن مووی کو ایک سیکنڈ سے بھی کم وقت میں کاپی کیا جا سکتا ہے۔

1.5 تھریڈز vs پروسیسز (Threads vs. Processes)

پروسیس (Process): ایک ایسا پروگرام ہے جس پر فی الحال کمپیوٹر کے ذریعے عملدرآمد کیا جا رہا ہے۔ اس کی اپنی میموری کی جگہ، CPU کا وقت، اور دیگر وسائل (جیسے فائلیں یا نیٹ ورک کنکشن) ہوتے ہیں۔ پروسیسز استحکام اور سیکورٹی کو یقینی بنانے کے لیے ایک دوسرے سے الگ تھلگ ہوتے ہیں؛ ایک پروسیس میں مسئلہ عام طور پر دوسرے کو متاثر نہیں کرتا۔

تھریڈ (Thread): دوسری طرف تھریڈ ایک پروسیس کے اندر عملدرآمد کی سب سے چھوٹی اکائی ہے۔ ایک ہی پروسیس کے اندر متعدد تھریڈز موجود ہو سکتے ہیں، ہر ایک مختلف کام انجام دے رہا ہوتا ہے۔ ایک ہی پروسیس میں تمام تھریڈز ایک ہی میموری اور وسائل کو شیئر کرتے ہیں، لیکن آزادانہ طور پر کام کرتے ہیں۔

مثال: ایک ویب براؤزر پر غور کریں:

- پورا براؤزر ایپلیکیشن ایک پروسیس ہے۔
- اس پروسیس کے اندر، مختلف تھریڈز اس کے ذمہ دار ہیں:
- ایک تھریڈ ایک ویب پیج کو لوڈ اور رینڈر (تصویر کشی) کر رہا ہے۔
- دوسرا تھریڈ آڈیو یا ویڈیو مواد چلا رہا ہے۔
- ایک اور تھریڈ پس منظر میں فائلیں ڈاؤن لوڈ کر رہا ہے۔
- اوپر دیئے گئے متعدد تھریڈز کا استعمال کرتے ہوئے، براؤزر ویڈیو چلاتے ہوئے بھی نیا مواد لوڈ کرنا جاری رکھ سکتا ہے

- صارف ایک کام کے ختم ہونے کا انتظار کیے بغیر دوسرا کام شروع کر سکتا ہے۔

ملٹی تھریڈنگ

ملٹی تھریڈنگ ایک آپریٹنگ سسٹم کی تکنیک ہے جو ایک ہی پروسیس کو اپنے کام کو تھریڈز نامی چھوٹی اکائیوں میں تقسیم کر کے ایک ہی وقت میں متعدد کام انجام دینے کی اجازت دیتی ہے۔ ہر تھریڈ آزادانہ طور پر چلتا ہے لیکن پروسیس کے ہی میموری اور وسائل کو شیئر کرتا ہے، جو تیز رفتار عملدر آمد، بہتر جوابدہی، اور سسٹم کے وسائل کے موثر استعمال کو ممکن بناتا ہے۔

ملٹی تھریڈنگ کے فوائد: ملٹی تھریڈنگ کئی فوائد پیش کرتی ہے، جن میں سے کچھ اہم ذیل میں زیر بحث ہیں:

- **بہتر کارکردگی (Enhanced Performance):**

کاموں کو متعدد تھریڈز میں تقسیم کیا جاسکتا ہے اور متوازی طور پر عملدر آمد کیا جاسکتا ہے، جس سے پیچیدہ آپریشنز تیزی سے مکمل ہو سکتے ہیں اور مجموعی سسٹم کی کارکردگی میں بہتری آتی ہے۔

- **بہتر ردعمل (Improved Responsiveness):**

جب ایک تھریڈ کسی مخصوص کام میں مصروف ہوتا ہے تب بھی ایپلی کیشنز ردعمل دیتی رہتی ہیں۔

مثال: ایک ورڈ پروسیسر مسلسل ٹائپنگ کو ممکن بناتا ہے جبکہ دوسرا تھریڈ پس منظر میں ہجے کی جانچ (Spell Checking) کرتا ہے۔

- **ہم آہنگ (متوازی) آپریشنز کے لیے سپورٹ (Support for Concurrent Parallel Operations):**

متعدد کام ایک ہی مدت کے دوران ہو سکتے ہیں۔ جو خاص طور پر گیمز، ویڈیو ایڈیٹنگ، اور ریئل ٹائم مواصلاتی ٹولز جیسی ایپلی کیشنز میں اہم ہے۔

- **وسائل کا موثر استعمال (Efficient Use of Resources):**

تھریڈز اپنے پیرنٹ پروسیس کی ہی میموری اور وسائل کو شیئر کرتے ہیں، جس سے الگ پروسیسز بنانے کے مقابلے میں سسٹم کے وسائل کی ضرورت کم ہوتی ہے۔

1.6 سسٹم کالز

سسٹم کال ایک درخواست ہے جو کسی پروگرام کے ذریعے آپریٹنگ سسٹم سے ایک مخصوص کام انجام دینے کے لیے کی جاتی ہے جو پروگرام براہ راست نہیں کر سکتا۔ وہ صارف کے پروگرامنگ اور کرنل (kernel) کے درمیان ایک پل کا کام کرتی ہیں، جس سے ایپلی کیشنز کو ہارڈ ویئر اور آپریٹنگ سسٹم کے بنیادی افعال تک محفوظ طریقے سے رسائی حاصل کرنے کی اجازت ملتی ہے۔ سسٹم کالز کے بغیر، وہ پروگرام جو براہ راست ہارڈ ویئر کو کنٹرول کرتے ہیں، غیر محفوظ اور پیچیدہ ہو سکتے ہیں۔

مثال: جب آپ ایک ٹیکسٹ ایڈیٹر میں کوئی فائل محفوظ کرتے ہیں، تو پروگرام آپریٹنگ سسٹم کو اسٹوریج ڈرائیو (جیسے ہارڈ ڈرائیو) پر ڈیٹا لکھنے کے لیے کہنے کے لیے ایک سسٹم کال کا استعمال کرتا ہے۔

سسٹم کالز کی اقسام

سسٹم کالز کی اہم اقسام میں شامل ہیں:

1. open: پڑھنے یا لکھنے کے لیے ایک فائل کھولتا ہے۔
   - مثال: چلانے کے لیے ایک میوزک فائل کھولنا۔
2. read: ایک فائل یا ان پٹ ڈیوائس سے ڈیٹا حاصل کرتا ہے۔
   - مثال: کی بورڈ سے ان پُٹ لینا
3. write: ایک فائل یا آؤٹ پٹ ڈیوائس کو ڈیٹا بھیجتا ہے۔
   - مثال: LED/LCD پر ایک تصویر دیکھنا۔
4. fork: ایک موجودہ پروسیس کی نقل کر کے ایک نیا پروسیس بناتا ہے۔
   - مثال: ایک نیا براوزر ٹیب کھولنا، جہاں OS ایک اور پروسیس بنانے کے لیے fork کا استعمال کرتا ہے۔

کیا آپ جانتے ہیں؟

لینکس اور میک او ایس میں چند سو سسٹم کالز ہیں، جبکہ ونڈوز تقریباً 2,000 سسٹم کالز استعمال کرتا ہے۔ یہ کالز فائلوں کو کھولنے سے لے کر ایپس کو چلانے اور آپ کے کام کے دوران پس منظر میں گرافکس دکھانے تک سب کچھ سنبھالتی ہیں۔

1.7 فائل سسٹم کی ساخت اور انتظام (File System Structure and Management)

ایک آپریٹنگ سسٹم نہ صرف پروگرام چلاتا ہے بلکہ صارفین کے ڈیٹا کو بھی محفوظ اور منظم کرتا ہے۔ یہ فائل سسٹم کے ذریعے حاصل کیا جاتا ہے، جو صارفین اور ایپلی کیشنز کو اسٹوریج ڈیوائسز پر معلومات کو ذخیرہ کرنے، تلاش کرنے اور منظم کرنے کا ایک منظم طریقہ فراہم کرتا ہے۔ ایک اچھی طرح سے ڈیزائن کیا گیا فائل سسٹم اس بات کو یقینی بناتا ہے کہ ڈیٹا منظم، محفوظ، اور آسانی سے قابل بازیافت رہے، یہاں تک کہ جب ہزاروں فائلیں پیچیدہ ڈیٹا پر مشتمل ہوں۔

فائلیں (Files)

فائل کمپیوٹر پر ذخیرہ شدہ متعلقہ ڈیٹا کا ایک مجموعہ ہے۔ اس میں ٹیکسٹ، تصاویر، آڈیو، ویڈیو، یا پروگرام کی ہدایات شامل ہو سکتی ہیں۔

فولڈرز (Folders)

ایک فولڈر (جسے ڈائریکٹری بھی کہا جاتا ہے) ایک کنٹینر ہے جو فائلوں اور دیگر فولڈرز کو ایک منطقی ساخت میں منظم کرنے کے لیے استعمال ہوتا ہے، جس سے معلومات کو تلاش کرنا اور منظم کرنا آسان ہو جاتا ہے۔

میٹاڈیٹا (Metadata)

میٹاڈیٹا سے مراد ایک فائل کے بارے میں تفصیلات ہے، جیسے اس کا نام، قسم، سائز، تخلیق کی تاریخ، اور آخری ترمیم کی تاریخ۔ یہ معلومات آپریٹنگ سسٹم اور صارف دونوں کو فائلوں کو کھولے بغیر ان کی شناخت اور منظم کرنے میں مدد کرتی ہیں۔

فائل سسٹمز (File Systems)

ایک فائل سسٹم وہ طریقہ ہے جس سے ایک آپریٹنگ سسٹم ایک اسٹوریج ڈیوائس، جیسے ہارڈ ڈرائیو، SSD، یا USB پر فائلوں کو ذخیرہ اور منظم کرتا ہے۔ یہ فیصلہ کرتا ہے کہ ہر فائل کہاں رکھی جائے گی اور اس کے مقام کو یاد رکھتا ہے تاکہ بعد میں اس تک رسائی حاصل کی جا سکے۔

کچھ بڑے پیمانے پر استعمال ہونے والے فائل سسٹمز میں درج ذیل شامل ہیں:

- FAT 32: USB فلیش ڈرائیوز میں استعمال ہوتا ہے۔
- NTFS: ونڈوز OS میں استعمال ہوتا ہے۔
- HFS+ / APFS: macOS میں استعمال ہوتا ہے۔
- EXT4: عام طور پر لینکس OS کے لیے استعمال ہوتا ہے۔

فائل سسٹم کا انتخاب کارکردگی، اسٹوریج کی صلاحیت اور سیکیورٹی خصوصیات کا تعین کرتا ہے۔

مثال: اگر آپ اپنے کمپیوٹر پر ایک تصویر محفوظ کرتے ہیں، تو فائل سسٹم یقینی بناتا ہے کہ یہ صحیح جگہ پر ذخیرہ ہو اور جب آپ اسے دوبارہ کھولتے ہیں تو اسے کہاں تلاش کرنا ہے، یہ جانتا ہے، ایک لائبریرین کی طرح جو ایک کتاب کو صحیح سیکشن میں رکھتا ہے اور جانتا ہے کہ بعد میں اسے کہاں تلاش کرنا ہے۔

سرگرمی:

مقصد: مظاہرہ کریں کہ فائلز کا انتظام کرتے وقت سسٹم کالز کیسے کام کرتی ہیں۔

ہدایات:

تیاری: ہر طالب علم کو ایک فائل کے نام کا لیبل لگا ہوا ایک چھوٹا کارڈ دیں۔ (مثلاً، photo.jpg، homework.docx، song.mp3)

تختہ سیاہ (board) پر open، read، write، اور close لکھیں۔

طریقہ کار:

1. ایک طالب علم کو آپریٹنگ سسٹم کہیں۔
2. ایک اور طالب علم ایپلیکیشن کا کردار ادا کرتا ہے (مثلاً، فوٹو ویور)۔
3. ایپلیکیشن درخواست کرتی ہے: "OS، براہ کرم photo.jpg کھولیں"۔
4. OS فائل کی فہرست سے کارڈ منتخب کر کے اسے طالب علم کے حوالے کر کے جواب دیتا ہے۔
5. مختلف فائلوں کے لیے read، write، اور close کے ساتھ دہرائیں۔

بحث کا اختتام: وضاحت کریں کہ ہر عمل ایک سسٹم کال کی نمائندگی کرتا ہے۔

1.8 آپریٹنگ سسٹمز کی اقسام

آپریٹنگ سسٹمز کو ڈیوائس کی ضروریات اور اس کے انجام دینے والے کام کے مطابق ڈیزائن کیا جاتا ہے۔ ہر قسم ایک مختلف مقصد کی تکمیل کرتی ہے اور مخصوص کاموں کے لیے موزوں ہوتی ہے۔

ریئل-ٹائم آپریٹنگ سسٹم (RTOS)

ایک ریئل-ٹائم آپریٹنگ سسٹم (RTOS) کو ایک مقررہ وقت (deadline) کے اندر ڈیٹا پر کارروائی کرنے اور جواب دینے کے لیے ڈیزائن کیا ہے۔ یہ وہاں استعمال ہوتا ہے جہاں معمولی تاخیر بھی سسٹم کی ناکامی یا سنگین نتائج کا سبب بن سکتی ہے۔

- اہم خصوصیت: کاموں کو پہنچتے ہی فوری طور پر پروسیس کرتا ہے۔
- کہاں استعمال ہوتا ہے: حساس سسٹمز جیسے ہوائی ٹریفک کنٹرول، دل کی نگرانی کرنے والے آلات، یا صنعتی روبوٹس وغیرہ۔

ایمبیڈڈ آپریٹنگ سسٹم (Embedded Operating System) (EOS)

ایک ایمبیڈڈ OS (Embedded OS) ایک چھوٹا اور انتہائی موثر آپریٹنگ سسٹم ہے جو صرف ان افعال کو کنٹرول کرنے کے لیے ایک مخصوص ڈیوائس میں بنایا جاتا ہے جن کی اسے ضرورت ہوتی ہے۔ یہ عام مقصد کی کمپیوٹنگ کے لیے نہیں ہے بلکہ ایک کام یا کاموں کے ایک چھوٹے سیٹ کے لیے موزوں ہے۔

- اہم خصوصیت: بہت کم میموری اور بجلی استعمال کرتا ہے، اور اکثر ڈیوائس کے اندر مستقل طور پر محفوظ ہوتا ہے۔
- کہاں استعمال ہوتا ہے: گھریلو آلات (مائیکروویوز، واشنگ مشینیں)، پرنٹرز، سمارٹ ٹی وی، اور ATM وغیرہ۔

نیٹ ورک آپریٹنگ سسٹم (Network Operating System) (NOS)

انیٹ ورک آپریٹنگ سسٹم (Netwrok OS) ایک نیٹ ورک کے ذریعے جڑے ہوئے متعدد کمپیوٹرز کو منظم اور سپورٹ کرتا ہے۔ یہ صارفین کے درمیان فائلز، پرنٹرز، اور انٹرنیٹ کنکشنز جیسے وسائل کو شیئر کرواتا ہے۔ (جیسے ونڈوز ملٹی پوائنٹ سرورز)۔

- اہم خصوصیت: کمپیوٹرز کے درمیان مواصلات اور ہم آہنگی پر توجہ مرکوز کرتا ہے۔
- کہاں استعمال ہوتا ہے: دفاتر، اسکول، اور ڈیٹا سینٹرز۔

مثال: اسکول کی کمپیوٹر لیب میں، طلباء مختلف کمپیوٹر استعمال کرتے ہیں، لیکن وہ سب ایک ہی سرور پر فائلیں محفوظ کر سکتے ہیں اور ایک ہی پرنٹر سے پرنٹ کر سکتے ہیں۔

موبائل آپریٹنگ سسٹم (Mobile Operating System)

ایک موبائل (Mobile OS) سمارٹ فونز، ٹیبلٹس، اور دیگر ہینڈ ہیلڈ ڈیوائسز کے لیے ڈیزائن کیا گیا ہے۔ یہ ٹچ-اسکرین کے استعمال، بیٹری کی بچت، اور موبائل ایپس کے لیے موزوں ہے۔

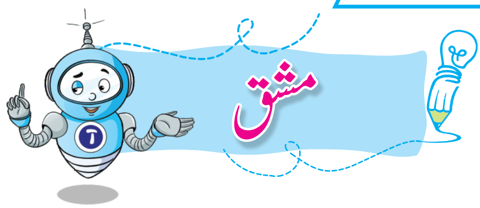
- اہم خصوصیت : وائرلیس کنیکٹیوٹی، کیمرے، سینسرز، اور ایپ اسٹورز کو سپورٹ کرتا ہے۔
- کہاں استعمال ہوتا ہے : سمارٹ فونز، ٹیبلٹس، سمارٹ واچز۔

کیا آپ جانتے ہیں؟

سب سے پہلے موبائل آپریٹنگ سسٹمز، جیسے Symbian (جو ابتدائی نوکیا فونز میں استعمال ہوتا تھا)، صرف چند چھوٹی ایپس چلا سکتے تھے۔ آج کے موبائل OS پلیٹ فارمز جیسے اینڈرائیڈ اور آئی او ایس لاکھوں ایپس، تھری ڈی گیمز، اور یہاں تک کہ پیشہ ورانہ ویڈیو ایڈیٹنگ ٹولز کو بھی سنبھال سکتے ہیں۔

خلاصہ (Summary)

- آپریٹنگ سسٹم (OS) ایک قسم کا سسٹم سافٹ ویئر ہے جو ہارڈ ویئر کا انتظام کرتا ہے، ایپلی کیشن سافٹ ویئر چلاتا ہے، اور صارف کو انٹرفیس فراہم کرتا ہے۔
- کرنل (Kernel) آپریٹنگ سسٹم کا مرکزی حصہ ہے، جو براہ راست کمپیوٹر کے ہارڈ ویئر اور سسٹم سافٹ ویئر جیسے کہ CPU، میموری اور دیگر آلات کو کنٹرول کرتا ہے۔
- شیل (Shell) آپریٹنگ سسٹم کا بیرونی حصہ ہے جو صارف کے ساتھ رابطہ (interact) کرتا ہے۔
- سسٹم لائبریریاں تیار شدہ ہدایات کے مجموعے ہیں جنہیں پروگرام عام کام انجام دینے کے لیے استعمال کر سکتے ہیں، جیسے فائلیں کھولنا یا اسکرین پر تحریر دکھانا۔
- آپریٹنگ سسٹم کمپیوٹر پر چلنے والے تمام پروگراموں کے انتظام کا ذمہ دار ہے۔ اس تناظر میں، ان چلنے والے پروگراموں کو پراسیس کہا جاتا ہے۔
- ملٹی ٹاسکنگ میں، آپریٹنگ سسٹم ایک ہی وقت میں ایک ہی صارف کو ایک سے زیادہ پروگرام کھولنے اور استعمال کرنے کی اجازت دیتا ہے۔
- کانکورنسی (Concurrency) میں، آپریٹنگ سسٹم میں ایک ہی وقت میں ایک سے زیادہ پراسیس فعال ہوتے ہیں، لیکن CPU انہیں انتہائی تیز رفتار چکروں میں ایک ایک کر کے پروسیس کرتا ہے۔
- FCFS طریقہ کار میں، CPU کاموں کو اسی ترتیب سے پروسیس کرتا ہے جس ترتیب سے وہ آتے ہیں۔ سب سے پہلے آنے والا پراسیس پہلے مکمل ہوتا ہے، اور اگلا پراسیس پچھلے کے ختم ہونے کے بعد ہی شروع ہوتا ہے۔
- جب RAM بھر جاتی ہے، تو آپریٹنگ سسٹم کمپیوٹر کے اسٹوریج ڈرائیو جیسے ہارڈ ڈرائیوز، سالڈ اسٹیٹ ڈرائیوز (SSD)، اور Non Volatile Memory Express (NVMe) کے کچھ حصے کو ورچوئل میموری کے طور پر استعمال کرتا ہے۔
- پراسیس ایک آزاد پروگرام ہے جو اس وقت کمپیوٹر کے ذریعے چلایا جا رہا ہوتا ہے۔ دوسری طرف، تھریڈ (thread) ایک پراسیس کے اندر عملدرآمد کی سب سے چھوٹی اکائی ہے۔
- سسٹم کال (System call) ایک ایسی درخواست ہے جو ایک پروگرام کی طرف سے آپریٹنگ سسٹم کو کسی خاص کام کو انجام دینے کے لیے دی جاتی ہے جسے پروگرام براہ راست نہیں کر سکتا۔



کثیر انتخابی سوالات

i درج ذیل میں سے کون سی چیز آپریٹنگ سسٹم کی مثال نہیں ہے؟

الف لینکس(Linux)　　ب ونڈوز(Windows)

ج فوٹوشاپ(Photoshop)　　د میک او ایس(macOS)

ii ملٹی یوزر ماحول میں، OS اس بات کو یقینی بناتا ہے:

الف مساوی رسائی اور ڈیٹا کی رازداری(Equal access and data privacy)

ب تمام صارفین ایک ہی فائلیں شیئر کرتے ہیں

ج ہارڈویئر کبھی استعمال نہیں ہوتا

د انٹرنیٹ کی رفتار بڑھ جاتی ہے

iii ونڈوز میں، یوزر اکاؤنٹس کس کے ذریعے بنائے جاسکتے ہیں؟

الف ٹاسک مینیجر(Task Manager)　　ب کنٹرول پینل یا سیٹنگز(Control Panel or Settings)

ج ڈسک مینجمنٹ(Disk Management)　　د BIOS سیٹنگز

iv OS کا بنیادی حصہ جو براہ راست ہارڈویئر کے ساتھ تعامل کرتا ہے، وہ ہے:

الف شیل(Shell)　　ب کرنل(Kernel)

ج ڈیوائس ڈرائیور(Device Driver)　　د سسٹم لائبریری(System Library)

v گرافیکل شیل یوزر کو اجازت دیتا ہے:

الف صرف کمانڈ ٹائپ کرنا

ب آئیکن پر کلک کرنا اور مینوز کا استعمال کرنا (Click icon and use menus)

ج صرف کوڈ کے ذریعے رابطہ کرنا

د صرف BIOS تک رسائی حاصل کرنا

vi FCFS شیڈولنگ طریقہ کار میں، پروسیسز کو سرو کیا جاتا ہے:

الف بے ترتیب (Randomly)
ب سب سے پہلے مختصر کام
ج اسی ترتیب میں جس میں وہ پہنچتے ہیں (In the order they arrive)
د صرف ترجیح (priority) کے لحاظ سے

vii ایک ہی پروسیس میں تھریڈز:

الف الگ میموری کی جگہ رکھتے ہیں
ب ایک ہی میموری اور وسائل کو شیئر کرتے ہیں۔
ج مختلف OS پر چلتے ہیں
د ایک ساتھ نہیں چل سکتے

viii نیا پروسیس بنانے کے لیے کون سا سسٹم کال استعمال ہوتا ہے؟

الف اوپن
ب ریڈ (Read)
ج رائٹ (Write)
د فورک

مختصر سوالات

1. آپریٹنگ سسٹم کو مرکزی کنٹرولر کیوں کہا جاتا ہے؟
2. OS صارف اور ہارڈویئر کے درمیان مترجم کے طور پر کیسے کام کرتا ہے؟
3. ونڈوز میں نیا یوزر اکاؤنٹ بنانے کا ایک طریقہ بیان کریں۔
4. کرنل شیل سے کیسے مختلف ہے؟
5. ایک سسٹم لائبریری اور اس کے مقصد کی ایک مثال دیں۔
6. FCFS شیڈولنگ کا ایک فائدہ اور ایک نقصان بیان کریں۔
7. ورچوئل میموری RAM سے سست کیوں ہے؟
8. سسٹم کال کی تعریف کریں۔
9. فائل سسٹم کا کردار کیا ہے؟
10. کمپیوٹر سسٹم میں ملٹی ٹاسکنگ کی تعریف کریں۔

تفصیلی سوالات

1. آپریٹنگ سسٹم کے آرکیٹیکچر کو بیان کریں، جس میں کرنل، شیل، اور لیئرڈ ڈیزائن کی مثال کے ساتھ وضاحت کی جائے۔
2. پروسیس لائف سائیکل کی مثال کے ساتھ تفصیل سے وضاحت کریں۔
3. RAM اور ورچوئل میموری کے درمیان فرق واضح کریں، اور وضاحت کریں کہ ملٹی تھریڈنگ کارکردگی کو کیسے بہتر بناتی ہے؟
4. سسٹم کالز کیا ہیں؟ ان کے مقصد کو بیان کریں، اور کم از کم تین اقسام کی مثالیں دیں۔

5. ایک کمپیوٹر سسٹم پروسیسز کو منظم کرنے کے لیے فرسٹ-کم، فرسٹ-سروڈ (FCFS) شیڈولنگ طریقہ کار استعمال کرتا ہے۔
درج ذیل ٹیبل میں مین پانچ پروسیسز کی آمد کا وقت اور CPU کے کام کرنے (Burst) کا وقت (عملدرآمد کے لیے درکار وقت) دکھایا گیا ہے:

پروسیس (Process)	آمد کا وقت (Arrival Time - seconds)	برسٹ کا وقت (Burst Time - seconds)
P1	0	4
P2	1	3
P3	2	1

ضروریات (Requirements):

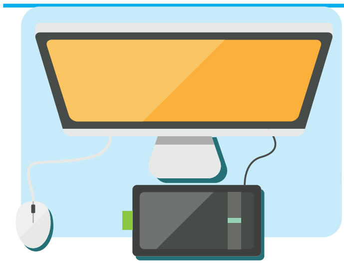
- FCFS شیڈولنگ طریقہ کار کے مطابق پروسیسز کو اس ترتیب میں لکھیں جس میں وہ چلائے جائیں گے۔
- تمام پروسیسز کی عملدرآمد کی ترتیب دکھائیں۔
- ہر پروسیس کے لیے شروع ہونے کا وقت اور تکمیل کے وقت کا حساب لگائیں۔
- ہر پروسیس کے لیے انتظار کا وقت (Waiting Time) معلوم کریں۔
- تمام پروسیسز کے لیے اوسط انتظار کا وقت (Average Waiting Time) کا حساب لگائیں۔

کثیر انتخابی سوالات کے جوابات

1. ج فوٹوشاپ (Photoshop)
2. الف مساوی رسائی اور ڈیٹا کی رازداری (Equal access and data privacy)
3. ب کنٹرول پینل یا سیٹنگز (Control Panel or Settings)
4. ب کرنل (Kernel)
5. ب آئیکنز پر کلک کرنا اور مینیوز کا استعمال کرنا (Click icons and use menus)
6. ج اسی ترتیب میں جس میں وہ پہنچتے ہیں (In the order they arrive)
7. ب ایک ہی میموری اور وسائل کو شیئر کرتے ہیں (Share the same memory and resources)
8. د fork

یونٹ

2 سسٹم ریکوری اینڈ مینٹیننس



حاصلاتِ تعلم: اس باب کے اختتام تک، آپ اس قابل ہو جائیں گے کہ:

- پوسٹ ٹربل شوٹنگ مینٹیننس کی تعریف کر سکیں اور سسٹم میں بار بار پیدا ہونے والے مسائل کی روک تھام میں اس کے کردار کی وضاحت کر سکیں۔
- ٹاسک مینیجر، ریسورس مانیٹر، اور سسٹم لاگز کا جائزہ لے کر CPU، میموری، اور ڈسک کے استعمال کی نگرانی کر سکیں، اور نان رسپونسو ایپلیکیشنز کو ختم کر سکیں۔
- ونڈوز کو سیف موڈ (Safe Mode) میں شروع کر سکیں اور وضاحت کر سکیں کہ یہ موڈ کب کار آمد ہوتا ہے۔
- سسٹم کو پچھلی کام کرنے والی حالت میں واپس لانے کے لیے ری سٹور پوائنٹ بنا اور استعمال کر سکیں۔
- سسٹم ری سٹور اور سسٹم ری سیٹ کے درمیان فرق بتا سکیں، اور شناخت کر سکیں کہ ہر ایک کو کب استعمال کرنا ہے۔
- بوٹ ایبل ریکوری میڈیا (Bootable Recovery Media) کے مقصد اور اس کی ضرورت کی وضاحت کر سکیں۔
- روفس (Rufus) یا ونڈوز کے بلٹ ان ٹولز کا استعمال کرتے ہوئے، درست پارٹیشن سٹائل (MBR یا GPT) کا انتخاب کرتے ہوئے، ایک بوٹ ایبل USB بنا سکیں۔
- GPT اور MBR کے درمیان فرق کی شناخت کر سکیں اور کسی ڈسک کے پارٹیشن سٹائل کو چیک کر سکیں۔
- سسٹم سٹارٹ اپ میں BIOS اور UEFI کے کردار کی وضاحت کر سکیں۔
- پاور-آن سیلف-ٹیسٹ (POST) کے عمل اور اس کی اہمیت کو بیان کر سکیں۔
- BIOS/UEFI سیٹنگز تک رسائی حاصل کر کے بوٹ آرڈر کو تبدیل کر سکیں یا ڈیفالٹس پر ری سیٹ کر سکیں۔
- بلٹ ان ٹولز کے ساتھ ساتھ ٹینور شیئر (Tenorshare) جیسے تھرڈ پارٹی ریکوری ٹولز کا استعمال کرتے ہوئے ڈیلیٹ شدہ فائلوں کو ریکور کر سکیں۔
- عارضی اور غیر ضروری فائلوں کو ختم کرنے کے لیے ڈسک کلین اپ کر سکیں۔
- سیکورٹی اور استحکام کو بہتر بنانے کے لیے سافٹ ویئر اپ ڈیٹس (Patch Management) کا اطلاق کر سکیں۔
- ونڈوز ڈیفنڈر کا استعمال کرتے ہوئے وائرس سکین چلا سکیں اور ریئل ٹائم پروٹیکشن کو فعال کر سکیں۔
- باقاعدہ اپ ڈیٹس، کلین اپس، اور سکینز کو شیڈول کرنے کے لیے ایک مینٹیننس کیلنڈر بنا سکیں۔
- معیاری ٹیمپلیٹ کا استعمال کرتے ہوئے ریپیئر اور اپ ڈیٹ لاگز کو برقرار رکھ سکیں۔
- درست سسٹم مینٹیننس ریکارڈز رکھنے کی اہمیت کی وضاحت کر سکیں۔

تعارف

اس باب میں، ہم ٹربل شوٹنگ کے بعد کمپیوٹر سسٹم کی حالت کو برقرار رکھنے اور بحال کرنے کی وضاحت کریں گے۔ آپ ٹاسک مینیجر، ریسورس مانیٹر، اور ایونٹ ویوور (Event Viewer) جیسے آلات کا استعمال کرتے ہوئے سسٹم کی صحت کی نگرانی کرنا سیکھیں گے، اور سیف موڈ، سسٹم ری سٹور، اور بوٹ ایبل ڈرائیوز جیسے ریکوری آپشنز کو دریافت کریں گے۔

یہ باب بیسک ان پٹ / آؤٹ پٹ سسٹم (BIOS) اور یونیفائیڈ ایکسٹینسیبل فرم ویئر انٹرفیس (UEFI) کے سسٹم سٹارٹ اپ میں کردار، سادہ ڈیٹا ریکوری تکنیکوں، اور ڈسک کلین اپ، اپ ڈیٹس، اور وائرس سکینز جیسے اہم حفاظتی مینٹیننس (preventive maintenance) کے کاموں کو بھی متعارف کراتا ہے۔ اس باب کے اختتام تک، آپ سمجھ جائیں گے کہ سسٹم لاگز اور مینٹیننس ریکارڈز کو کیسے رکھنا ہے تاکہ ایک کمپیوٹر سسٹم کو آسانی سے چلایا جاسکے اور عام ریکوری کے کاموں کو مؤثر طریقے سے سنبھالا جاسکے۔

2.1 پوسٹ ٹربل شوٹنگ کا تعارف

عام کمپیوٹر مسائل کو ٹھیک کرنے کے لیے ٹربل شوٹنگ کے اقدامات، جن میں سسٹم کو ری سٹارٹ کرنا، فزیکل کنکشنز کو چیک کرنا، اور سافٹ ویئر یا ہارڈ ویئر سے متعلق سادہ مسائل کو حل کرنا شامل ہے اگلا مرحلہ یہ سیکھنا ہے کہ مسئلہ حل ہونے کے بعد کیا اقدامات کیے جائیں۔ اس عمل کو پوسٹ ٹربل شوٹنگ مینٹیننس کہا جاتا ہے۔ اس میں اضافی اقدامات شامل ہیں جو کمپیوٹر کو صحیح طریقے سے کام جاری رکھنے اور مستقبل میں وہی مسائل دوبارہ پیدا ہونے سے بچنے میں مدد کرتے ہیں۔ یہ باب وضاحت کرتا ہے کہ کمپیوٹر سسٹم کو کس طرح برقرار رکھا جائے اور اسے مستحکم، محفوظ، موثر اور قابل اعتماد رکھا جائے۔

مسلسل مینٹیننس کی اہمیت

کسی کمپیوٹر کے مسئلے کو حل کرنا صرف پہلا مرحلہ ہے۔ سسٹم کو صحیح طریقے سے کام جاری رکھنے کے لیے اب بھی باقاعدہ دیکھ بھال کی ضرورت ہے۔ مینٹیننس کے بغیر، وہی مسئلہ دوبارہ ہو سکتا ہے، یا نئے مسائل پیدا ہو سکتے ہیں۔ باقاعدہ مینٹیننس سسٹم کو مستحکم رکھتی ہے اور مستقبل کے مسائل کو روکتی ہے۔ یہ کمپیوٹر کے استعمال میں بہتر کارکردگی، لمبی عمر، اور کم رکاوٹوں کو یقینی بناتی ہے۔

مثال: اگر کمپیوٹر سٹوریج مکمل بھر جانے کی وجہ سے سست ہو جاتا ہے، تو چند فائلوں کو ڈیلیٹ کرنے سے اس کی رفتار تھوڑی دیر کے لیے بہتر ہو سکتی ہے۔ تاہم، اگر غیر ضروری فائلیں باقاعدگی سے ہٹائی نہ جائیں یا سسٹم کو اپ ڈیٹ نہ کیا جائے، تو وہی مسئلہ دوبارہ ہو سکتا ہے۔

خراب مینٹیننس کا اثر

اگر کسی کمپیوٹر کو باقاعدگی سے صحیح نہیں رکھا جاتا ہے، تو اسے کئی مسائل کا سامنا کرنا پڑ سکتا ہے جو اس کی کارکردگی اور قابل اعتماد ہونے کو متاثر کرتے ہیں۔ مناسب مینٹیننس کی کمی کی وجہ سے عام طور پر پیش آنے والے مسائل درج ذیل ہیں:

- **سست کارکردگی(Slow Performance)** : عارضی فائلیں، غیر استعمال شدہ پروگرامز، اور پس منظر کے عوامل وقت کے ساتھ جمع ہو سکتے ہیں۔ اگر ان کا انتظام نہ کیا جائے، تو کمپیوٹر سست اور نان رسپونسو ہو سکتا ہے۔
- **سسٹم کی ناکامیاں(System Failures)** : اگر اہم سسٹم اپ ڈیٹس انسٹال نہ ہوں، تو کمپیوٹر صحیح طریقے سے کام کرنا بند کر سکتا ہے۔ یہ فریز یا کریش کر سکتا ہے، یا سٹارٹ ہونے میں ناکام ہو سکتا ہے۔
- **سیکورٹی خطرات(Security Risks)** : جب سسٹم یا ایپس کو اپ ڈیٹ نہیں کیا جاتا، تو نقصان دہ سافٹ ویئر جیسے وائرس آسانی سے کمپیوٹر میں داخل ہو سکتے ہیں۔
- **ڈیٹا کا نقصان(Data Loss)** : بیک اپ کے بغیر، کریشز (Crashes) یا ہارڈ ویئر کی خرابی کے دوران اہم فائلیں ضائع ہو سکتی ہیں۔
- **ہارڈ ویئر کو نقصان(Hardware Damage)** : اگر سسٹم کو صاف نہ کیا جائے یا مناسب طریقے سے ہوا دار نہ رکھا جائے۔ دھول اور زیادہ گرمی (overheating) ہارڈ ویئر کے اجزاء کو نقصان پہنچا سکتی ہے۔

سسٹم ہیلتھ مانیٹرنگ کا تعارف

سسٹم کی کارکردگی کو صحیح رکھنے کے لیے، اس کی کارکردگی کو باقاعدگی سے چیک کرنا ضروری ہے۔ اس عمل کو سسٹم ہیلتھ مانیٹرنگ کہا جاتا ہے۔ مائیکروسافٹ ونڈوز جیسے جدید آپریٹنگ سسٹمز میں بلٹ ان ٹولز شامل ہیں جو سسٹم کے افعال کی نگرانی میں مدد کرتے ہیں۔

یہ ٹولز آپ کو درج ذیل کاموں / امور کی اجازت دیتے ہیں:

- چیک کریں کہ کتنی میموری اور پروسیسنگ پاور استعمال ہو رہی ہے۔
- پس منظر کی ایپلیکیشنز اور سروسز کی نگرانی کریں۔
- سسٹم وارننگز اور ایرر میسجز دیکھیں۔
- ان پروگراموں کی شناخت کریں جو کمپیوٹر کو سست کر سکتے ہیں۔

باقاعدہ نگرانی چھوٹی خامیوں کو سنگین بننے سے پہلے پکڑنے میں مدد کرتی ہے۔ ان ٹولز کا استعمال کرتے ہوئے، صارفین کارکردگی کو بہتر بنانے اور مستقبل کے مسائل سے بچنے کے لیے بروقت اقدامات کر سکتے ہیں۔

2.2 بلٹ ان تشخیصی ٹولز کا استعمال

بنیادی ٹربل شوٹنگ کے اقدامات کے علاوہ، جدید آپریٹنگ سسٹمز بلٹ ان ٹولز فراہم کرتے ہیں جو صارفین کو سسٹم کی کارکردگی کی مزید تفصیل سے نگرانی کرنے میں مدد کرتے ہیں۔ یہ ٹولز کمپیوٹر کی موجودہ حالت کے بارے میں مفید معلومات دیتے ہیں اور میموری، سٹوریج، پروسیسنگ پاور، اور سسٹم ایررز سے متعلق مسائل کی شناخت میں مدد کرتے ہیں۔

یہ سیکشن مائیکروسافٹ ونڈوز میں دستیاب تین اہم تشخیصی آلات کو متعارف کراتا ہے:

- ٹاسک مینیجر (Task Manager)
- ریسورس مانیٹر (Resource Monitor)
- ایونٹ ویور (Event Viewer)

ان میں سے ہر ایک ٹول سسٹم کی صحت اور کارکردگی سے متعلق مسائل کو حل کرنے میں مختلف کردار ادا کرتا ہے۔

ٹاسک مینیجر

ٹاسک مینیجر مائیکروسافٹ ونڈوز میں ایک بلٹ ان ٹول ہے جو صارفین کو اپنے کمپیوٹر کی کارکردگی کی نگرانی میں مدد کرتا ہے۔ یہ سسٹم کے پروسیسر (CPU)، میموری (RAM)، اور سٹوریج (Disk) کے استعمال کے بارے میں رئیل ٹائم کی معلومات دکھاتا ہے۔ ٹاسک مینیجر کا Processes ٹیب تمام فعال پروگراموں اور وہ کتنا سسٹم ریسورسز استعمال کر رہے ہیں دکھاتا ہے، جبکہ Performance ٹیب CPU، میموری، ڈسک، اور نیٹ ورک کے استعمال کے لیے تفصیلی گرافس دکھاتا ہے۔

ٹاسک مینیجر کو کیسے کھولیں؟

ٹاسک مینیجر کو کھولنے کے دو آسان طریقے ہیں:

i. کی بورڈ سے شارٹ کٹ کی Ctrl + Shift + Esc دبائیں۔

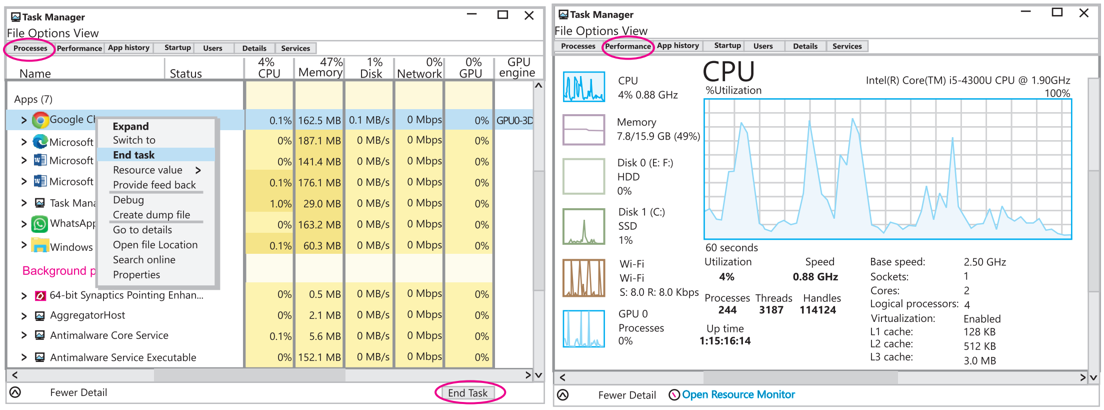
ii. یا ٹاسک بار پر دائیں کلک کریں اور مینو سے Task Manager منتخب کریں۔

آپ ٹاسک مینیجر سے کیا کر سکتے ہیں؟

ٹاسک مینیجر اس وقت مددگار ہوتا ہے جب آپ کا کمپیوٹر سست ہو، یا کوئی ایپلیکیشن رسپونڈ کرنا بند کر دے۔ یہ آپ کو درج ذیل کی اجازت دیتا ہے:

- دیکھیں کہ ہر پروگرام کتنی پروسیسنگ پاور اور میموری استعمال کر رہا ہے۔
- اگر کوئی پروگرام صحیح طریقے سے کام نہیں کر رہا ہے تو اسے بند کرنے کے لیے ٹاسک مینیجر کے Processes ٹیب سے ایک پروگرام کو منتخب کریں اور End Task پر کلک کریں۔ (شکل 2.1 دیکھیں)۔
- پس منظر میں خاموشی سے چلنے والے پروگراموں کو دیکھیں۔

ٹاسک مینیجر کا Performance ٹیب آپ کے کمپیوٹر کی کارکردگی کو ظاہر کرنے والے (real-time) گرافس اور ڈیٹا کو دکھاتا ہے (شکل 2.1 دیکھیں)۔



شکل 2.1: ونڈوز میں فعال پراسیسز اور سسٹم کی کارکردگی کو ظاہر کرتا ہوا ٹاسک مینیجر

ریسورس مانیٹر

ریسورس مانیٹر مائیکروسافٹ ونڈوز میں ایک ایڈوانسڈ سسٹم ٹول ہے جو آپ کے کمپیوٹر کے ریسورسز (میموری، C P U، اور ڈسک) کے استعمال کی تفصیل دیتا ہے، جیسا کہ شکل 2 . 2 میں دکھایا گیا ہے۔ اگرچہ ٹاسک مینیجر ایک فوری جائزہ دیتا ہے، ریسورس مانیٹر اس بارے میں زیادہ تفصیلی معلومات فراہم کرتا ہے کہ کون سے پروگرام اور پس منظر کی سروسز سسٹم ریسورسز استعمال کر رہی ہیں۔ یہ خاص طور پر مددگار ہے جب سسٹم سست چل رہا ہو، زیادہ گرم ہو رہا ہو، یا کسی واضح وجہ کے بغیر بہت زیادہ میموری استعمال کر رہا ہو۔

ریسورس مانیٹر کو کیسے کھولیں؟

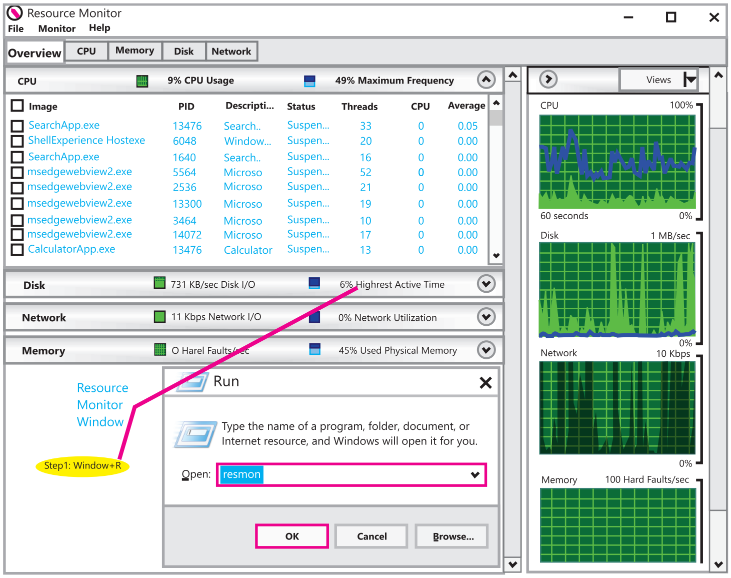
ریسورس مانیٹر کھولنے کے دو عام طریقے ہیں:

- ٹاسک مینیجر کھولیں، Performance ٹیب پر جائیں، اور نیچے Open Resource Monitor پر کلک کریں۔
- یا شارٹ کٹ کی (Windows + R) دبائیں، ظاہر ہونے والے ڈائیلاگ باکس میں کمانڈ resmon ٹائپ کریں، اور Enter دبائیں۔

ریسورس مانیٹر کی اہم خصوصیات

ریسورس مانیٹر کی کچھ اہم خصوصیات ذیل میں درج ہیں:

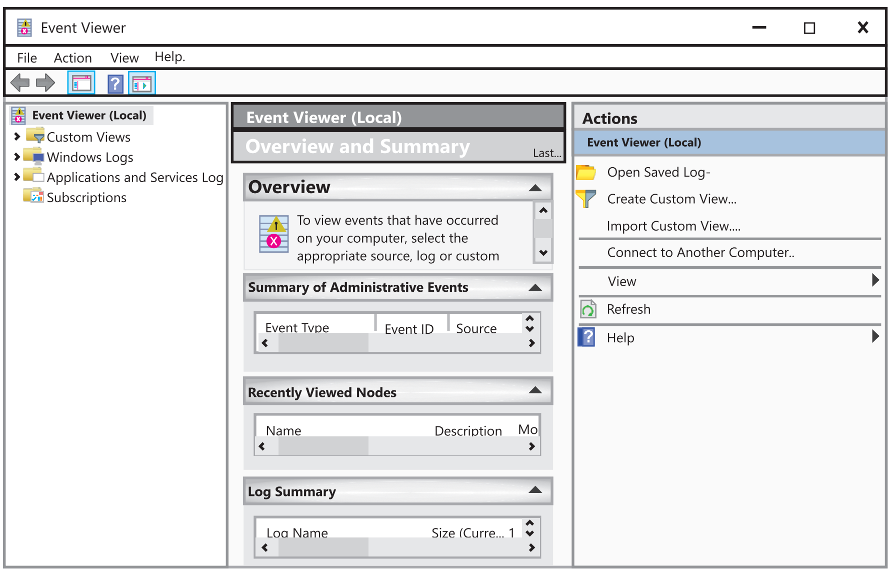
- **سی پی یو ٹیب:** CPU Tab دکھاتا ہے کہ کون سے پراسیسز اور سروسز پروسیسر استعمال کر رہی ہیں۔
- **میموری ٹیب:** Memory Tab ظاہر کرتا ہے کہ کتنی میموری استعمال شدہ، خالی، یا مخصوص ہے۔
- **ڈسک ٹیب:** Disk Tab نگرانی کرتا ہے کہ کون سے پروگرام ہارڈ ڈرائیو سے پڑھ رہے ہیں یا لکھ رہے ہیں۔
- **نیٹ ورک ٹیب:** Network Tab دکھاتا ہے کہ کون سی ایپلیکیشنز انٹرنیٹ یا نیٹ ورک پر ڈیٹا بھیج رہی ہیں یا وصول کر رہی ہیں۔



شکل 2.2: سسٹم ریسورسز کے استعمال کی نگرانی کے لیے ریسورس مانیٹر۔

ایونٹ ویوور (Event Viewer)

ایونٹ ویوور (Event Viewer) بھی ونڈوز کا ایک بلٹ ان ٹول ہے جو سسٹم کی سرگرمیوں کے تفصیلی لاگز (logs) کو ریکارڈ کرتا ہے۔ یہ صارفین کو سافٹ ویئر، ہارڈ ویئر، اور سیکورٹی سے متعلق ایررز (errors)، وارننگز (warnings)، اور اہم واقعات (important events) کو چیک کرنے میں مدد کرتا ہے، جیسا کہ شکل 2.3 میں دکھایا گیا ہے۔



شکل 2.3: سسٹم مینٹیننس سے متعلق آنے والے واقعات کو دیکھنے کے لیے ایونٹ ویوور

ایونٹ ویور کو کیسے کھولیں؟

شارٹ کٹ کی Windows + R دبائیں، ظاہر ہونے والے ڈائیلاگ باکس میں کمانڈ eventvwr ٹائپ کریں، اور Enter دبائیں۔

سرگرمی:

مقصد: ٹاسک مینیجر کا استعمال کرتے ہوئے ایک غیر ضروری پس منظر کے پروگرام کی شناخت کریں اور اسے بند کریں۔

اقدامات:

1. Ctrl + Shift + Esc کا استعمال کرتے ہوئے ٹاسک مینیجر کھولیں۔
2. Processes ٹیب پر جائیں۔
3. پس منظر میں چلنے والا کوئی ایسا پروگرام تلاش کریں جسے آپ استعمال نہیں کر رہے ہیں (مثلاً، میوزک ایپ، اپ ڈیٹ سروس)۔
4. اسے منتخب کریں اور End Task پر کلک کریں (صرف اس صورت میں جب آپ کو یقین ہو کہ اسے بند کرنا محفوظ ہے)۔
5. مشاہدہ کریں کہ آیا آپ کے کمپیوٹر کی رفتار میں بہتری آئی ہے۔

2.3 سسٹم ریکوری آپشنز

بعض صورتوں میں، کسی مسئلے کو ٹھیک کرنے کے لیے بنیادی ٹربل شوٹنگ کافی نہیں ہوتی۔ کمپیوٹر کریش ہو سکتا ہے، نان رسپونسو ہو سکتا ہے، یا سٹارٹ ہونے میں ناکام ہو سکتا ہے۔ ایسی صورت حال کو ہینڈل کرنے کے لیے، مائیکروسافٹ ونڈوز جیسے آپریٹنگ سسٹمز خصوصی ٹولز فراہم کرتے ہیں جنھیں سسٹم ریکوری آپشنز کہا جاتا ہے۔ یہ ریکوری آپشنز زیادہ تر صورتوں میں اہم ڈیٹا کو کھوئے بغیر کمپیوٹر کو کام کرنے والی حالت میں بحال کرنے میں مدد کرتے ہیں۔ اس سیکشن میں، ہم عام طور پر استعمال ہونے والے تین ریکوری ٹولز کے بارے میں جانیں گے۔

QUICK QUIZ

سوالات کے جوابات دیں:

- کیا ٹاسک کو ختم کرنے سے سسٹم کی کارکردگی میں بہتری آئی؟
- آپ کے خیال میں ایسا کیوں ہوا؟

سیف موڈ

سیف موڈ (Safe Mode) ونڈوز کو صرف سب سے بنیادی ڈرائیورز اور سروسز کے ساتھ سٹارٹ کرتا ہے۔ یہ پس منظر کے غیر ضروری پروگراموں اور سٹارٹ اپ سافٹ ویئر کو غیر فعال کر دیتا ہے، جس سے خراب ایپس یا سیٹنگز کی وجہ سے ہونے والے مسائل کو ٹھیک کرنا آسان ہو جاتا ہے۔

کب اور کیسے استعمال کریں؟

سیف موڈ اس وقت مفید ہے جب:

- سسٹم بار بار کریش یا ری سٹارٹ (Restart) ہو رہا ہو۔
- کوئی پروگرام یا ڈرائیور کمپیوٹر کو فریز کر رہا ہو۔

- آپ کو (Malware) کے لیے سکین کرنے یا حال ہی میں انسٹال کی گئی ایپ کو ان انسٹال کرنے کی ضرورت ہو۔

سیف موڈ کیسے کھولیں؟

زیادہ تر سسٹمز پر:

Settings>Update&Security>Recovery>Advanced Startup>Restart Now پر کلک کریں۔

پھر Troubleshoot>Advanced Options>Startup Settings>Restart منتخب کریں۔

جب سسٹم ری بوٹ ہو تو F4 دبائیں۔

بعض پرانے سسٹمز پر:

سسٹم ری سٹارٹ کریں اور ونڈوز کا لوگو ظاہر ہونے سے پہلے بار بار F8 دبائیں۔ Safe Mode کو منتخب کریں اور Enter دبائیں۔

سیف موڈ اور نیٹ ورکنگ

سیف موڈ کا یہ ورژن نیٹ ورک ڈرائیورز کو شامل کرتا ہے، لہٰذا آپ انٹرنیٹ تک رسائی حاصل کر سکتے ہیں۔ یہ اس وقت مددگار ہوتا ہے جب:

- آپ کو اپ ڈیٹس یا ڈرائیورز ڈاؤن لوڈ کرنے کی ضرورت ہو۔
- آپ آن لائن ٹولز استعمال کرنا چاہتے ہوں۔
- آپ کلاؤڈ پر مبنی اینٹی وائرس سکین چلانا چاہتے ہوں۔

اس میں داخل ہونے کے لیے، اوپر دیے گئے اقدامات پر عمل کریں اور F4 کے بجائے F5 دبائیں۔

SMART TIPS

سمارٹ ٹپس

سیف موڈ استعمال کرتے وقت، ان احتیاطی تدابیر پر عمل کریں:

- جب تک آپ ان کا مقصد نہ سمجھیں، سیٹنگز کو تبدیل نہ کریں۔
- سسٹم فائلوں کو ڈیلیٹ کرنے سے گریز کریں؛ وہ ونڈوز کے صحیح طریقے سے کام کرنے کے لیے اہم ہو سکتی ہیں۔
- مسائل کو حل کرنے کے لیے صرف بلٹ ان یا قابل اعتماد ٹولز کا استعمال کریں۔
- کام مکمل کرنے کے بعد اپنے کمپیوٹر کو دوبارہ سٹارٹ کریں۔
- اگر یقین نہ ہو تو اپنے استاد یا آئی ٹی ماہر سے مدد طلب کریں۔

سیف موڈ طاقتور ہے، لیکن اسے احتیاط سے استعمال کیا جانا چاہیے۔

سسٹم ری سٹور

سسٹم ری سٹور آپ کو ذاتی فائلوں کو ڈیلیٹ کیے بغیر اپنے کمپیوٹر کو پہلی حالت میں واپس لے جانے کی اجازت دیتا ہے۔ یہ ری سٹور پوائنٹس کا استعمال کرتے ہوئے کام کرتا ہے، جو بڑی تبدیلیوں سے پہلے محفوظ کیے گئے آپ کے سسٹم کے ورژن ہیں۔

ری سٹور پوائنٹس کیا ہیں؟

ایک ری سٹور پوائنٹ آپ کے سسٹم کی سیٹنگز اور فائلوں کے محفوظ کردہ سنیپ شاٹ کی طرح ہے۔ یہ آپ کے کمپیوٹر کو پہلے کی، کام کرنے والی حالت میں واپس لانے میں مدد کرتا ہے اگر کچھ غلط ہو جائے، جیسے کہ کوئی خراب پروگرام یا ڈرائیور انسٹال کرنے کے بعد۔ سسٹم اپ ڈیٹس یا ڈرائیور کی انسٹالیشن (Installation) سے پہلے ری سٹور پوائنٹس خود بخود بنائے جاتے ہیں، یا ضرورت پڑنے پر خود بنا تا ہے۔

ری سٹور پوائنٹ بنانے کے اقدامات

خود ری سٹور پوائنٹ بنانے کے لیے ان اقدامات پر عمل کریں:

- Start بٹن پر کلک کریں اور Control Panel کھولیں۔
- System and Security پر جائیں، پھر System پر کلک کریں۔
- بائیں طرف، System Protection پر کلک کریں۔
- نئی ونڈو میں، System Protection ٹیب کے نیچے، Create بٹن پر کلک کریں۔
- اپنے ری سٹور پوائنٹ کے لیے ایک نام ٹائپ کریں (مثال کے طور پر، "پرنٹر ڈرائیور انسٹال کرنے سے پہلے") اور Create پر کلک کریں۔
- تصدیق کے لیے چند سیکنڈ انتظار کریں۔ آپ کا ری سٹور پوائنٹ اب محفوظ ہو گیا ہے۔

ری سٹور پوائنٹ استعمال کرنے کے اقدامات

اگر آپ کا کمپیوٹر صحیح طریقے سے کام نہیں کر رہا ہے، تو پچھلی حالت میں واپس جانے کے لیے ان اقدامات پر عمل کریں:

- Control Panel کھولیں۔
- Recovery پر جائیں اور Open System Restore پر کلک کریں۔
- ایک ونڈو ظاہر ہو گی جو دستیاب ری سٹور پوائنٹس دکھائے گی۔
- سب سے حالیہ ری سٹور پوائنٹ کو منتخب کریں جب آپ کا سسٹم ٹھیک کام کر رہا تھا۔
- پھر Next اور پھر Finish پر کلک کریں۔
- آپ کا سسٹم ری سٹارٹ ہو گا اور منتخب کردہ سیٹنگز کو بحال کر دے گا۔

بوٹ ایبل ریکوری میڈیا

بعض اوقات، ایک کمپیوٹر مکمل طور پر کام کرنا بند کر سکتا ہے اور سٹارٹ ہونے میں ناکام ہو سکتا ہے۔ ایسی صورتوں میں، سیف موڈ اور

ری سٹور پوائنٹس جیسے عام ریکوری آپشنز کام نہیں کر سکتے۔ لہٰذا، اس صورت میں، ہم سسٹم کو بحال کرنے کے لیے بوٹ ایبل ریکوری میڈیا استعمال کرتے ہیں۔

بوٹ ایبل میڈیا اور اس کے استعمال کو سمجھنا

بوٹ ایبل ریکوری میڈیا ایک USB فلیش ڈرائیو یا DVD ہے جس میں سسٹم کی اہم فائلیں ہوتی ہیں۔ یہ کمپیوٹر کو بلٹ ان سٹوریج کے بجائے بیرونی ڈیوائس سے سٹارٹ کرنے میں مدد کرتا ہے۔ سسٹم سٹارٹ کرنے کے بعد، آپ بلٹ ان ریکوری ٹولز استعمال کر سکتے ہیں جیسے:

- Startup Repair ان مسائل کو ٹھیک کرنے کے لیے جو ونڈوز کو لوڈ ہونے سے روکتے ہیں۔
- System Restore ایک ری سٹور پوائنٹ کا استعمال کرتے ہوئے کام کرنے والی حالت میں واپس جانے کے لیے۔
- Command Prompt ایڈوانسڈ ٹربل شوٹنگ کے لیے۔
- System Reset or Reinstallation اگر ضرورت ہو تو آپریٹنگ سسٹم کو دوبارہ انسٹال کرنے کے لیے۔

ہم بوٹ ایبل میڈیا کب استعمال کرتے ہیں؟

ہم یہ میڈیا ان صورتوں میں استعمال کرتے ہیں جہاں سسٹم:

- سیف موڈ قابل رسائی نہ ہو۔
- آپ کو آپریٹنگ سسٹم کی مرمت یا دوبارہ انسٹال کرنے کی ضرورت ہو۔

بوٹ ایبل USB کیسے بنائیں اور استعمال کریں؟

ایک بوٹ ایبل USB بنانے کے لیے:

- اپنے آپریٹنگ سسٹم فراہم کنندہ کی آفیشل ویب سائٹ سے سسٹم امیج یا انسٹالر ڈاؤن لوڈ کریں (جیسے Windows ISO, Linux ISO یا macOS installer)۔
- ونڈوز کے لیے، Rufus (ایک مفت USB ٹول) کو اس کی آفیشل سائٹ (https://rufus.ie/en/) سے ڈاؤن لوڈ کریں۔ macOS کے لیے، آپ اس کے بلٹ ان ٹولز جیسے Terminal یا Disk Utility استعمال کر سکتے ہیں۔
- ایک USB ڈرائیو (16GB یا اس سے زیادہ) لگائیں۔
- Rufus کھولیں، USB اور ISO فائل کو منتخب کریں۔

ونڈوز یا لینکس سسٹمز کے لیے، جدید PCs کے لیے GPT (GUID Partition Table) اور پرانے سسٹمز کے لیے Master (Boot Record) MBR کا انتخاب کریں۔ پھر بوٹ ایبل USB بنانے کے لیے Start پر کلک کریں۔

کیا آپ جانتے ہیں؟

GPTاورMBRسٹوریج ڈرائیو (ہارڈ ڈرائیو یاSSD)پر ڈیٹا کو منظم کرنے کے دو مختلف طریقے ہیں۔
MBR (Master Boot Record)پرانا سسٹم ہے۔ یہ 2 ٹی بی (2 TB) تک کی ڈسک سائز اور زیادہ سے زیادہ 4 پرائمری پارٹیشنز کو سپورٹ کرتا ہے۔
GPTجدید معیار ہے۔ یہ 2 ٹی بی سے بڑی ڈسک کو سپورٹ کرتا ہے، 100 سے زیادہ پارٹیشنز کی اجازت دیتا ہے، اورUEFIپر مبنی سسٹمز کے لیے ضروری ہے۔
یہی وجہ ہے کہ نئے کمپیوٹرز اور آپریٹنگ سسٹمز ابMBRپرGPTکو ترجیح دیتے ہیں!

پارٹیشن سٹائل (MBRیاGPT) کیسے چیک کریں؟

پارٹیشن سٹائل چیک کرنے کے لیے، آپ کو درج ذیل کی ضرورت ہے:

i. Win + Xدبائیں اورDisk Managementکو منتخب کریں۔

ii. وہ ڈرائیو تلاش کریں جہاں ونڈوز انسٹال ہے (عام طور پر Disk 0).

iii. ڈسک لیبل پر رائٹ کلک کریں (مثلاً، Disk 0)، پھرPropertiesکا انتخاب کریں۔

iv. Volumesٹیب پر جائیں۔

Partition Styleدیکھیں:

یہ یا تو: GPT (GUID Partition Table) یاMBR (Master Boot Record)دکھائے گا۔

SMART TIPS

سمارٹ ٹپس

اگر آپ کا کمپیوٹرGPTاستعمال کرتا ہے، لیکن آپUSBکوMBRفارمیٹ میں بناتے ہیں، تو ہو سکتا ہے کہ یہ بالکل بوٹ (start) نہ ہو۔ یہی صورت حال تب بھی ہوتی ہے اگر آپ کا سسٹمMBRاستعمال کر رہا ہو اور آپ GPT USB استعمال کریں۔

سرگرمی:

مقصد: ونڈوز کے بلٹ ان ٹولز کا استعمال کرتے ہوئے ایک بوٹ ایبل ریکوری ڈرائیو (bootable recovery drive)بنانا سیکھیں۔

درکار سافٹ اور ہارڈ ویئر

- ایکWindows 10یا 11 کمپیوٹر۔
- ایک خالیUSBڈرائیو (کم از کم 16 GB)۔

اقدامات:

1. USBکو کمپیوٹر میں لگائیں۔
2. Windows keyدبائیں،Recovery Driveٹائپ کریں، اور اسے کھولیں۔
3. "Backup system files to the recovery drive"کو چیک کریں اورNextپر کلک کریں۔
اپنیUSBڈرائیو کو منتخب کریں اورNextپر کلک کریں (کلاس میں اجازت کے بغیر "Create" پر کلک نہ کریں)۔

2.4 BIOS/UEFI اور بوٹ پراسیس سے آگاہی

QUICK QUIZ

سوالات کے جوابات دیں:

- ونڈوز کے مطابق، یہ ریکوری ڈرائیو کس چیز میں آپ کی مدد کرے گی؟

جب آپ اپنے کمپیوٹر پر پاور بٹن دباتے ہیں، تو آپریٹنگ سسٹم (جیسے ونڈوز) سکرین پر ظاہر ہونے سے پہلے کئی پوشیدہ اقدامات ہوتے ہیں۔ ان ابتدائی اقدامات کو فرم ویئر (firmware) نامی خصوصی بلٹ ان سافٹ ویئر کے ذریعے کنٹرول کیا جاتا ہے۔

اس سیکشن میں، آپ سیکھیں گے کہ فرم ویئر کیا ہے، بیسک ان پٹ/آؤٹ پٹ سسٹم (BIOS) اور یونیفائیڈ ایکسٹینسیبل فرم ویئر انٹرفیس (UEFI) کس طرح کمپیوٹر کو سٹارٹ ہونے میں مدد کرتے ہیں، اور اگر ضرورت ہو تو آپ بنیادی سٹارٹ اپ سیٹنگز تک کیسے رسائی حاصل کرسکتے ہیں۔

کمپیوٹر سٹارٹ اپ میں BIOS/UEFI کا کردار

تمام کمپیوٹرز میں مدر بورڈ میں خصوصی سافٹ ویئر موجود ہوتا ہے جسے فرم ویئر کہا جاتا ہے۔ یہ ہارڈ ڈرائیو پر محفوظ نہیں ہوتا اور کمپیوٹر بند ہونے پر ڈیلیٹ نہیں ہوتا۔ فرم ویئر کمپیوٹر کو سٹارٹ کرنے کے بارے میں بنیادی ہدایات دیتا ہے۔

فرم ویئر کی دو عام اقسام ہیں:

- BIOS پرانا فرم ویئر سافٹ ویئر جو ایک بنیادی، صرف ٹیکسٹ مینیو دکھاتا ہے اور صرف کی بورڈ کے ساتھ کام کرتا ہے۔
- UEFI نئی قسم کا فرم ویئر سافٹ ویئر، جو ایک گرافیکل مینیو کو سپورٹ کرتا ہے جو ماؤس اور کی بورڈ دونوں کا استعمال کرتا ہے، اور بڑی ہارڈ ڈرائیوز کو سپورٹ کرتا ہے۔

کمپیوٹر سٹارٹ کرنے میں BIOS/UEFI کا کردار:

- چیک کرتا ہے کہ آیا ہارڈ ویئر صحیح طریقے سے کام کر رہا ہے (جیسے RAM، کی بورڈ، اور ہارڈ ڈرائیو)۔
- بوٹ ڈیوائس (USB، ہارڈ ڈسک، وغیرہ) کو تلاش کرتا ہے اور آپریٹنگ سسٹم لوڈ کرتا ہے۔
- صارفین کو سسٹم سیٹنگز جیسے تاریخ، وقت، اور بوٹ آرڈر کو تبدیل کرنے کی اجازت دیتا ہے۔

DO YOU KNOW

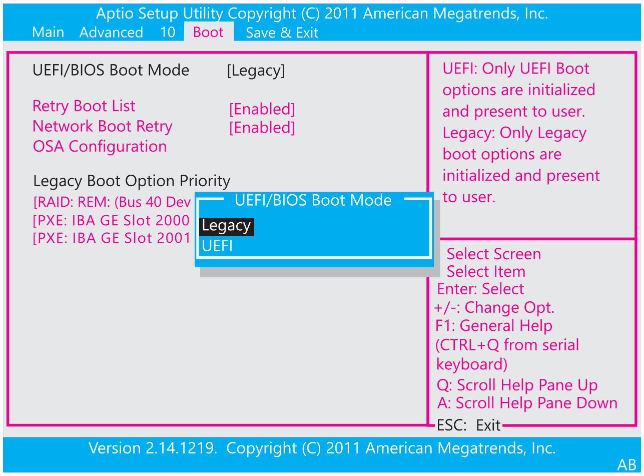
کیا آپ جانتے ہیں؟

BIOS سب سے پہلے 1980 کی دہائی میں تیار کیا گیا تھا، جبکہ UEFI اب زیادہ تر جدید کمپیوٹرز میں استعمال ہوتا ہے کیونکہ یہ تیز اور زیادہ محفوظ ہے۔

POST ٹیسٹ

POST کا مطلب ہے Power-On Self-Test ۔یہ پہلی چیز ہے جو فرم ویئر کمپیوٹر آن ہونے کے بعد کرتا ہے۔ POST سب سے اہم ہارڈویئر کو چیک کرتا ہے تاکہ یہ یقینی بنایا جاسکے کہ سب کچھ صحیح طریقے سے کام کر رہا ہے۔ اگر تمام پرزے ٹھیک ہیں، تو کمپیوٹر آپریٹنگ سسٹم کو لوڈ کرنا جاری رکھتا ہے۔ اگر کچھ غلط ہے، تو یہ بیپ (Beep) کر سکتا ہے یا سکرین پر ایک ایرر میسج دکھا سکتا ہے۔

مثال: اگر RAM غائب ہے، تو کمپیوٹر مسلسل بیپ (Beep) کرے گا۔



شکل 2.4: IOS/UEFI

BIOS تک رسائی اور ڈیفالٹس پر ری سیٹ کرنا

بعض اوقات آپ کو سیٹنگز کو تبدیل کرنے یا مسائل کو ٹھیک کرنے کے لیے BIOS/UEFI میں داخل ہونے کی ضرورت ہوتی ہے۔

BIOS/UEFI سیٹنگز کیسے کھولیں؟

اس سیٹنگ کو کھولنے کے لیے، آپ کو اپنا کمپیوٹر بند کرنا ہوگا۔

- اسے آن کریں اور فوری طور پر Del، F2، F10، یا Esc جیسی (Key) کی دبائیں (درست کی برانڈ پر منحصر ہے، جیسے HP، DELL، اور MSI)۔
- BIOS یا UEFI سکرین ظاہر ہو جائے گی۔

ڈیفالٹ سیٹنگز پر ری سیٹ کرنا:

اگر غلط سیٹنگز محفوظ ہو جائیں، تو آپ BIOS/UEFI سیٹنگز کے اندر "Load Setup Defaults" یا "Reset to Default" کو منتخب کر کے سسٹم کو اس کی اصل حالت میں بحال کر سکتے ہیں۔

بوٹ آرڈر اور بوٹ ریپیر

بوٹ آرڈر سے مراد وہ ترتیب ہے جس میں کمپیوٹر آپریٹنگ سسٹم کو لوڈ کرنے کے لیے ایک ڈیوائس تلاش کرتا ہے۔ مثال کے طور پر، یہ پہلے USB ڈرائیو، پھر ہارڈ ڈرائیو کو آزما سکتا ہے۔

بوٹ آرڈر کیسے تبدیل کریں؟

- BIOS/UEFI سیٹنگز میں داخل ہوں۔
- Boot ٹیب یا مینو پر جائیں۔

- اپنی مطلوبہ ڈیوائس (مثلاً USB) کو فہرست میں سب سے اوپر منتقل کریں۔
- تبدیلیاں محفوظ کریں اور باہر نکلیں۔

بعض اوقات کمپیوٹر گمشدہ فائلوں یا ایررز کی وجہ سے سٹارٹ نہیں ہو پاتا۔ اسے ونڈوز ریکوری USB یا DVD سے Startup Repair کا استعمال کرتے ہوئے ٹھیک کیا جا سکتا ہے۔

2.5 ڈیٹا ریکوری کی تکنیکس

بعض اوقات، ہم غلطی سے فائلیں ڈیلیٹ کر دیتے ہیں یا سسٹم کے مسائل کی وجہ سے انہیں کھو دیتے ہیں۔ زیادہ تر صورتوں میں، بلٹ ان ٹولز یا خصوصی تھرڈ پارٹی سافٹ ویئر کی مدد سے کھوئی ہوئی فائلوں کو بحال کرنا ممکن ہے۔ یہ سیکشن فائل ریکوری کے بنیادی طریقوں اور ڈیٹا کے مستقل نقصان سے بچنے کے طریقے بیان کرتا ہے۔

ڈیلیٹ شدہ فائلوں کو بحال کرنے کے دو اہم طریقے ہیں:

i. بلٹ ان ٹولز کا استعمال۔

ii. تھرڈ پارٹی سافٹ ویئر کا استعمال۔

بلٹ اِن ٹولز

یہ ٹولز پہلے ہی آپریٹنگ سسٹم کا حصہ ہوتے ہیں، اسی لیے ہم انہیں بلٹ ان ٹولز کہتے ہیں۔ اس سیکشن میں، ہم ڈیلیٹ شدہ فائلوں کو بحال کرنے کے لیے دستیاب کچھ بلٹ ان ٹولز پر بات کریں گے۔

ری سائیکل بن

جب ونڈوز میں کوئی فائل ڈیلیٹ ہوتی ہے، تو یہ عام طور پر ری سائیکل بن میں چلی جاتی ہے۔ یہ وہاں رہتی ہے جب تک کہ بن خالی نہ ہو جائے۔ آپ ری سائیکل بن کو کھول سکتے ہیں، فائل پر دائیں کلک کر سکتے ہیں، اور اسے اس کی اصل جگہ پر واپس لانے کے لیے Restore کو منتخب کر سکتے ہیں۔

DO YOU KNOW — کیا آپ جانتے ہیں؟

> ہم ہر ڈرائیو کے لیے ری سائیکل بن کی (Key) کی پراپرٹیز کو استعمال کرتے ہوئے اس کا سائز مختص کر سکتے ہیں۔ اس کے آئیکون پر رائیٹ کلک کرتے ہوئے۔

فائل ہسٹری

فائل ہسٹری ونڈوز میں ایک بیک اپ فیچر ہے۔ اگر یہ آن ہو تو، سسٹم باقاعدہ وقفوں پر آپ کی فائلوں کی کاپیاں محفوظ رکھتا ہے۔ آپ پہلے کے ورژنز سے فائلوں کو بحال کر سکتے ہیں، یہاں تک کہ اگر وہ ڈیلیٹ یا تبدیل ہو چکی ہوں۔ اسے آن کرنے کے لیے Setting > Update & Security > Backup پر جائیں، ایک بیرونی USB یا ہارڈ ڈرائیو منسلک کریں، اور "Add a drive" کو منتخب کریں۔ ایک بار فعال ہونے پر، ونڈوز باقاعدگی سے آپ کی فائلوں جیسے Documents, Pictures, اور Desktop کا اس ڈرائیو پر بیک اپ لے گا۔

سابقہ ورژن (Previous Versions)

یہ فیچر آپ کو کسی فائل یا فولڈر کے سابقہ ورژن کو بحال کرنے کی اجازت دیتا ہے۔ فائل یا فولڈر پر رائیٹ کلک کریں، Properties کا انتخاب کریں، پھر Previous Versions ٹیب پر جائیں۔ اگر دستیاب ہو، تو آپ ایک پرانی کاپی کو منتخب اور Restore کر سکتے ہیں۔

تھرڈ پارٹی ٹولز

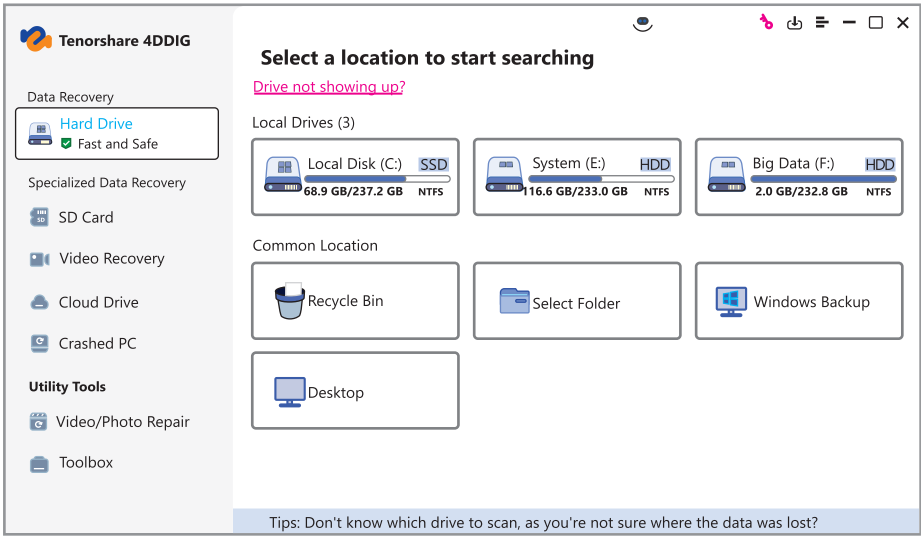
بعض اوقات، ڈیلیٹ شدہ فائلیں ری سائیکل بن یا سابقہ ورژن میں نہیں ملتیں۔ ایسی صورتوں میں، آپ تھرڈ پارٹی ریکوری سافٹ ویئر استعمال کرنے کی کوشش کر سکتے ہیں۔ ٹینور شیئر (Tenorshare) ایک مفت اور استعمال میں آسان فائل ریکوری پروگرام ہے جسے شکل 2.5 میں دکھایا گیا ہے۔ یہ آپ کی ہارڈ ڈرائیو، U S B یا میموری کارڈ کو سکین کر سکتا ہے تاکہ حال ہی میں ڈیلیٹ کی گئی فائلوں کو تلاش کر سکے۔ پھر آپ منتخب کر سکتے ہیں کہ کن فائلوں کو ریکور کرنا ہے۔

ٹینور شیئر کیسے استعمال کریں؟

ٹینور شیئر استعمال کرنے کے اقدامات:

- ٹینور شیئر کھولیں، وہ لوکیشن منتخب کریں جہاں سے آپ فائل ری سٹور کرنا چاہتے ہیں، جیسے کہ ایک مخصوص ڈرائیو یا فولڈر (شکل 2.5 دیکھیں)۔
- ری سٹور کرنے کے لیے فائل کی قسم منتخب کریں (مثلاً، تصاویر، دستاویزات)۔
- کھوئی ہوئی فائلوں کو تلاش کرنا شروع کرنے کے لیے "Scan" پر کلک کریں۔
- وہ فائلیں منتخب کریں جنہیں آپ بحال کرنا چاہتے ہیں اور انہیں بحال کرنے کے لیے "Recover" پر کلک کریں۔

ریکوری کی کامیابی کا انحصار اس بات پر ہے کہ اسے کتنی دیر پہلے ڈیلیٹ کیا گیا تھا اور آیا نئے ڈیٹا نے اسے اوور رائٹ (overwrite) تو نہیں کر دیا۔



شکل 2.5: ٹینور شیئر (Tenorshare) ڈیٹا ریکوری ٹول۔

ریکوری کی حدود (Limitations of recovery)

اگرچہ ڈیٹا ریکوری ٹولز مددگار ہو سکتے ہیں، لیکن ان کی درج ذیل حدود ہیں:

- اگر کوئی فائل کافی عرصہ پہلے ڈیلیٹ ہو گئی ہو، تو ہو سکتا ہے کہ اب وہ قابل بازیابی نہ ہو۔
- اگر ہارڈ ڈرائیو کو نقصان پہنچا ہو یا فارمیٹ کر دی گئی ہو تو ریکوری ہمیشہ ممکن نہیں ہوتی۔
- فائلیں مستقل طور پر ضائع ہو سکتی ہیں اگر ری سائیکل بن خالی کر دی گئی ہو، اور کوئی بیک اپ نہ بنایا گیا ہو۔

SMART TIPS

سمارٹ ٹپس

- ریکوری سافٹ ویئر کو کبھی بھی اسی ڈرائیو پر انسٹال نہ کریں جہاں آپ کی کھوئی ہوئی فائل محفوظ تھی۔ اس سے کامیاب ریکوری کا امکان کم ہو سکتا ہے۔
- ایڈوانسڈ ریکوری ٹولز کو آزمانے سے پہلے ہمیشہ ری سائیکل بن کو چیک کریں۔ یہ ڈیلیٹ شدہ فائلوں کو بحال کرنے کا سب سے آسان اور محفوظ طریقہ ہے۔

سرگرمی:

مقصد : طلباء کو ٹینور شیئر (Tenorshare) کا استعمال کرتے ہوئے ایک ٹیسٹ فائل پر فائل ریکوری کے طریقہ کار کو سمجھنے میں مدد کرنا۔

اقدامات:

1. Desktop پر "TestFile.txt" کے نام سے ایک ٹیکسٹ فائل بنائیں۔
2. اسے ڈیلیٹ کر دیں اور ری سائیکل بن (Recycle Bin) کو خالی کر دیں۔
3. ٹینور شیئر کھولیں۔ Select All Files منتخب کریں۔ لوکیشن کے طور پر Desktop کا انتخاب کریں۔
4. Scan پر کلک کریں۔ جب فائل ظاہر ہو جائے تو اسے منتخب کریں اور Recover پر کلک کریں۔
5. اسے کسی اور فولڈر میں محفوظ کریں (مثلاً، Documents)۔

سوالات کے جوابات دیں:

- کیا آپ فائل کو بحال کرنے میں کامیاب ہوئے؟
- ڈیلیٹ کرنے کے بعد آپ کو اسی ڈرائیو پر نئی فائلیں محفوظ کرنے سے کیوں گریز کرنا چاہیے؟

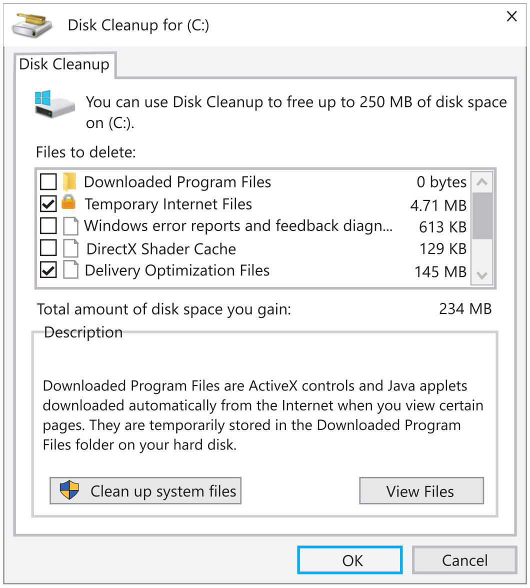
2.6 حفاظتی مینٹیننس کے بہترین طریقے

بالکل اسی طرح جیسے ہم اپنی صحت کا خیال رکھتے ہیں یا اپنی گاڑیوں کی دیکھ بھال کرتے ہیں، کمپیوٹر کو بھی صحیح طریقے سے کام جاری رکھنے کے لیے باقاعدہ توجہ کی ضرورت ہوتی ہے۔ حفاظتی مینٹیننس کا مطلب ہے سادہ کام کرنا جو مستقبل کے بڑے مسائل سے بچنے میں مدد کرتے ہیں۔ یہ کام آپ کے کمپیوٹر کی رفتار کو بہتر بنا سکتے ہیں، اسے محفوظ رکھ سکتے ہیں، اور اسے زیادہ دیر تک چلنے میں مدد دے سکتے ہیں۔

ڈسک کلین اپ

کچھ وقت کے بعد، کمپیوٹر غیر ضروری فائلیں جمع کر لیتا ہے جیسے کہ انٹرنیٹ کی عارضی فائلیں، سسٹم کیشے، ونڈوز اپ ڈیٹ کی فائلیں، اور ری سائیکل بن میں موجود آئٹمز۔ یہ فائلیں جگہ لیتی ہیں اور سسٹم کی کارکردگی کو سست کر سکتی ہیں۔ ڈسک کلین اپ ونڈوز کا ایک بلٹ ان ٹول ہے جو ایسی فائلوں کو محفوظ طریقے سے ختم کرنے میں مدد کرتا ہے۔

ڈسک کلین اپ کرنے کے اقدامات



شکل 2.6: ڈسک کلین اپ مینیو

- ونڈوز سرچ بار میں Disk Cleanup ٹائپ کریں اور فہرست سے ٹول کھولیں۔ یہ ایک چھوٹی ونڈو کھولتا ہے جس میں آپ سے کلین کرنے کے لیے ایک ڈرائیو کا انتخاب کرنے کو کہا جاتا ہے، جیسا کہ شکل 2.6 میں دکھایا گیا ہے۔
- وہ ڈرائیو منتخب کریں جسے آپ کلین کرنا چاہتے ہیں (عام طور پر :C) اور OK پر کلک کریں۔
- OK کے بعد ظاہر ہونے والے ڈائیلاگ باکس سے، ان فائلوں کی اقسام کے لیے باکسز کو چیک کریں جنہیں آپ ڈیلیٹ کرنا چاہتے ہیں، جیسا کہ شکل 2.6 میں دکھایا گیا ہے۔
- منتخب فائلوں کو ختم کرنے کے لیے OK پر کلک کریں۔

پیچ مینجمنٹ (Patch Management)

سافٹ ویئر اپ ڈیٹس، جنہیں پیچز (patches) بھی کہا جاتا ہے، ڈویلپرز کی طرف سے باقاعدگی سے جاری کیے جاتے ہیں تاکہ bugs ٹھیک کیے جا سکیں، نئی خصوصیات شامل کی جا سکیں، اور سیکورٹی کو بہتر بنایا جا سکے۔ اپنے سسٹم کو اپ ڈیٹ رکھنا اسے وائرس اور سیکورٹی خطرات سے بچانے میں مدد کرتا ہے۔

دو قسم کی اپ ڈیٹس ہیں:

i آپریٹنگ سسٹم اپ ڈیٹس (ونڈوز، macOS، وغیرہ کے لیے)۔

ii ایپلیکیشن اپ ڈیٹس (براؤزرز، میڈیا پلیئرز، وغیرہ جیسے پروگراموں کے لیے)۔

ونڈوز میں اپ ڈیٹس چیک کرنے کے لیے:

- Setting > Update & Security کھولیں۔
- Check for updates پر کلک کریں۔
- اگر دستیاب ہوں تو اپ ڈیٹس انسٹال کریں۔

ونڈوز ڈیفنڈر کے ساتھ محفوظ رہنا

ونڈوز ڈیفنڈر (Windows Defender) ونڈوز میں بلٹ ان سیکورٹی ٹول ہے جو آپ کے کمپیوٹر کو وائرس، مالویئر، اور دیگر خطرات سے بچاتا ہے۔ یہ دو قسم کے سکینز پیش کرتا ہے۔

- Quick Scan عام مسائل والے ڈرائیوز / جگہوں کو چیک کرنے کے لیے۔
- Full Scan پورے سسٹم کو اچھی طرح چیک کرنے کے لیے۔

ونڈوز ڈیفنڈر ریئل ٹائم پروٹیکشن (real-time protection) بھی فراہم کرتا ہے، جس کا مطلب ہے کہ یہ پس منظر میں آپ کے سسٹم کی نگرانی کرتا رہتا ہے اور اگر کچھ مشکوک ہوتا ہے تو آپ کو الرٹ جاری کرتا ہے۔

مینٹیننس کیلنڈر رکھنا

بنیادی مینٹیننس کے کاموں کے لیے ایک باقاعدہ شیڈول مقرر کرنا ایک اچھا خیال ہے۔ ایک مینٹیننس کیلنڈر آپ کو یاد رکھنے میں مدد کرتا ہے کہ اپنے سسٹم کو کب صاف کرنا ہے، وائرس کے لیے سکین کرنا ہے، یا اپ ڈیٹس کو چیک کرنا ہے۔ آپ اسے لکھ سکتے ہیں یا اپنے فون یا کمپیوٹر پر یاد دہانیاں (reminders) سیٹ کر سکتے ہیں۔

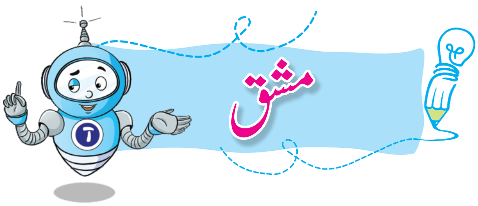
2.7 سسٹم کی دستاویزات اور لاگز

مناسب ریکارڈ رکھنا کمپیوٹر سسٹم کی مینٹیننس کا ایک اہم حصہ ہے۔ جب ہم جو کچھ کرتے ہیں اسے دستاویز کرتے ہیں، جیسے کہ سافٹ ویئر انسٹال کرنا، مسائل کو ٹھیک کرنا، یا اپ ڈیٹس کرنا، جس سے مستقبل میں سسٹمز کو ٹربل شوٹ کرنا آسان ہو جاتا ہے۔

خلاصہ (Summary)

- ٹربل شوٹنگ کے اقدامات، بشمول سسٹم کو ری اسٹارٹ کرنا، فزیکل کنیکشن (physical connections) کی جانچ پڑتال کرنا، اور سافٹ ویئر یا ہارڈ ویئر سے متعلق سادہ مسائل کو حل کرنا تاکہ کمپیوٹر کے عام مسائل کو ٹھیک کیا جا سکے۔
- جدید آپریٹنگ سسٹمز، جیسے مائیکروسافٹ ونڈوز، میں ایسے بلٹ ان (built-in) ٹولز شامل ہوتے ہیں جو سسٹم کی سرگرمیوں کی نگرانی کرنے میں مدد دیتے ہیں۔
- ٹاسک مینیجر مائیکروسافٹ ونڈوز میں ایک بلٹ ان ٹول ہے جو صارفین کو ان کے کمپیوٹر کی کارکردگی کی نگرانی کرنے میں مدد دیتا ہے۔
- ریسورس مانیٹر مائیکروسافٹ ونڈوز میں ایک جدید سسٹم ٹول ہے جو اس بات کا گہرا جائزہ فراہم کرتا ہے کہ آپ کے کمپیوٹر کے وسائل (میموری، سی پی یو، اور ڈسک) کیسے استعمال ہو رہے ہیں۔

- ایونٹ ویور (Event Viewer) بھی ونڈوز کا ایک بلٹ ان ٹول ہے جو سسٹم کی سرگرمیوں کے تفصیلی لاگز (logs) ریکارڈ کرتا ہے۔ یہ صارفین کو سافٹ ویئر، ہارڈ ویئر اور سیکیورٹی سے متعلق ایرز، وارننگز اور اہم واقعات (Events) کو چیک کرنے میں مدد دیتا ہے۔
- سسٹم ریسٹور آپ کو ذاتی فائلوں کو حذف کیے بغیر اپنے کمپیوٹر کو پچھلی حالت میں واپس لے جانے کی اجازت دیتا ہے۔
- بوٹ ایبل ریکوری میڈیا (Bootable recovery media) ایک USB فلیش ڈرائیو یا DVD ہے جس میں سسٹم کی اہم فائلیں ہوتی ہیں۔
- POST سے مراد 'پاور آن سیلف ٹیسٹ' (Power-On Self-Test) ہے۔ کمپیوٹر آن ہونے کے بعد فرم ویئر (firmware) سب سے پہلے یہی کام انجام دیتا ہے۔



کثیر انتخابی سوالات

i ٹاسک مینیجر بنیادی طور پر استعمال ہوتا ہے:

(الف) چلنے والی ایپلیکیشنز کی نگرانی
(ب) ڈسک کی صفائی
(ج) وائرس کے لیے سکین
(د) فائلوں کا انتظام

ii سیف موڈ ونڈوز کو کس طرح سٹارٹ کرتا ہے؟

(الف) تمام ڈرائیورز اور سروسز
(ب) صرف بنیادی ڈرائیورز
(ج) نیٹ ورک سروسز
(د) اینٹی وائرس ٹولز

iii بوٹ ایبل USB ڈرائیوز کس لیے استعمال ہوتی ہیں؟

(الف) گیمز انسٹال کرنا
(ب) سسٹم کی مرمت اور ریکوری
(ج) ویڈیوز دیکھنا
(د) ای میل بھیجنا

iv. UEFI سسٹمز میں، ایک بوٹ ایبل USB کو کس سٹائل میں بنایا جانا چاہیے؟

الف) NTFS فارمیٹ
ب) MBR سٹائل
ج) GPT سٹائل
د) ای میل بھیجنا

v. پاور-آن سیلف-ٹیسٹ (POST) کب کیا جاتا ہے؟

الف) سسٹم بندش کے دوران
ب) لاگ ان کے بعد
ج) سسٹم سٹارٹ اپ پر
د) سافٹ ویئر انسٹال کرتے وقت

vi. ٹینور شیئر (Tenorshare) ایک ٹول ہے جو کس کے لیے استعمال ہوتا ہے؟

الف) ویڈیوز چلانا
ب) کھوئی ہوئی فائلیں بحال کرنا
ج) چارٹ بنانا
د) CPU نگرانی

vii. فائل ہسٹری کس میں مدد کرتی ہے؟

الف) ویب سائٹ ہسٹری کو ٹریک کرنا
ب) ذاتی فائلوں کی کاپیاں محفوظ کرنا
ج) ایپس انسٹال کرنا
د) مالویئر کے لیے سکین

vii. ونڈوز ڈیفنڈر (Windows Defender) کیا فراہم کرتا ہے؟

الف) فائل کمپریشن
ب) وائرس سے تحفظ
ج) ڈسک فارمیٹنگ
د) موسیقی چلانا

مختصر سوالات

1. ریسورس مانیٹر کا ایک اہم فنکشن لکھیں۔
2. سسٹم ریکوری میں سیف موڈ کے استعمال کی تعریف کریں۔
3. روفوس (Rufus) کا استعمال کرتے ہوئے بوٹ ایبل USB بنانے کے اقدامات درج کریں۔
4. پاور-آن سیلف-ٹیسٹ (POST) کے دوران کیا ہوتا ہے؟
5. USB ڈرائیو سے ریکوری ٹولز استعمال کرتے وقت بوٹ آرڈر (Boot Order) کی سیٹنگ کیوں اہم ہے؟

6. باقاعدگی سے ڈسک کلین اپ چلانے کے دو فائدے درج کریں۔
7. ونڈوز ڈیفنڈر (Windows Defender) سسٹم کی صحت کو برقرار رکھنے میں کیسے مدد کرتا ہے؟

تفصیلی سوالات

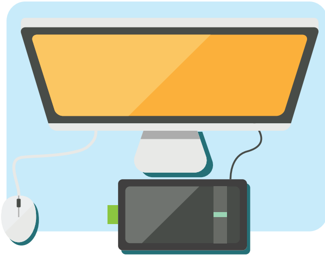
1. سسٹم مینٹیننس میں ٹاسک مینیجر اور ایونٹ ویور (Event Viewer) کے کردار کی وضاحت کریں۔ ہر ٹول کے لیے ایک مثال دیں جہاں یہ کسی مسئلے کو حل کرنے میں مدد کر سکتا ہے۔
2. آپ کا کمپیوٹر سسٹم کسی مسئلے کی وجہ سے صحیح طریقے سے سٹارٹ نہیں ہو رہا ہے۔ بیان کریں کہ ذاتی فائلوں کو ڈیلیٹ کیے بغیر مسئلے کو ٹھیک کرنے کے لیے سسٹم ری سٹور یا ایک بوٹ ایبل ریکوری USB کو کیسے استعمال کیا جا سکتا ہے۔
3. کوئی سی دو حفاظتی مینٹیننس (Preventive Maintenance) تکنیکوں کی مختصر وضاحت کریں جو کمپیوٹر کو آسانی سے چلانے میں مدد کرتی ہیں۔
4. وضاحت کریں کہ ڈیلیٹ شدہ فائلوں کو تھرڈ پارٹی سافٹ ویئر کا استعمال کرتے ہوئے کیسے بحال کیا جا سکتا ہے۔ نیز، بحث کریں کہ ریکوری ہمیشہ کیوں کامیاب نہیں ہوتی۔
5. ایک ہم جماعت ونڈوز کی ایک بڑی اپ ڈیٹ کے بعد ڈسک کلین کرنا چاہتا ہے۔ وضاحت کریں کہ ڈسک کلین اپ (Disk Cleanup) کا استعمال کیسے کیا جائے، بشمول سسٹم فائلوں کو محفوظ طریقے سے ریمو (Remove) کرنے کے۔

کثیر انتخابی سوالات کے جوابات

1. الف — چلنے والی ایپلیکیشنز کی نگرانی
2. ب — صرف بنیادی ڈرائیورز
3. ب — سسٹم کی مرمت اور ریکوری
4. ج — GPT سٹائل
5. ج — سسٹم سٹارٹ اپ پر
6. ب — کھوئی ہوئی فائلیں بحال کرنا
7. ب — ذاتی فائلوں کی کاپیاں محفوظ کرنا
8. ب — وائرس سے تحفظ

یونٹ

3 تعارف برائے پائتھن پروگرامنگ



حاصلاتِ تعلم: اس باب کے اختتام تک، آپ اس قابل ہو جائیں گے کہ:

- بنیادی پروگرامنگ کے تصوّرات کو سمجھ سکیں اور پائتھن ڈویلپمنٹ انوائرنمنٹ سیٹ اپ کر سکیں۔
- بنیادی پائتھن سینٹیکس اور ساخت لکھ اور سمجھ سکیں، جس میں ویری ایبلز، ڈیٹا ٹائپس، اور ان پٹ / آؤٹ پٹ آپریشنز شامل ہیں۔
- پائتھن میں مختلف آپریٹرز اور ایکسپریشنز، بشمول حسابی(arithmetic)، موازنہ(comparison)، اسائنمنٹ(assignment)، منطقی(logical) آپریٹرز، اور آپریٹرز کی ترجیح کا استعمال کر سکیں۔

تعارف

پائتھن پروگرامنگ کی دنیا میں خوش آمدید!

پائتھن اپنی سادگی اور پڑھنے میں آسانی کی وجہ سے ایک مقبول لینگوئج ہے، جو اسے نئے یوزرز (beginners) اور پیشہ ور افراد دونوں کے لیے مثالی بناتی ہے۔ اس باب میں، ہم بنیادی باتوں سے شروعات کریں گے، جن میں ڈیویلپمنٹ ماحول سیٹ اپ کرنا اور بنیادی پروگرامنگ تصوّرات سیکھنا شامل ہیں۔ اس باب کے اختتام تک، آپ پائتھن کی ایک جامع سمجھ حاصل کر چکے ہوں گے، جو آپ کو اپنے پروگرام لکھنے، جانچنے اور غلطیاں درست (debug) کرنے کے قابل بنائے گی۔

3.1: پائتھن پروگرامنگ کا تعارف

پائتھن ایک وسیع پیمانے پر استعمال ہونے والی ہائی لیول پروگرامنگ لینگوئج ہے جو اپنی سادگی اور پڑھنے میں آسانی کے لیے مشہور ہے۔ یہ ایک ہمہ گیر (versatile) لینگوئج ہے اور مختلف شعبوں میں لاگو ہوتی ہے، جن میں ویب ڈویلپمنٹ، ڈیٹا کا تجزیہ (data analysis)، اور مصنوعی ذہانت وغیرہ شامل ہیں۔ چاہے آپ ایک سادہ سکرپٹ بنا رہے ہوں یا کوئی پیچیدہ سافٹ ویئر تیار کر رہے ہوں، پائتھن کی صارف دوست فطرت آپ کو تیزی اور مؤثر طریقے سے آغاز کرنے میں مدد کرتی ہے۔

بنیادی پروگرامنگ کے تصوّرات

کمپیوٹر پروگرامنگ ہدایات کا ایک ایسا سیٹ بنانے کا عمل ہے جو کمپیوٹر کو بتاتا ہے کہ کوئی کام کیسے کرنا ہے۔ یہ ہدایات ایک ایسی پروگرامنگ لینگوئج میں لکھی جاتی ہیں جسے کمپیوٹر سمجھ سکتا ہے اور اس پر عمل (execute) کر سکتا ہے۔ کمپیوٹر پروگرامنگ کو ایسے سمجھیں جیسے آپ اپنے دوست کو اپنے گھر پہنچنے کی ہدایات دے رہے ہوں۔ آپ کو واضح اور درست ہونا چاہیے تاکہ آپ کا دوست گم

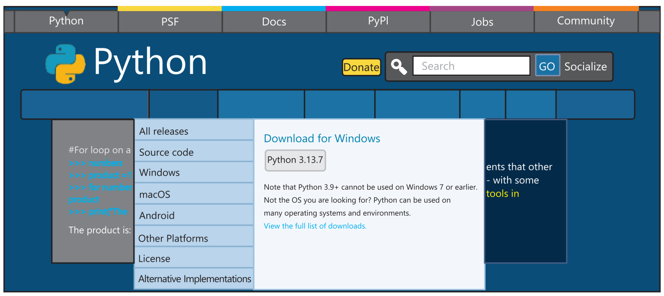
نہ ہو جائے۔ اسی طرح، جب ہم پروگرام لکھتے ہیں، تو ہم مخصوص کاموں کو مکمل کرنے کے لیے کمپیوٹر کو واضح اور درست ہدایات دیتے ہیں۔

پائتھن ڈویلپمنٹ انوائرنمنٹ کا سیٹ اپ

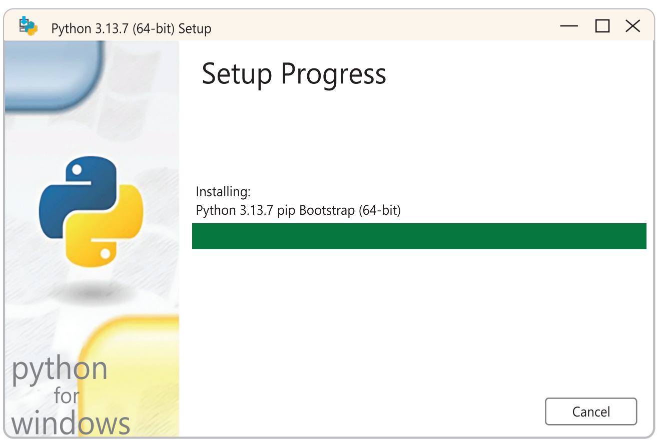
ڈویلپمنٹ انوائرنمنٹ سے مراد کمپیوٹر پائتھن کوڈ کو مؤثر طریقے سے لکھنے، اس پر عمل کرنے / عملدرآمد کرنے اور غلطیاں درست (debug) کرنے کے لیے تیار کرنے کا عمل ہے۔ اس میں ضروری سافٹ ویئر، ٹولز، اور لائبریریز کو انسٹال اور ترتیب دینا شامل ہے جو ڈیولپمنٹ کے عمل کو زیادہ ہموار اور مؤثر بناتے ہیں۔

نوٹ:

جب پائتھن انسٹال کریں، تو باکس کو ضرور چیک کریں جس پر لکھا ہو "Add Python to PATH" (پائتھن کو PATH میں شامل کریں)۔ یہ کمانڈ لائن (command line) سے پائتھن پر عمل درآمد آسان بنا دیتا ہے۔ ہم پائتھن پروگرام لکھنے اور اس پر عملدرآمد کرنے کے لیے آن لائن سروسز بھی استعمال کر سکتے ہیں۔



شکل 3.1: پائتھن کا ہوم پیج



شکل 3.2: سیٹ اپ پراسس

پائتھن انسٹالیشن کے مراحل

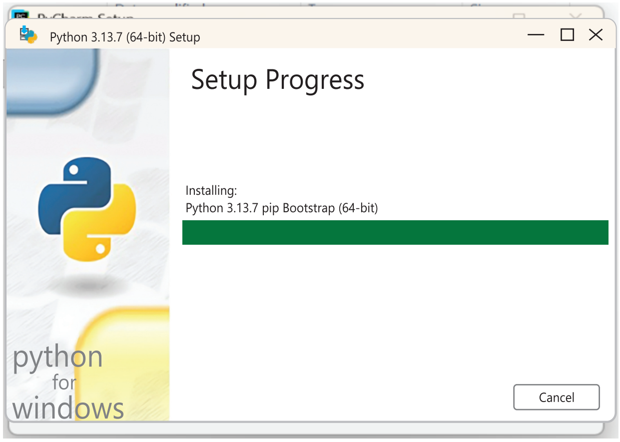
پائتھن کی انسٹالیشن (installation) کو مکمل کرنے کے لیے، درج ذیل اقدامات پر عمل کریں:

- پائتھن ڈاؤن لوڈ کریں (Download Python)
- ویب سائٹ وزٹ کریں: https://www.python.org/۔
- ڈاؤن لوڈز سیکشن: "Downloads" (ڈاؤن لوڈز) کے سیکشن پر جائیں۔
- تازہ ترین ورژن: اپنے آپریٹنگ سسٹم (Windows، macOS، یا Linux) کے لیے پائتھن کے تازہ ترین ورژن (مثلاً، Python 3.13.x) پر کلک کریں۔

انسٹالر خود بخود ڈاؤن لوڈ ہو جائے گا۔

-2 پائتھن انسٹال کریں (Install Python)

- ڈاؤن لوڈ کی گئی .exe فائل کو چلائیں۔
- انسٹالیشن کے دوران "Add Python to PATH" (پائتھن کو PATH میں شامل کریں) کے باکس پر ضرور چیک لگائیں۔
- "Install Now" کو منتخب کریں یا اگر ضرورت ہو تو انسٹالیشن کو اپنی مرضی کے مطابق (customize) بنائیں (مثلاً، انسٹالیشن کی ڈائریکٹری کا انتخاب کریں، انسٹالیشن مکمل ہونے کا انتظار کریں، پھر "Close" پر کلک کریں۔



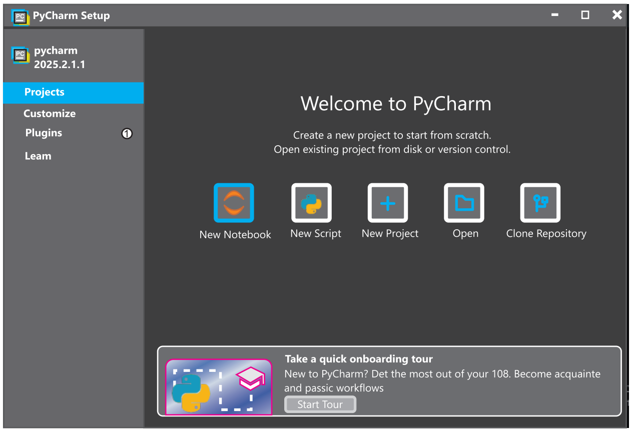
شکل 3.3: انسٹالیشن پراسس

-3 IDE سیٹ اپ کرنا

- آپ PyCharm، VS Code، یا IDLE (جو پائتھن کے ساتھ شامل ہوتا ہے) جیسے مشہور IDE میں سے کوئی بھی ڈاؤن لوڈ اور انسٹال کریں۔
- PyCharm پائتھن ڈویلپمنٹ کے لیے ایک طاقتور IDE ہے، جو ایک مفت کمیونٹی ایڈیشن یا ایک پیڈ پروفیشنل ایڈیشن میں دستیاب ہے۔
- PyCharm کو ڈاؤن لوڈ کرنے کے لیے https://www.jetbrains.com/pycharm
- پر جائیں۔ آپریٹنگ سسٹم (Windows، macOS، یا Linux) کے لیے مناسب ورژن ڈاؤن لوڈ کریں۔

-۴ IDE کو پائتھن کے ساتھ کنفگر کرنا

- PyCharm کو کھولیں اور ایک نیا پراجیکٹ شروع کریں۔
- "Settings" یا "Configure" (سیٹنگز) مینیو میں، "Project Interpreter" (پراجیکٹ انٹرپریٹر) پر جائیں۔



شکل 3.4: پائیتھن IDE

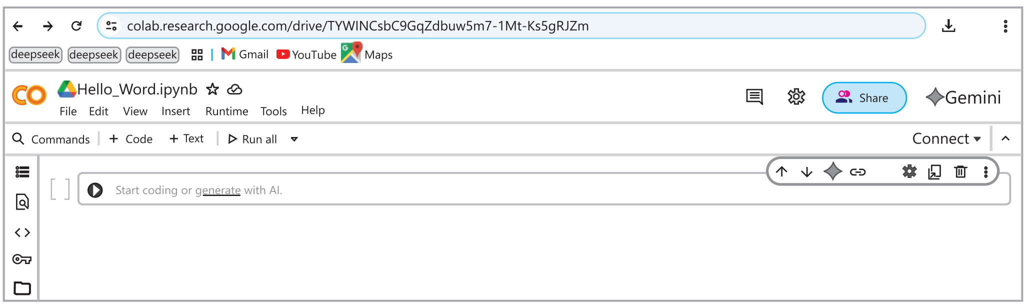
- گیئر کے آئیکن (Gear Icon) پر کلک کریں، "Add Interpreter" (انٹرپریٹر شامل کریں) کو منتخب کریں، اور اپنے پائتھن ایگزیکیوٹیبل (executiable) کا پاتھ منتخب کریں (مثلاً، Windows پر C:\Python312\python.exe یا Linux/macOS پر /usr/bin/python3)۔
- اگر یہ خودکار طریقے سے نہیں ملتا، تو اپنے انسٹالیشن فولڈر کو براؤز میں تلاش کریں۔
- سیٹنگز کو اپلائی کریں اور PyCharm کو انٹرپریٹر کو انڈیکس کرنے دیں۔

5- کنفیگریشن کی ٹیسٹنگ (Testing the Configuration)

- اپنے IDE میں ایک نئی فائل (مثال کے طور پر test.py) بنائیں۔
- ایک سادہ اسکرپٹ لکھیں: print("Hello, Python!")
- اس اسکرپٹ کو چلائیں (IDLE/PyCharm میں "Run" بٹن استعمال کریں یا VS Code میں ٹاسک سیٹ اپ کرنے کے بعد F5 دبائیں)۔
- اگر آؤٹ پٹ مثال کے طور پر "Hello, Python!" ظاہر ہوتا ہے، تو آپ کا IDE درست طریقے سے کنفگر ہو گیا ہے۔

گوگل کولیب (Google Colab)

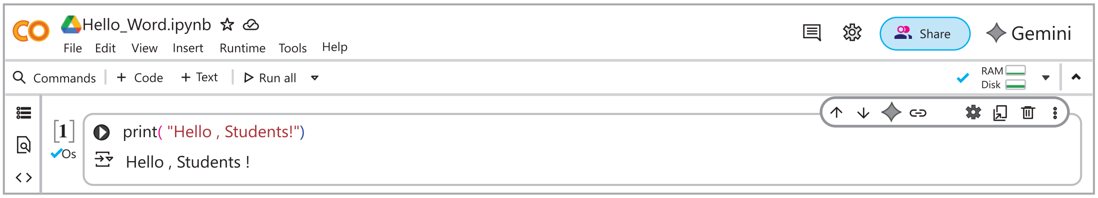
- پائتھن استعمال کرنے کا ایک اور بہترین آپشن Google Colab ہے۔
- یہ ایک مفت کلاؤڈ بیسڈ پلیٹ فارم ہے جو Jupyter Notebook کا ماحول فراہم کرتا ہے جس میں پائتھن اور مشہور لائبریریاں جیسے NumPy، Pandas، اور Matplotlib پہلے سے انسٹال ہوتی ہیں۔
- آغاز کرنے کے لیے، https://colab.research.google.com پر جائیں، گوگل اکاؤنٹ سے سائن ان کریں، اور ایک نئی نوٹ بک بنائیں۔
- اس کے لیے لوکل انسٹالیشن کی ضرورت نہیں ہے، جو اسے ابتدائی یا محدود وسائل والے کمپیوٹرز پر کام کرنے والوں کے لیے مثالی بناتا ہے۔
- آپ سیلز (cells) میں پائتھن کوڈ لکھ سکتے ہیں اور چلا سکتے ہیں۔



شکل 3.5: گوگل کولیب IDE

3.2 بنیادی پائتھن سینٹیکس اور سٹرکچر

درج ذیل پائتھن پروگرام لینگوئج کی سادگی اور پڑھنے میں آسانی کو ظاہر کرتا ہے۔



اس مثال میں، print فنکشن کو " " علامات کے اندر موجود پیغام کو آؤٹ پٹ کرنے کے لیے استعمال کیا گیا ہے۔ یہ پائتھن کے سادہ سینٹیکس کی وضاحت کرتا ہے، جہاں print جیسے فنکشنز کو ٹیکسٹ ظاہر کرنے جیسے کام کرنے کے لیے استعمال کیا جاتا ہے۔

پائتھن میں کمنٹس

- کمنٹس وہ لائنیں ہوتی ہیں جو پائتھن انٹرپریٹر کے ذریعے چلائی (execute) نہیں جاتیں۔
- ان کا مقصد کوڈ کے بارے میں پروگرامر کو وضاحت یا نوٹس فراہم کرنا ہوتا ہے۔
- سنگل لائن کمنٹ لکھنے کے لیے # کا نشان استعمال کیا جاتا ہے۔

```
# This is a single - line comment
print ( "K2 is the second-highest mountain in the world " )
'''
This is a multi-line comment.
It can span multiple lines.
'''
print("Edhi Foundation operates the world's largest volunteer
ambulance network.")
Output:
K2 is the second-highest mountain in the world
Edhi Foundation operates the world's largest volunteer ambulance
network.
```

کیا آپ جانتے ہیں؟

ہم "........" کو ملٹی لائن کمینٹ کے طور پر استعمال کرسکتے ہیں لیکن درحقیقت یہ کمنٹس نہیں ہیں۔ یہ ملٹی لائن سٹرنگ لیٹرز ہیں جن کو رن ٹائم پر نظر انداز کر دیا جاتا ہے۔

پائتھن میں ویری ایبلز (Variables in Python)

پروگرامنگ میں ویری ایبل ایک عارضی کنٹینر کی طرح کام کرتا ہے جو کمپیوٹر کی میموری میں ڈیٹا محفوظ کرتا ہے۔
ویری ایبل کے ذریعے ہم کوئی ویلیو محفوظ کر سکتے ہیں، جسے بعد میں پروگرام میں دوبارہ استعمال یا تبدیل کیا جا سکتا ہے۔
نیچے دی گئی مثال میں age ایک ویری ایبل ہے جو نومیرک ڈیٹا محفوظ کرنے کے لیے استعمال ہوا ہے۔
Iqbal lived for 60 years # ویری ایبل کی ویلیو پروگرام کے چلنے کے دوران بدل سکتی ہے، اسی لیے اسے "Variable" کہا جاتا ہے۔

```
age = 71
print ("Azam lived for", age, "years")
age =60
print ("Iqbal lived for", age, "years")
# Output
# Azam lived for 71 years
# Iqbal lived for 60 years
```

یہ مثال واضح کرتی ہے کہ کس طرح age ویری ایبل کو مختلف ویلیوز تفویض کی جاتی ہیں تاکہ اعظم (Azam) اور اقبال (Iqbal) کی عمروں کی نمائندگی کی جا سکے، جنہیں بعد میں آؤٹ پٹ کے حصے کے طور پر پرنٹ کیا جاتا ہے:

پائتھن میں ویری ایبل کے نام رکھنے کے اصول

ویری ایبل کے نام واضح، بامعنی اور درست (valid) ہونے چاہئیں تاکہ آپ کا کوڈ آسانی سے سمجھا جا سکے۔ پائتھن میں ویری ایبل کے نام رکھتے وقت درج ذیل قواعد پر سختی سے عمل کرنا ضروری ہے:

ویری ایبل کے نام رکھنے کے قوانین:

- ابتدائی حرف: نام کا آغاز کسی حرف (A-Z ،a-z) یا ایک انڈر سکور (_) سے ہونا چاہیے۔ یہ عددی ہندسے (digits) سے شروع نہیں ہو سکتا۔
- دیگر حروف: پہلے حرف کے بعد، نام میں حروف، ہندسے (0-9)، یا انڈر سکور (_) شامل ہو سکتے ہیں۔
- ویری ایبل کے نام کیس سینسیٹو ہوتے ہیں، یعنی age اور Age کو دو مختلف ویری ایبلز سمجھا جائے گا۔
- پائتھون کے ریزروڈ، جیسے کہ for، while، if وغیرہ، کو ویری ایبل کے نام کے طور پر استعمال نہیں کیا جا سکتا۔

کیا آپ جانتے ہیں؟

پائتھون میں مخصوص الفاظ (Reserved Keywords) وہ الفاظ ہوتے ہیں جن کا پائتھن کے اندر مخصوص اور پہلے سے طے شدہ معنی ہوتا ہے۔ انہیں کسی بھی صورت میں ویری ایبل، فنکشن، یا کسی دوسرے شناختی (Identifier) نام کے طور پر استعمال نہیں کیا جا سکتا۔

پائتھن کے مخصوص الفاظ کی فہرست

درج ذیل پائتھن کے کچھ مخصوص الفاظ ہیں:

print.if, else, elif, for, while, break, continue, and, or, not, in, is, True, False, None

ہم انہیں ویری ایبل کے نام کے طور پر استعمال نہیں کر سکتے۔

ویری ایبل کی اقسام

پائتھن میں، آپ مختلف قسم کا ڈیٹا محفوظ کرنے کے لیے مختلف اقسام کے ویری ایبلز بنا سکتے ہیں۔ پائتھن خود بخود طے کر لیتا ہے کہ ویری ایبل میں رکھی قیمتوں کی بنیاد پر اس کی قسم کیا ہو گی۔

درج ذیل ویری ایبل کی کچھ عام اقسام ہیں:

قسم	وضاحت	مثال
انٹیجر (Integer)/int	یہ مکمل اعداد (whole numbers) کو محفوظ کرتا ہے۔	age = 17
فلوٹنگ پوائنٹ /(float)	یہ اعشاریہ والے اعداد (decimal numbers) کو محفوظ کرتا ہے۔	price = 19.99
سٹرنگ (String)/str	یہ متن (text) کو محفوظ کرتا ہے، جسے اقتباس کے نشانات (quotes) میں لکھا جاتا ہے۔	name = "Ali”
بُولین / (bool)	یہ صرف دو قدروں (درست /True یا غلط /False) کو محفوظ کرتا ہے۔	is_student = True

پائتھن میں ان پٹ اور آؤٹ پٹ آپریشنز

اِن پُٹ اور آوٹ پُٹ آپریشنز آپ کو صارف سے بات چیت کرنے کی اجازت دیتے ہیں۔ آپ صارف کو اِن پُٹ دینے کا کہہ سکتے ہیں (اِن پُٹ) اور صارف کو انفارمیشن دیکھا سکتے ہیں (آوٹ پُٹ)۔

ان پٹ فنکشن

ہم صارف سے ان پٹ لینے کے لیے input() فنکشن کا استعمال کرتے ہیں۔
Input() کنکشن سے لی گئی اِن پٹ ہمیشہ سٹرنگ (String) ڈیٹا ٹائپ میں سے ڈیٹا لے گی
input() فنکشن سکرین پر ایک پیغام ظاہر کرتا ہے اور صارف کے کچھ ٹائپ کرنے اور Enter Key دبانے کا انتظار کرتا ہے۔
صارف کی طرف سے داخل کیا گیا ٹیکسٹ پھر ایک متغیر میں محفوظ ہو جاتا ہے۔
مثال کے طور پر
کوڈ کی پہلی لائن صارف سے اُن کا نام داخل کرنے کے لیے کہتی ہے اور اسے name نامی ویری ایبل میں محفوظ کرتی ہے۔

```
name = input("Enter your name: ")
print (" Hello, " + name + "!")
# Output
# Enter your name: Maryam Ashraf
 # Hello, Maryam Ashraf!
```

آؤٹ پٹ آپریشنز

آؤٹ پٹ آپریشنز
اسکرین پر معلومات ظاہر کرنے کے لیے print() فنکشن استعمال کیا جاتا ہے۔print() فنکشن ایک یا ایک سے زیادہ آرگیومنٹ لیتا ہے اور انہیں دکھاتا ہے۔
مثال: کوڈ کی دوسری لائن ایک پیغام دکھاتی ہے جس میں صارف کا نام شامل ہوتا ہے۔

کیا آپ جانتے ہیں؟

کیا آپ جانتے ہیں؟
1. Print() فنکشن کے ذریعے ایک ہی وقت میں زیادہ ویلیو کی آؤٹ پٹ لینے کے لیے اُن کے درمیان ',' استعمال کرتے ہیں۔
2. پائتھن میں اِنڈینٹیشن آپ کے کوڈ کا سکوپ (scop) یا ویری ایبلز (structure) کی ساخت بناتی ہے۔ غلط اِنڈینٹیشن آپ کے کوڈ میں ایرر کا سبب بن سکتی ہے۔

انٹیجر اور فلوٹ ان پٹ کو ہینڈل کرنا:

نومیرک اِن پٹ کو ہینڈل کرنے کے لیے، عام طور پر int() فنکشن یا float() فنکشن استعمال کیا جاتا ہے۔ یہ فنکشنز اِن پٹ سٹرنگز کو بالترتیب انٹیجرز (مکمل اعداد) یا فلوٹنگ پوائنٹ نمبرز (اعشاریہ والے اعداد) میں تبدیل کرتے ہیں۔
مثال: انٹیجر ان پٹس

```
# Example : Handling integer input
  user_age = int(input("Enter your age: "))
  print("Your age is:",user_age)
# Output
# Enter your age: 16
# Your age is : 16
```

فلوٹ اِن پٹس

```
# Example: Handling float input
user_height = float(input("Enter your height in meters: ") )
print("Your height is", user_height,"meters")
# Output
# Enter your height in meters: 1.5
# Your height is 1.5 meters
```

3.3 آپریٹرز اور ایکسپریشنز

آپریٹرز ایسی علامتیں ہیں جو متغیرات اور مستقل قدروں پر آپریشنز انجام دیتی ہیں۔ ایک ایکسپریشن متغیرات، آپریٹرز، اور ویلیوز کا ایک مجموعہ ہے جو ایک نتیجہ پیدا کرتا ہے۔ آیئے پائتھن میں مختلف قسم کے آپریٹرز کو دیکھتے ہیں۔

حسابی آپریٹرز (Arithmetic Operators)

حسابی آپریٹرز بنیادی ریاضیاتی عوامل انجام دینے کے لیے استعمال ہوتے ہیں جیسے کہ جمع (addition)، نفی (subtraction)، ضرب (multiplication)، تقسیم (division)، ماڈیولس (modulus)، ایکسپونینشیل (exponentional)، اور فلور تقسیم (floor division) جیسا کہ مندرجہ ذیل کوڈ میں دکھایا گیا ہے۔

```
# Define variables
a= 10
b= 3
# Perform all arithmetic operations on these numeric
variables and print results
print(a, "+", b, "=", a + b) # Output: 10+3=13
print(a, "*", b, " = " , a * b) # Output: 10 * 3 = 30
print(a, "/", b, " = ", a / b)
# Output: 10 / 3 = 3.3333333333333333
print(a, "//", b, " = ", a // b) #floor division
# Output: 10 // 3=3
print(a, "%", b, " = ", a % b)
# Output: 10 % 3 = 1
print(a, "**", b, " = ", a ** b)
# ** represent power operator:
# Output: 10**3= 1000
```

تقابلی آپریٹرز (Comparison Operators)

تقابلی آپریٹرز دو قیمتوں یا ایکسپریشنز کا موازنہ کرنے کے لیے استعمال ہوتے ہیں۔ یہ ان کے درمیان ریلیشنل لاجک کا تعین کرتے ہیں، جیسے کہ مساوات (equality)، عدم مساوات (inequality)، زیادہ ہونا (greater than)، کم ہونا (less than) یہ آپریٹرز

موازنہ کے نتیجے کی بنیاد پر ایک بولیئن ویلیو (یعنی True یا False) دیتے ہیں۔ یہاں ایک پائتھن پروگرام ہے جو تمام تقابلی آپریٹرز کے استعمال کو ظاہر کرتا ہے۔

```
Define variables
X = 10
Y = 5
# Greater than
print (x, “>”, y, “=”, x > y) # Output: 10 > 5 = True
# Less than
print (x, “<”, y, “=”, x < y) # Output: 10 < 5 = False
```

```
#Equal to
print (x, “==”, y, “=”, x == y) # Output: 10 == 5 = False
# Not equal to
print (x, “!=”, y, “=”, x != y) # Output: 10 != 5 = True
# Greater than or equal to
print (x, “>=”, y, “=”, x >= y) # Output: 10 >= 5 = True
# Less than or equal to
print (x, “<=”, y, “=”, x <= y) # Output: 10 <= 5 = False
```

مذکورہ بالا کوڈ پائتھن کے تقابلی آپریٹرز کو ظاہر کرتا ہے، جس میں دو متغیرات x اور y کا موازنہ مختلف آپریٹرز جیسے >, <, ==, !=, >=, اور <= کا استعمال کرتے ہوئے کیا گیا ہے۔

اسائنمنٹ آپریٹرز

اسائنمنٹ آپریٹرز متغیرات (variables) کو ویلیوز (values) تفویض کرنے کے لیے استعمال ہوتے ہیں۔ سب سے عام اسائنمنٹ آپریٹر مساوات کی علامت (=) ہے، جو دائیں طرف کی قدر کو بائیں طرف کے متغیر کو تفویض (assign) کرتی ہے۔ کمپاؤنڈ اسائنمنٹ آپریٹرز بھی ہیں جیسے +=، -=، *=، اور /= جو حسابی عمل کو اسائنمنٹ کے ساتھ جوڑتے ہیں۔

```
# Define initial values
a = 10
b = 5

# Assignment
x = a; print("a =" , x)# Output: a = 10
# Addition assignment
a += b; print("a after addition =" , a)# Output: a after addition = 15
# Subtraction assignment
a -= b; print("a after subtraction =" , a)# Output: a after subtraction= 10
# Multiplication assignment
a *= b; print("a after multiplication =" , a)# Output: a after multiplication = 50
# Division assignment
a /= b; print("a after division =" , a)# Output: a after division = 10.0
#Floor division assignment
a //= b; print("a after floor division  =" , a)# Output: a after floor division = 2
# Modulus assignment
a %= b; print("a after modulus = " , a)# Output: a after modulus = 2.0
# Exponentiation assignment
a **= b; print("a after exponentiation =", a)# Output: a after exponentiation = 32.0
```

مذکورہ بالا کوڈ پائتھن میں کمپاؤنڈ اسائنمنٹ آپریٹرز کے استعمال کی وضاحت کرتا ہے۔ یہ ایک متغیر "a" کو ایک ویلیو تفویض کرنے سے شروع ہوتا ہے۔ اس کے بعد یہ مختلف حسابی آپریشنز کو ظاہر کرتا ہے جن میں جمع، نفی، ضرب، تقسیم، فلور تقسیم، ماڈیولس، اور ایکسپونینشیل شامل ہیں۔ ہر آپریشن "a" کی ویلیو کو اپ ڈیٹ کرتا ہے، اور نتائج اپ ڈیٹ شدہ ویلیو کے ساتھ پرنٹ کیے جاتے ہیں۔ کوڈ کمنٹس ہر آپریشن کے آؤٹ پٹ کو دکھاتے ہیں۔

منطقی آپریٹرز (Logical Operators)

منطقی آپریٹرز ایک پروگرام میں متعدد شرائط یا ایکسپریشنز کو جوڑنے کے لیے استعمال ہوتے ہیں۔ سب سے عام منطقی آپریٹرز "and، or اور not" ہیں۔ یہ منطقی عمل انجام دینے کے لیے استعمال ہوتے ہیں اور شامل ایکسپریشنز کے جائزہ (evaluation) کی بنیاد پر بولینئن قیمتیں واپس کرتے ہیں۔

```
# Define variables
x = True
y = False
# Logical AND
logical_and = x and  y
 print(x , "and ", y ,"=", logical_and) # Output: True and False =False
# Logical OR
logical_or = x or y
print(x , "or " , y , " =", logical_or) # Output: True or False = True
# Logical NOT
```

```
logical_not_x = not x
print("not", x, " = " , logical_not_x) # Output not True= False
logical_not_y = not y
print("not", y, "=" , logical_not_y)
# Output:  not False=  True
```

مذکورہ بالا کوڈ پائتھن میں منطقی آپریٹرز کے استعمال کو ظاہر کرتا ہے۔ یہ دو بولینئن متغیرات x اور y کی تعریف کرتا ہے اور ان پر "and، or اور not" جیسے منطقی عمل انجام دیتا ہے۔ ان منطقی عمل کے نتائج پرنٹ کیے جاتے ہیں، جو یہ دکھاتے ہیں کہ آیا ایکسپریشنز True ہیں یا False۔ کوڈ میں ہر منطقی عمل کے نتیجے کے ساتھ کمنٹس بھی شامل ہیں۔

پائتھن میں آپریٹر کی ترجیح

آپریٹر کی ترجیح (Operator Precedence) اس ترتیب کا تعین کرتی ہے جس میں ایک ایکسپریشن میں آپریشنز انجام پاتے ہیں۔ پائتھن کے ساتھ ساتھ ریاضی میں بھی، کچھ آپریٹرز زیادہ ترجیح رکھتے ہیں اور دوسروں سے پہلے ان کا جائزہ لیا جاتا ہے۔ اس کو سمجھنا یہ یقینی بنانے میں مدد کرتا ہے کہ آپ کے حسابی عمل درست طریقے سے کیے گئے ہیں۔

قوسین '()': سب سے زیادہ فوقیت۔ قوسین کے اندر کے عمل سب سے پہلے کیے جاتے ہیں۔ مثال کے طور پر، 4*(2+3) کا نتیجہ 20 آتا ہے۔

- قوت نما (Exponentiation): اس کے بعد پاور آپریشنز انجام دیے جاتے ہیں۔ مثال کے طور پر، 2+2*2**3=18
- ضرب (*)، تقسیم (/)، اور ماڈیولس (%): یہ عمل اگلے نمبر پر آتے ہیں۔
- جمع (+) اور نفی (-): ان کی فوقیت ضرب اور تقسیم کے مقابلے میں کم ہوتی ہے۔
- مثال کے طور پر: 4*3+10/2-11%3
- پہلے ضرب، تقسیم، اور ماڈیولس کے عمل کریں:
- 4*3 کا نتیجہ 12 آتا ہے۔
- 10/2 کا نتیجہ 5.0 آتا ہے۔
- 11%3 کا نتیجہ 2 آتا ہے۔
- پھر جمع اور نفی کے عمل انجام دیں: 12 + 5.0 – 2 = 15.0
- مثال: آپریٹر کی ترجیح (Operator precedence) کو ظاہر کرنے کے لیے درج ذیل کوڈ پر غور کریں:

```
print("===  PYTHON  OPERATOR  PRECEDENCE  DEMONSTRATION ===\n")
# 1. Arithmetic Operators
print("1. ARITHMETIC OPERATORS:")
print(f"3 + 2 * 5    = {3 + 2 * 5}")   # Multiplication before addition
print(f"(3 + 2) * 5 = {(3 + 2) * 5}")# Parentheses change order
print(f"2 ** 3 * 4   = {2 ** 3 * 4}") # Exponentiation before multiplication
print(f"15 / 3 * 2   = {15 / 3 * 2}") # Division and multiplication (left to right)
print(f"15 // 4 + 2 = {15 // 4 + 2}")# Floor division before addition
print(f"11 // 4 + 2 = {11 // 4 + 2}")# Floor division before addition
print(f"17 % 5 * 2   = {17 % 5 * 2}") # Modulo before multiplication
# Output
# === PYTHON OPERATOR PRECEDENCE DEMONSTRATION ===
```

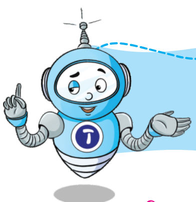
```
# 1. ARITHMETIC OPERATORS:

# 3 + 2 * 5      = 13
# (3 + 2) * 5    = 25
# 2 ** 3 * 4     = 32
# 15 / 3 * 2     = 10.0
# 15 // 4 + 2    = 5
# 11 // 4 + 2    = 4
# 17 % 5 * 2     = 4
```

کیا آپ جانتے ہیں؟

پائتھن میں، سٹرنگ سے پہلے "f" کا استعمال سٹرنگ انٹرپولیشن (string interpolation) سٹرنگ کے اندر متغیرات یا ایکسپریشنز کو شامل کرنے کے لیے کیا جاتا ہے۔

خلاصہ (Summary)

- پائتھن ایک اعلیٰ سطحی (ہائی لیول)(high-level)، کثیر المقاصد (versatile) پروگرامنگ لینگوئج ہے۔ یہ اپنی سادگی اور پڑھنے میں آسانی کے لیے جانی جاتی ہے۔
- پائتھن ویب ڈویلپمنٹ، ڈیٹا اینالیسس اور مصنوعی ذہانت (AI) وغیرہ میں استعمال ہوتی ہے۔
- پروگرامنگ میں کمپیوٹرز کو درست ہدایات دینا شامل ہے۔ پروگرام کمپیوٹرز کو بتاتے ہیں کہ مخصوص کام کیسے انجام دیتے ہیں۔
- پائتھن کو python.org سے ڈاؤن لوڈ کریں (تازہ ترین ورژن)۔
- انسٹالیشن کے مراحل: ڈاؤن لوڈ کریں انسٹالر چلائیں "Add Python to PATH" کو چیک (کلک) کریں۔
- IDE آپشنز: PyCharm، VS Code، IDLE۔
- آن لائن آپشن: Google Colab (کلاؤڈ پر مبنی، انسٹالیشن کی ضرورت نہیں)۔
- حسابی آپریٹرز (Arithmetic): +,-,*,/,//,%,**
- تقابلی آپریٹرز (Comparison): >=,<=,!=,==,>,<
- اسائنمنٹ آپریٹرز (Assignment): %=, //=, /=,*=,-=,+=,=
- منطقی آپریٹرز (Logical): not,or,and
- کمنٹس آپریٹرز (Comments) کوڈ کو پڑھنے میں آسان بنانے کے لیے استعمال ہوتے ہیں: #



مشق

کثیر انتخابی سوالات

i. درج ذیل میں سے کون سا مرحلہ بنیادی پروگرامنگ کے عمل کا حصہ نہیں ہے؟

(الف) کوڈ لکھنا　　(ب) مرتب کرنا / تشریح کرنا

(ج) عمل درآمد (Execute) کرنا　　(د) غلطیوں کو نظر انداز کرنا

ii. کمانڈ لائن سے آسانی سے چلانے کے لیے پائتھن انسٹال کرتے وقت آپ کو کیا کرنا چاہیے؟

(الف) Add Python to PATH کو اَن چیک کریں　　(ب) ایک مختلف IDE کا انتخاب کریں

(ج) Add Python to PATH کو چیک کریں　　(د) صرف IDE انسٹال کریں

iii print(10/ /3) کا نتیجہ کیا ہو گا؟

الف 3.333
ب 3
ج 1
د 4

iv x =123 میں x کی ڈیٹا ٹائپ کیا ہے؟

الف انٹیجر
ب فلوٹ
ج سٹرنگ
د بولیئن

v پائتھن میں کس آپریٹر کی سب سے زیادہ ترجیح (highest precedence) ہے؟

الف جمع (+)
ب ضرب (*)
ج قوت نما (**)
د قوسین ()

vi 2<1and 3>5 کا نتیجہ کیا ہے؟

الف True
ب False
ج Error
د None

vii += آپریٹر کیا کرتا ہے؟

الف دو نمبروں کو جمع کرتا ہے
ب قدروں کا موازنہ کرتا ہے
ج جمع کرتا ہے اور تفویض (assign) کرتا ہے
د قدروں کو ضرب دیتا ہے

vii کون سا فنکشن سٹرنگ ان پٹ کو انٹیجر میں تبدیل کرتا ہے؟

الف str()
ب int()
ج Float()
د input()

vii درج ذیل کوڈ کا نتیجہ کیا ہے؟ age = 25; print(" Age : " , age)

الف Age: 25
ب 25
ج Age
د Age

1. پائتھن میں کمنٹس (comments) کے لیے کون سی علامت استعمال ہوتی ہے؟

الف //
ب */
ج #
د –

مختصر سوالات

1. پروگرامنگ میں کمنٹس کیوں اہم ہیں؟
2. پائتھن میں متغیرات کی تین بنیادی اقسام پر بحث کریں۔
3. پائتھن میں integer اور float ڈیٹا اقسام کے درمیان فرق بیان کریں۔
4. منطقی آپریٹرز کیا ہیں اور تینوں کے نام بتائیں؟
5. آپ پائتھن میں صارف سے ان پٹ کیسے حاصل کرتے ہیں؟
6. پائتھن میں print() سٹیٹمنٹ کا مقصد کیا ہے؟
7. آپریٹر کی ترجیح کیا ہے؟ ایک مثال دیں۔
8. پائتھن میں متغیرات کے نام رکھنے کے تین اہم اصول کیا ہیں؟

تفصیلی سوالات

1. پائتھون میں بنیادی ڈیٹا کی اقسام کیا ہیں؟ مثالوں کے ساتھ integers، floats، strings، اور Booleans کی وضاحت کریں۔
2. پائتھون میں حسابی آپریٹرز کیا ہیں۔ ہر آپریٹر کی مثالیں فراہم کریں اور وضاحت کریں کہ آپ انہیں کب استعمال کریں گے۔
3. بیان کریں کہ پائتھن میں صارفین سے ان پٹ کیسے حاصل کیا جائے اور آوٹ پٹ کیسے دکھایا جائے۔ مثالوں کے ساتھ input() اور print() فنکشنز کی وضاحت کریں۔
4. وضاحت کریں کہ تقابلی اور منطقی آپریٹرز مشروط سٹیٹمنٹس (conditional statements) میں مل کر کیسے کام کرتے ہیں۔ ایک عملی مثال فراہم کریں۔
5. پائتھن میں مشروط سٹیٹمنٹس کیا ہیں؟ مناسب سینٹکس اور عملی مثالوں کے ساتھ if، else، اور else-if سٹیٹمنٹس کی وضاحت کریں۔

6. پائتھن میں ایک ایسا پروگرام لکھیں جو صارف سے ایک نام لے اور دیے گئے نام کو خوش آمدید (greeting) کا پیغام دکھائے۔

7. پائتھن میں ایک ایسا پروگرام لکھیں جو صارف سے دو نمبر لے اور ان کا حاصل جمع دکھائے۔

8. پائتھن میں ایک ایسا پروگرام لکھیں جو صارف سے پانچ نمبر لے اور ان کا اوسط دکھائے۔

9. درج ذیل ایکسپریشنز کو حل کریں:

a) `8 + 3 * 4 - 2 ** 2`

b) `20 % 7 * 3 + 10 // 3`

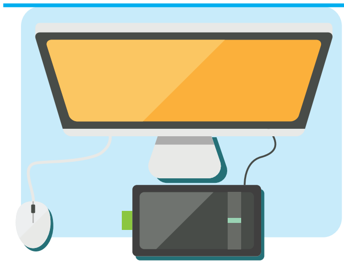
c) `2 * 3 ** 2 % 4 + 10 - 6 / 2`

کثیر انتخابی سوالات کے جوابات

1. د غلطیوں کو نظر انداز کرنا (Ignore Errors)
2. ج "Add Python to PATH" کو چیک کریں
3. ب 3
4. ج سٹرنگ (String)
5. د قوسین ()
6. ب False
7. ج جمع کرتا ہے اور تفویض کرتا ہے (Adds and assigns)
8. ب int()
9. الف Age: 25
10. ج #

یونٹ

4 پائتھن میں کنٹرول اسٹرکچرز



حاصلاتِ تعلم: اس باب کے اختتام تک، آپ اس قابل ہو جائیں گے کہ:

- پائتھن میں کنٹرول اسٹرکچرز کو نافذ کر سکیں، جیسے کہ فیصلہ سازی کی سٹیٹمنس اور لوپس۔
- پائتھن کے ماڈیولز، فنکشنز، اور بلٹ اِن ڈیٹا اسٹرکچرز جیسے کہ لسٹس (Lists) کے ساتھ کام کر سکیں۔
- لِسٹس (Lists) پر مختلف آپریشنز پرفارم کر سکیں۔

تعارف

پروگرامنگ میں، ہمیں اکثر مختلف شرائط (conditions) کی بنیاد پر اپنے پروگرام کے فلو کو کنٹرول کرنے یا کسی خاص عمل کو کئی بار دہرانے کی ضرورت ہوتی ہے۔ یہی وہ مقام ہے جہاں کنٹرول اسٹرکچرز اپنا کردار ادا کرتے ہیں۔

کنٹرول اسٹرکچرز بہت ضروری ہیں کیونکہ یہ ہمارے پروگراموں میں فیصلہ سازی کے عمل (decision-making process) اور کاموں کی تکرار (repetition of tasks) کو منظم کرنے میں ہماری مدد کرتے ہیں۔

کنٹرول اسٹرکچرز کی دو اہم اقسام ہیں:

1. فیصلہ سازی (Decision making)
2. تکرار (Looping)

4.1 فیصلہ سازی

پروگرامنگ میں فیصلہ سازی پروگرام کو شرائط (Conditions) کی بنیاد پر مختلف اقدامات کا انتخاب کرنے کی اجازت دیتی ہے۔

یہ بالکل ویسا ہی ہے جیسے ہم حقیقی زندگی میں فیصلے کرتے ہیں۔ مثال کے طور پر، موسم کی صورتحال کی بنیاد پر چھتری لینے یا نہ لینے کا فیصلہ کرنا۔

پائتھن فیصلہ سازی کو نافذ کرنے کے لیے کئی طرح کے کنڈیشنل سٹیٹمنٹس فراہم کرتا ہے۔

if سٹیٹمنٹ

if سٹیٹمنٹ ہمیں شرائط کی بنیاد پر فیصلے کرنے دیتا ہے۔

اگر شرط درست (True) ہو، تو یہ کوڈ کے ایک بلاک (Block) کو چلاتا ہے۔

پائتھن میں، اِنڈینٹیشن (Indentation) ہی آپ کے کوڈ کے ڈھانچے / بلاک یا دائرہ کار (Scope) کا تعین کرتی ہے۔
پائتھن میں مناسب اِنڈینٹیشن لازمی ہے۔ غلط اِنڈینٹیشن کی وجہ سے آپ کے کوڈ میں "Indentation Error" آتا ہے۔

if اسٹیٹمنٹ کا سِنٹیکس

```
# Syntax of if statement
if condition:
        code to run if the condition is true
```

اگر شرط درست ہو:

یہ کوڈ اس وقت چلے گا جب شرط درست (True) ہو

مثال: اگر درجہ حرارت 30 ڈگری سے زیادہ ہو، تو ہم ایک پیغام (مثلاً: "آج بہت گرمی ہے!") پرنٹ کرتے ہیں۔

```
temperature = 35
if temperature > 30:
       print("It's a hot day")
# Output:  It's a hot day
```

if-else (Statement)

if-else ہمیں اجازت دیتا ہے کہ اگر شرط درست (True) ہو تو کوڈ کا ایک بلاک چلائیں اور اگر شرط غلط (False) ہو تو کوڈ کا دوسرا بلاک چلائیں۔

if - else اسٹیٹمنٹ کا سِنٹیکس

if-else کا ڈھانچہ۔

if کنڈیشن: شرط درست ہونے پر عمل درآمد ہوتا ہے۔

else بلاک: شرط درست نہ ہونے (یعنی غلط ہونے) پر عمل درآمد ہوتا ہے۔

مثال:

if-else اسٹیٹمنٹ کا استعمال کرتے ہوئے درجہ حرارت چیک کرنے کے لیے ایک پائتھن پروگرام لکھیں۔"
اگر درجہ حرارت 30 ڈگری سے زیادہ ہو تو پروگرام یہ دکھائے 'It's a hot day'
اور اگر درجہ حرارت 30 ڈگری سے کم یا برابر ہو تو پروگرام یہ ('It's not a hot day') دکھائے

```
Temperature = 15
 if temperature > 30:
     print("It's a hot day ")
 else :
     print("It's not a hot day " )
# It's not a hot day
```

نیسٹیڈ کینڈیشنز

کبھی کبھی ہمیں ایک شرط کے اندر دوسری شرط چیک کرنے کی ضرورت ہوتی ہے۔ اسے نیسٹنگ کہا جاتا ہے۔

 # Syntax of nested if statement — اسٹیٹمنٹ کا سِنٹیکس

if condition1: — if کنڈیشن 1:

    if condition 2: — if کنڈیشن 2:

یہ کوڈ تب چلے گا جب دونوں کنڈیشنز درست ہوں تو

        code to run if both condition1 and condition2 are true

    else: — else

یہ کوڈ تب چلے گا جب پہلی کنڈیشن درست ہو اور دوسری کنڈیشن غلط ہو۔

        code to run if condition 1 is true but condition2 is false

else: — else

    code to run if condition 1 is false — یہ کوڈ تب چلے گا جب پہلی کنڈیشن غلط ہو۔

مثال: اگر موسم بارش والا ہو اور درجہ حرارت 15 ڈگری سے کم ہو، تو ہم رین کوٹ پہنتے ہیں۔ اگر صرف بارش ہو رہی ہو تو ہم چھتری لیتے ہیں۔ اگر موسم بارش والا نہ ہو تو ہم بس دن کا مزہ لیتے ہیں۔

```
weather = "rainy"
temperature = 10
if weather == "rainy":
       if temperature < 15:
              print("Wear a raincoat")
        else :
               print("Take an umbrella")
else:
         print("Enjoy your day!")
# Output: Wear a raincoat
```

وضاحت

اس مثال میں، کوڈ سب سے پہلے یہ چیک کرتا ہے کہ آیا موسم بارش والا ہے یا نہیں۔

اگر موسم بارش والا ہو تو یہ پھر یہ چیک کرتا ہے کہ آیا درجہ حرارت 15 ڈگری سے کم ہے یا نہیں۔ اگر دونوں شرائط درست ہوں تو یہ پیغام دکھاتا ہے: "رین کوٹ پہنیں"۔

اگر موسم بارش والا ہو لیکن درجہ حرارت 15 ڈگری سے کم نہ ہو تو یہ پیغام دکھاتا ہے: "چھتری لیں"۔

اگر موسم بارش والا نہ ہو تو یہ پیغام دکھاتا ہے: "دن کا مزہ لیں"۔

لوپنگ کنسٹرکٹ (Looping Constructs)

لوپس ہمیں کسی کام کو بار بار دہرانے میں مدد دیتے ہیں، جس سے ہمارا کوڈ زیادہ مؤثر اور پڑھنے میں آسان ہو جاتا ہے۔

پائتھن میں لوپس کی دو اہم اقسام ہیں : while for

whileلوپ

ایکwhileلوپ تب تک چلتاہے جب تک دی گئی شرط درست ہو۔ یہ ہر تکرار(iterartion)سے پہلے شرط کو چیک کرتا ہے اور جب شرط درست نہ رہے تو لوپ رک جاتا ہے۔

while لوپ کا سِنٹیکس

```
# Syntax of while loop
while condition:
        code to run while the condition is true
```

whileکنڈیشن

یہ کوڈ تب تک چلے گا جب تک شرط درست

مثال:ایک نمبر میں 1 جمع کرتے جائیں جب تک وہ 10 تک نہ پہنچ جائے۔

```
number = 1
while number < 10:
    print(number)
    number += 1
```

```
'''
Output:
1
2
3
4
5
6
7
8
9
'''
```

وضاحت:

اس مثال میں، کوڈ ایک نمبر سے شروع ہوتا ہے جس کی قیمت 1 رکھی گئی ہے۔ `while` لوپ چیک کرتا ہے کہ کیا نمبر 10 سے کم ہے۔

اگر نمبر 10 سے کم ہے تو پروگرام موجودہ نمبر پرنٹ کرتا ہے اور پھر اس میں 1 جمع کر دیتا ہے۔

یہ لوپ اسی طرح چلتا رہتا ہے جب تک نمبر 10 تک نہ پہنچ جائے، جس کے بعد لوپ رک جاتا ہے۔

forلوپ

forلوپ ایک کوڈ کے ایک بلاک کو ایک مخصوص تعداد میں دہراتا ہے۔ یہ عام طور پر ایک ترتیب جیسے کہ ایک لسٹ، ٹوپل، یا سٹرنگ پر تکرار کرنے کے لیے استعمال ہوتا ہے۔

forلوپ کا سنٹیکس

```
# Syntax of for loop
for variable in sequence:
    code to run for each element in the sequence
```

کسی ترتیب میں شامل ہر قدر کے لیے

ترتیب میں موجود ہر ایلیمنٹ کے لیے کوڈ چلے گا

مثال:دوستوں کی فہرست میں موجود ہر دوست کو "ویلکم" لکھیں۔

```
  friends = ["Sami", "Raza", "Moosa"]
  for friend in friends:
         print("Welcome to ", friend)
'''
Output:
Welcome to: Sami
Welcome to: Raza
Welcome to: Moosa
'''
```

range() فنکشن

ہم range() فنکشن کا استعمال نمبروں کی ایک ترتیب (sequence) بنانے کے لیے کر سکتے ہیں، جو اکثر for لوپس میں استعمال ہوتا ہے۔

رینج فنکشن کا سینٹکس

range(stop)

range (start, stop)

range (start, stop, step)

مثال: پہلے 5 مکمل اعداد (Whole Numbers) پرنٹ کریں۔

```
for i in range (5):
     print (i)
'''
Output:
0
1
2
3
4
'''
```

end سٹیٹمنٹ (end statement)

پائتھن میں، end سٹیٹمنٹ کوئی الگ کمانڈ نہیں ہے، بلکہ یہ print() فنکشن کا ایک آپشنل پیرامیٹر (optional parameter) ہے۔ بطور ڈیفالٹ، پائتھن میں ہر print() سٹیٹمنٹ ایک نئی لائن پر ختم ہوتی ہے، لیکن آپ end پیرامیٹر کا استعمال کرتے ہوئے یہ تبدیل کر سکتے ہیں کہ آخر میں کیا پرنٹ کیا جائے۔

مثال: "end" سٹیٹمنٹ کا استعمال کرتے ہوئے دو سٹرنگز کو ایک ہی لائن پر پرنٹ کریں۔

```
print("Arshman", end=" ")
print("Ashbil")
# Output:  Arshman Ashbil
print("Ayyan", end="***")
print("Naveed")
# Output: Ayyan***Naveed
```

کیا آپ جانتے ہیں؟

پائتھن میں سٹرنگز یا تو سنگل کوٹیشن کی علامات "(Single quotation marks) یا ڈبل کوٹیشن کی علامات "" (double quotation marks) سے گھیرے جاتے ہیں۔ مثال کے طور پر 'Ayyan' اور "Ayyan" ایک جیسے ہیں۔

مثال : "end" سٹیٹمنٹ کا استعمال کرتے ہوئے 1 سے 5 تک کے اعداد کو ایک ہی لائن پر پرنٹ کریں۔

```
for i in range(1, 6):
  # if you want to print Numbers on same line use following line
              print(i, end=" ")
  # Output: 1 2 3 4 5
```

سرگرمی:

1. () range فنکشن کا استعمال کرتے ہوئے ایک for لوپ لکھیں تاکہ 2 سے 10 تک کے جفت اعداد (even numbers) کو ایک ہی لائن پر پرنٹ کیا جا سکے۔
2. یک پائتھون پروگرام لکھیں جو for لوپ اور () range فنکشن کا استعمال کرتے ہوئے 3 کے پہلے 10multiples کو پرنٹ کرے۔

نیسٹیڈ لوپس

نیسٹیڈ لوپس وہ لوپس ہیں جو دوسرے لوپس کے اندر رہوتے ہیں۔ بیرونی لوپ (Outer loop) پہلے چلتا ہے۔ بیرونی لوپ کے ہر ایک تکرار (iteration) کے لیے، اندرونی لوپ (Inner loop) مکمل طور پر چلتا ہے۔ ہم نیسٹیڈ لوپس کا استعمال اس وقت کرتے ہیں جب ہمیں کسی کام کے اندر بار بار دہرائے جانے والے کام کو انجام دینے کی ضرورت ہوتی ہے۔

مثال: بنیادی نمبر پیٹرن کے لیے نیسٹیڈ لوپس

```
print("=== Basic Number Pattern ===")
for i in range(3):       # Outer loop
    for j in range(4):  # Inner loop
        print(f"i={i}, j={j}")
    print("---")        # Separator after each outer loop iteration
'''
# Output
=== Basic Number Pattern ===
i=0, j=0
i=0, j=1
i=0, j=2
i=0, j=3
---
i=1, j=0
i=1, j=1
i=1, j=2
i=1, j=3
---
i=2, j=0
i=2, j=1
i=2, j=2
i=2, j=3
_ '''
```

مثال:ستاروں کو تکونی پیٹرن میں پرنٹ کرنے کے لیے نیسٹیڈ لوپ

```
print("=== Right-Angled Triangle ===")
n = 5
for i in range(1, n+1):
    for j in range(i):      # Inner loop runs 'i' times
        print("*", end=" ")
    print()   # New line after each row
'''
# Output
*
* *
* * *
* * * *
* * * * *
'''
```

4.2 پائتھن میں لائبریریز

پائتھن ایک وسیع معیاری لائبریری پیش کرتا ہے جس میں بلٹ اِن ماڈیولز(built-in modules)اور ڈیٹا سٹرکچرز شامل ہیں۔ڈیٹا سٹرکچر سے مراد ڈیٹا کو منظم کرنے اور ذخیرہ کرنے کا ایک مخصوص فارمیٹ یا طریقہ ہے۔اس سیکشن میں،ہم پائتھن میں مختلف لائبریریز کے استعمال کا جائزہ لیں گے۔

پائتھن میں لائبریریز کا استعمال

پائتھن میں،لائبریریز کسی ٹول باکس کی طرح ہیں جو مفید ٹولز سے بھری ہوئی ہیں اور جو آپ کو ہر چیز کو از سر نو(from scratch) بنائے بغیر مختلف مسائل کو حل کرنے میں مدد کرتی ہیں۔لائبریریز پہلے سے بنے ہوئے ٹول کٹس کی طرح ہیں جنھیں آپ تمام کوڈ خود لکھے بغیر استعمال کر سکتے ہیں۔آیئے کچھ سادہ مثالوں کے ساتھ دیکھتے ہیں کہ مختلف لائبریریز کو کیسے اِمپورٹ(import)اور استعمال کیا جائے۔

مثال:رینڈم لائبریری کو رینڈم اعداد(random numbers)بنانے کے لیے اِمپورٹ کریں۔

```
import random
# Generate a random number between 1 and 10
number = random.randint(1, 10)
print("The random number is:", number)
#Output:
# The random number is: 3
```

مثال: سٹیٹسٹکس(statistics)لائبریری کو شماریاتی کیلکولیشن انجام دینے کے لیے اِمپورٹ کریں۔

```
import statistics
# Calculate the mean of a list of numbers
data = [23, 45, 67, 89, 12, 44, 56]
mean_value = statistics.mean(data)
print("The mean value is:", mean_value)
# Output:
# The mean value is: 48
```

وضاحت:statisticsلائبریری بنیادی شماریاتی کیلکولیشن انجام دینے کے لیے ایک بہترین ٹول ہے،جیسے کہ ڈیٹا کے ایک سیٹ کا اوسط(mean)،درمیانی قیمت(median)،اور عام ویلیو(mode)معلوم کرنا۔یہ خاص طور پر ڈیٹا اینالیسس کے کاموں میں مفید ہو سکتا ہے۔

مختصراً،ان لائبریریز کا استعمال کرکے،آپ وقت اور محنت کی بچت کر سکتے ہیں،جس سے آپ پہیہ دوبارہ ایجاد (reinventing the wheel)کرنے کے بجائے زیرِ بحث مخصوص مسئلے کو حل کرنے پر توجہ مرکوز کر سکتے ہیں۔

SMART TIPS

سمارٹ ٹپس

لائبریریز کو اِمپورٹ کرتے وقت، یقینی بنائیں کہ آپ صرف وہی چیزیں اِمپورٹ کر رہے ہیں جن کی آپ کو ضرورت ہے۔ آپ مزید لائبریریز پر دیکھ سکتے ہیں۔https://docs.python.org/3

4.3 پائتھن میں لسٹیں

پائتھن میں، ایک لسٹ ایک کثیر المقاصد (versatile) ڈیٹا سٹرکچر ہے جو آئٹمز کا ایک مجموعہ رکھ سکتی ہے۔ آپ آسانی سے لسٹیں بنا، سکتے ہیں۔ اِن تک رسائی حاصل کر سکتے ہیں، اور اِن میں ترمیم کر سکتے ہیں۔

لسٹیں بنانا

ایک لسٹ مربع قوسین [] کے اندر آئٹمز کو رکھ کر بنائی جاتی ہے، جنہیں کاما (comma) سے الگ کیا جاتا ہے۔ لسٹوں میں مختلف اقسام کی اشیاء شامل ہو سکتی ہیں، جیسے کہ نمبرز، سٹرنگز، یا یہاں تک کہ دوسری لسٹیں بھی۔

مثال: اپنے پسندیدہ پھلوں کی ایک لسٹ بنائیں۔

```
fruits = ["Mango", "Apple","Banana"]
 print(fruits)
# Output: ['Mango' 'Apple ' 'Banana ' ]
```

وضاحت: یہ کوڈ fruits نامی ایک لسٹ بناتا ہے جس میں تین عناصر (elements) شامل ہیں، اور پھر اس لسٹ کو پرنٹ کرتا ہے۔

لسٹ آئٹمز تک رسائی

آپ ایک لسٹ میں موجود آئٹمز تک ان کے انڈیکس کا حوالہ دے کر رسائی حاصل کر سکتے ہیں، جو 0 سے شروع ہوتا ہے۔

مثال: پھلوں کی لسٹ سے دوسری آئٹم تک رسائی حاصل کریں اور اسے پرنٹ کریں۔

```
fruits = ["Mango", "Apple", "" Banana"]
print(fruits [1])
# Output: Apple
```

وضاحت: یہ کوڈ fruits نامی ایک لسٹ شروع کرتا ہے، جس میں "Apple، Mango"، اور 'Banana' شامل ہیں، اور پھر انڈیکس [1] کا استعمال کرتے ہوئے دوسری آئٹم 'Apple' کو پرنٹ کرتا ہے۔

سرگرمی:

ایک پائتھن پروگرام لکھیں جو رینڈم اور statistics لائبریریوں کو اِمپورٹ کرے۔ رینڈم کا استعمال کرتے ہوئے 1 سے 50 کے درمیان 10 رینڈم اعداد کی ایک لسٹ بنائیں۔ پھر، statistics کا استعمال کرتے ہوئے ان اعداد کا اوسط (mean) نکالیں اور پرنٹ کریں۔

لسٹ میں ترمیم کرنا

آپ لسٹ آئٹمز تک ان کے انڈیکس کے ذریعے رسائی حاصل کر کے اور ایک نئی قدر تفویض (assign) کر کے ترمیم کر سکتے ہیں۔

مثال: لسٹ میں پہلی آئٹم کو "Orange" میں تبدیل کریں اور ایک نیا پھل "Pineapple" شامل کریں۔

```
fruits = ["Mango", "Apple", "Banana"]
fruits [0] = "Orange"
fruits.append("Pineapple")
print(fruits)
# Output: ['Orange', 'Apple', 'Banana', 'Pineapple']
```

وضاحت: یہ کوڈ fruits لسٹ کے پہلے عنصر کو تبدیل کر کے 'Orange' کرتا ہے، آخر میں 'Pineapple' کا اِضافہ (appends) کرتا ہے، اور پھر اپ ڈیٹ شدہ لسٹ کو پرنٹ کرتا ہے۔

لسٹ آپریشنز

پائتھن لسٹوں کے ساتھ کام کرنے کے لیے کئی بلٹ اِن میتھڈز فراہم کرتا ہے۔ یہاں کچھ مفید میتھڈز درج ذیل ہیں:

- append(item): لسٹ کے آخر میں ایک آئٹم کا اِضافہ کرتا ہے۔
- remove(item): لسٹ میں سے آئٹم کے پہلے وقوع (first occurrence) کو ہٹاتا ہے۔
- sort(): لسٹ کو صعودی ترتیب (ascending order) میں ترتیب دیتا ہے۔
- reverse(): لسٹ کی ترتیب کو اُلٹ (reverse) دیتا ہے۔

مثال: طلباء کی لسٹ میں ایک نیا طالب علم شامل کریں اور لسٹ پرنٹ کریں۔

```
students = ["Ahmed", "Sara", "Ali"]
students.append("Hina")
 print(students)
#Output : ['Ahmed', 'Ali', 'Sara', 'Hina']
```

وضاحت: یہ کوڈ طلباء کی ایک لسٹ بناتا ہے، اس میں 'Hina' کو شامل (adds) کرتا ہے۔

مثال: طلباء کی لسٹ میں سے ایک طالب علم کو ہٹادیں (Remove) اور لسٹ پرنٹ کریں۔

```
# Remove a student from the list of students
students = ["Ahmed", "Sara", "Ali", "Hina", "Sara"]
students.remove("Sara")
print(students)
# Output: ['Ahmed', 'Ali', 'Hina', 'Sara']
```

وضاحت: یہ کوڈ طلباء کی ایک لسٹ بناتا ہے، لسٹ میں سے "Sara" کے پہلے وقوع (first occurrence) کو ہٹاتا ہے، اور پھر اپ ڈیٹ شدہ لسٹ کو پرنٹ کرتا ہے۔ نوٹ کریں کہ صرف پہلی مماثل آئٹم ہی ہٹائی جاتی ہے۔

مثال: اعداد کی ایک لسٹ کو صعودی ترتیب (ascending order) میں ترتیب دیں۔

```
# Sort a list of numbers in ascending order
numbers = [34, 12, 89, 5, 23]
numbers.sort()
print(numbers)
# Output: [5, 12, 23, 34, 89]
```

وضاحت: یہ کوڈ اعداد کی ایک لسٹ بناتا ہے، ()sort میتھڈ کا استعمال کرتے ہوئے انہیں صعودی ترتیب (ascending order) میں ترتیب دیتا ہے، اور پھر ترتیب شدہ لسٹ کو پرنٹ کرتا ہے۔

مثال: لسٹ میں موجود پھلوں کی ترتیب کو اُلٹ (Reverse) دیں۔

```
# Reverse the order of fruits in a list
fruits = ["Apple", "Banana", "Orange", "Mango"]
fruits.reverse()
print(fruits)
# Output: ['Mango', 'Orange', 'Banana', 'Apple']
```

وضاحت: یہ کوڈ پھلوں کی ایک لسٹ بناتا ہے، ()reverse میتھڈ کا استعمال کرتے ہوئے عناصر کی ترتیب کو اُلٹ (reverses) دیتا ہے، اور پھر اُلٹی ہوئی لسٹ کو پرنٹ کرتا ہے۔

4.5 ٹیسٹنگ اور ڈیبگنگ

پائتھن پروگرامنگ میں کوڈ کو ٹیسٹ کرنا اور اس میں سے غلطیاں نکالنا ایک ضروری عمل ہے۔

ٹیسٹنگ

کوڈ کو بار بار مختلف ویلیوز کے ساتھ ملا کر دیکھنا اور ممکنہ غلطیوں کی نشاندہی کرنا ٹیسٹنگ کہلاتا ہے۔ اس کا مقصد کوڈ کو استعمال میں لانے سے پہلے غلطیوں کو ختم کرنا ہوتا ہے۔

ٹیسٹنگ کی اقسام

- یونٹ ٹیسٹنگ : کوڈ کے انفرادی حصوں (جیسے فنکشنز یا کلاسز) کو الگ تھلگ کر کے جانچتا ہے۔ پائتھن کا یونٹ ٹیسٹ ماڈیول عام طور پر اس کے لیے استعمال ہوتا ہے۔
- انٹیگریشن ٹیسٹنگ : جانچتا ہے کہ کوڈ کے مختلف حصے ایک ساتھ کیسے کام کرتے ہیں۔
- فنکشنل ٹیسٹنگ : توثیق کرتا ہے کہ سافٹ ویئر صارف کے نقطہ نظر سے توقع کے مطابق کام کرتا ہے۔
- ریگریشن ٹیسٹنگ : یقینی بناتا ہے کہ نئی تبدیلیاں موجودہ فعالیت کو خراب نہیں کرتی ہیں۔

ڈیبگنگ (Debugging)

ڈیبگنگ آپ کے کوڈ میں موجود غلطیوں (errors) یا بگز (bugs) کو تلاش کرنے اور ٹھیک کرنے کا عمل ہے۔ اس میں مسائل کی بنیادی وجہ کی نشاندہی کرنا اور ضروری تبدیلیاں کرنا شامل ہے۔

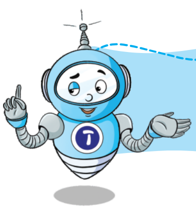
عام ڈیبگنگ تکنیکس

- **پرنٹ سٹیٹمنٹس** : کوڈ کے مختلف مراحل پر متغیرات کی ویلیو کو چیک کرنے کے لیے print سٹیٹمنٹس کا اضافہ کرنا۔
- **ڈیبگنگ ٹولز** : کوڈ سے گزرنے، متغیرات کا معائنہ کرنے، اور عمل درآمد کے بہاؤ (flow of execution) کو سمجھنے کے لیے pdb (Python Debugger) جیسے ٹولز کا استعمال کرنا۔
- **ایرر میسجز** : مسئلے کا ماخذ تلاش کرنے کے لیے ایرر میسجز کو پڑھنا اور ان کی تشریح کرنا۔

خلاصہ (Summary)

- پروگرامنگ میں فیصلہ سازی پروگرام کو اس قابل بناتی ہے کہ وہ شرائط کی بنیاد پر مختلف اقدامات کا انتخاب کر سکے۔
- if سٹیٹمنٹ ہمیں شرائط کی بنیاد پر فیصلے کرنے دیتی ہے۔ اگر شرط درست (true) ہو، تو یہ کوڈ کا ایک مخصوص حصہ (block) چلاتی ہے۔
- if-else سٹیٹمنٹ ہمیں اس بات کی اجازت دیتی ہے کہ اگر شرط درست ہو تو کوڈ کا ایک حصہ چلے، اور اگر شرط غلط (false) ہو تو دوسرا حصہ چلے۔
- کبھی کبھار ہمیں ایک شرط کے اندر مزید کئی شرائط کو چیک کرنے کی ضرورت ہوتی ہے، اسے نیسٹنگ (Nesting) کہا جاتا ہے۔

- لوپس(Loops)ہمیں کاموں کو دہرانے میں مدد دیتے ہیں، جس سے ہمارا کوڈ زیادہ مؤثر اور پڑھنے میں آسان ہو جاتا ہے۔
- ایک whileلوپ اس وقت تک چلتاہے جب تک شرط درست رہے۔ یہ ہر بار چلنے سے پہلے شرط کو چیک کرتا ہے اور شرط غلط ہونے کی صورت میں رک جاتا ہے۔
- ایک forلوپ کوڈ کے ایک حصے کو مخصوص تعداد میں دہراتا ہے۔ یہ عام طور پر کسی ترتیب(sequence) پر کام کرنے کے لیے استعمال ہوتا ہے۔
- ہم نمبروں کی ترتیب پیدا کرنے کے لیے ()range فنکشن استعمال کرسکتے ہیں، جو اکثر forلوپس میں استعمال ہوتا ہے۔
- پائیتھن میں endسٹیٹمنٹ کوئی الگ کمانڈ نہیں ہے، بلکہ یہ () print فنکشن کا ایک اختیاری پیرامیٹر ہے۔
- نیسٹڈ لوپس(Nested loops) ایسے لوپس ہوتے ہیں جو دوسرے لوپس کے اندر موجود ہوں۔ بیرونی لوپ (outer loop)پہلے چلتاہے، اور بیرونی لوپ کے ہر ایک چکر کے لیے اندرونی لوپ(inner loop) مکمل طور پر چلتا ہے۔
- پائیتھن میں لسٹ ایک ہمہ گیر ڈیٹا اسٹرکچر ہے جو مختلف اشیاء کا مجموعہ رکھ سکتی ہے۔ آپ آسانی سے لسٹ بنا سکتے ہیں، اسے استعمال کرسکتے ہیں اور اس میں تبدیلی کرسکتے ہیں۔
- ٹیسٹنگ کا مطلب کوڈ کو مختلف ان پٹس کے ساتھ چلا کر یہ دیکھنا ہے کہ آیا یہ توقع کے مطابق کام کر رہا ہے یا نہیں۔
- ڈی بگنگ کوڈ میں غلطیوں (bugs) کو تلاش کرنے اور انہیں ٹھیک کرنے کا عمل ہے۔ اس میں مسئلے کی اصل وجہ پہچاننا اور ضروری تبدیلیاں کرنا شامل ہے۔



مشق

کثیر انتخابی سوالات

i پائتھن میں ()range فنکشن کیا کرتا ہے؟

الف) اعداد کی ایک لسٹ بناتا ہے
ب) اعداد کی ایک ترتیب بناتا ہے
ج) اعداد کا حاصل جمع نکالتا ہے
د) اعداد کی ایک رینج پرنٹ کرتا ہے

ii درج ذیل کوڈ کا نتیجہ کیا ہوگا؟

```
count = 0
for i in range(2):
    for j in range(3):
        count += 1
print(count)
```

الف) 4
ب) 5
ج) 6
د) 7

iii. پائتھن میں لسٹ کے آخر میں ایک آئٹم شامل کرنے کے لیے کون سا میتھڈ استعمال ہوتا ہے؟

الف) insert()
ب) append()
ج) add()
د) extend()

iv. پروگرامنگ میں کنٹرول سٹرکچرز کا مقصد کیا ہے؟

الف) صرف کوڈ کو لمبا کرنا
ب) فیصلہ سازی اور کاموں کی ریپٹیشن کو منظم کرنا
ج) متغیرات بنانا
د) لائبریریز اِمپورٹ کرنا

v. کون سی سٹیٹمنٹ استعمال ہوتی ہے جب کوئی شرط غلط (false) ہو تو کوڈ کو چلایا جائے؟

الف) if
ب) elif
ج) else
د) while

vi. آپ print() کو نئی لائن شامل کرنے سے کیسے روکتے ہیں؟

الف) end="" استعمال کریں
ب) newline=False استعمال کریں
ج) Break استعمال کریں
د) Continue استعمال کریں

vii. رینڈم اعداد (random numbers) بنانے کے لیے کون سی لائبریری استعمال ہوتی ہے؟

الف) Datetime
ب) statistics
ج) Random
د) math

viii. list.append() کیا کرتا ہے؟

الف) پہلی آئٹم کو ہٹاتا ہے
ب) آخر میں ایک آئٹم شامل کرتا ہے
ج) لسٹ کو ترتیب دیتا ہے
د) لسٹ کو الٹ (reverse) دیتا ہے

ix. range() فنکشن کا مقصد کیا ہے؟

الف) اعداد کی ایک لسٹ بنانا
ب) اعداد کی ایک ترتیب پیدا کرنا
ج) قدروں کی رینج کا حساب لگانا
د) اعداد کو ترتیب دینا

x کون سی ڈیبگنگ تکنیک آؤٹ پٹ سٹیٹمنٹس کو شامل کرنے پر مشتمل ہے؟

الف یونٹ ٹیسٹنگ
ب پرنٹ سٹیٹمنٹس
ج ایرر میسجز
د انٹیگریشن ٹیسٹنگ

مختصر سوالات

1. forلوپ میں ()range فنکشن کے استعمال کی وضاحت کریں۔
2. پائتھن لسٹوں میں appendاور remove میتھڈز کے درمیان کیا فرق ہے؟
3. ڈی بگنگ کی تعریف کریں۔
4. ()range فنکشن کے تین پیرامیٹرز کیا ہیں؟
5. لسٹ کے تین میتھڈز اور ان کے مقاصد کے نام بتائیں۔
6. آپ پائتھن میں ایک لائبریری کو کیسے اِمپورٹ کرتے ہیں؟
7. یونٹ ٹیسٹنگ کیا ہے اور یہ کیوں اہم ہے؟
8. آپ ایک لسٹ میں تیسرے عنصر تک کیسے رسائی حاصل کرتے ہیں؟
9. ()print میں end پیرامیٹر کا مقصد کیا ہے؟
10. ہم لسٹ میں سے ایک عنصر تک رسائی کیسے حاصل کر سکتے ہیں؟
11. آپ ایک لسٹ میں موجود موجودہ آئٹم میں کیسے ترمیم کرتے ہیں؟

تفصیلی سوالات

1. پائتھن میں فیصلہ سازی سٹیٹمنٹس کی مختلف اقسام کی مثالوں کے ساتھ وضاحت کریں۔
2. whileلوپس اور forلوپس کا موازنہ اور تضاد کریں۔
3. پائتھن لسٹوں کے ساتھ کام کرنے کے طریقے کو بیان کریں، بشمول ان کو بنانے اور ان میں ترمیم کرنے کا طریقہ۔ آئٹمز شامل کرنے، ہٹانے اور عناصر تک رسائی حاصل کرنے کے لیے مثالیں فراہم کریں۔
4. پروگرامنگ میں ٹیسٹنگ اور ڈیبگنگ کی اہمیت کی وضاحت کریں۔ مختلف ڈیبگنگ تکنیکس پر بحث کریں۔
5. پائتھن لائبریریز پروگرامنگ کی کارکردگی (efficiency) کو کیسے بڑھاتی ہیں؟ ایک مثال فراہم کریں۔

6. ایک پائتھن پروگرام لکھیں جو ایک لوپ کا استعمال کرتے ہوئے 1 اور 20 کے درمیان تمام طاق اعداد (odd numbers) کو پرنٹ کرے۔

7. ایک پائتھن پروگرام لکھیں جو درج ذیل کام انجام دے:

a) 1 اور 20 کے درمیان 5 رینڈم اعداد کی ایک لسٹ بنائے۔

b) پیدا کردہ اعداد کا اوسط (mean) نکالنے اور پرنٹ کرنے کے لیے statistics لائبریری کا استعمال کرے۔

8. ایک پائتھن پروگرام لکھیں جو درج ذیل آپریشنز انجام دے:

a) مشہور پاکستانی کھانوں کی ایک لسٹ بنائیں: ["Biryani", "Nihari", "Karahi", "Seekh Kebabs"]۔

b) لسٹ میں ایک نیا کھانا "Quorma" شامل کریں۔

c) "Karahi" کو "Chicken Karahi" میں تبدیل کریں۔

d) لسٹ میں سے "Nihari" کو ہٹا دیں۔

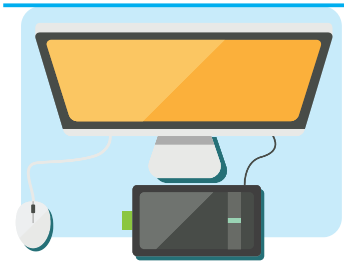
9. نیسٹیڈ لوپس کا استعمال کرتے ہوئے تین قطاروں (rows) اور پانچ کالموں (columns) میں ستاروں (*) کو مستطیل پیٹرن میں پرنٹ کرنے کے لیے کوڈ لکھیں۔

کثیر انتخابی سوالات کے جوابات

1. ب — اعداد کی ایک ترتیب بناتا ہے
2. ج — 6
3. ب — append()
4. ب — فیصلہ سازی اور کاموں کی ریپیٹیشن کو منظم کرنا
5. ج — else
6. الف — end="" استعمال کریں
7. ج — Random
8. ب — آخر میں ایک آئٹم شامل کرتا ہے
9. ب — اعداد کی ایک ترتیب پیدا کرنا
10. ب — ایرر میسجز

یونٹ

5 ڈیٹا سائنس کا تعارف



حاصلاتِ تعلم: اس باب کے اختتام تک، آپ اس قابل ہو جائیں گے کہ:

- ڈیٹا سائنس کے تصور اور روزمرہ زندگی میں اس کی اہمیت کی وضاحت کر سکیں۔
- ڈیٹا سائنس لائف سائیکل (DSLC) کے ہر مرحلہ کو بیان کر سکیں۔
- حقیقی دنیا کے ایسے مسائل کی شناخت اور تعریف کر سکیں جنہیں ڈیٹا کا استعمال کرتے ہوئے حل کیا جا سکتا ہے۔
- سروے، آن لائن فارمز، اور مشاہداتی تکنیک کا استعمال کرتے ہوئے ڈیٹا اکٹھا کر سکیں۔
- درستگی (accuracy) کو بہتر بنانے کے لیے ڈیٹا کلیننگ اور تصدیق (validation) کا عمل انجام دے سکیں۔
- ڈیٹابیس سے ڈیٹا محفوظ کرنے اور بازیافت کرنے کے لیے سادہ SQL سوالات (queries) (جیسے SELECT، INSERT) کا استعمال کر سکیں۔
- بنیادی الگورتھمز جیسے ترتیب (sorting)، درجہ بندی (classification)، کلسٹرنگ کا استعمال کرتے ہوئے ڈیٹا کا تجزیہ کر سکیں۔
- ڈیٹا سے حاصل ہونے والے بصیرت (insights) کو بصری (visual) طور پر ظاہر کرنے کے لیے بار چارٹس اور ڈیش بورڈز بنا سکیں۔
- چارٹس اور سمریز کا استعمال کرتے ہوئے نتائج کو واضح طور پر بیان کر سکیں۔

تعارف

ڈیٹا سائنس ڈیٹا کا استعمال کرتے ہوئے مسائل کو سمجھنے، نمونوں (patterns) کو دریافت کرنے اور دانشمندانہ فیصلے (smart decisions) کرنے کا عمل ہے۔ موجودہ ڈیجیٹل دور میں، ڈیٹا ہر جگہ موجود ہے، جیسے سروے، موبائل ایپس، سوشل میڈیا، آن لائن شاپنگ، اور صحت کی خدمات۔ ڈیٹا سائنس ہمیں اس خام ڈیٹا کو مفید معلومات میں بدلنے میں مدد کرتی ہے۔

چاہے موسم کی پیش گوئی کرنا ہو، طلباء کی دلچسپیوں کو سمجھنا ہو، یا صحت کی دیکھ بھال کو بہتر بنانا ہو، ڈیٹا سائنس ہمارے رہنے، سیکھنے اور کام کرنے کے طریقوں کو تبدیل کر رہی ہے۔ دنیا بھر کی تنظیمیں فیصلے سازی کو بہتر بنانے، اخراجات بچانے اور بہتر خدمات تخلیق کرنے کے لیے ڈیٹا کا استعمال کرتی ہیں۔

یہ باب ڈیٹا سائنس لائف سائیکل (DSLC) کا تعارف کراتا ہے، جس میں مسئلہ کو سمجھنا، ڈیٹا اکٹھا کرنا اور کلین (cleaning) کرنا، ڈیٹا کا تجزیہ اور تشریح کرنا، اور آخر میں چارٹس اور بصری (visual) معاونتوں کا استعمال کرتے ہوئے نتائج فراہم کرنا شامل ہے۔ ہر قدم تنقیدی سوچ اور مسئلہ حل کرنے کی مہارتوں کو فروغ دیتا ہے۔

ہم سیکھیں گے کہ سادہ ٹولز جیسے سروے، سپریڈ شیٹس، اور چارٹس کا استعمال کرتے ہوئے اپنا ڈیٹا کیسے اکٹھا کیا جاتا ہے۔ اس باب میں حقیقی دنیا

کی مثالیں اور عملی سرگرمیاں شامل ہیں جو انہیں کلاس روم اور روزمرہ زندگی میں ڈیٹا سائنس کا اطلاق کرنے میں مدد دیں گی۔

5.1 ڈیٹا سائنس کا تعارف

ہم ایک ڈیٹا سے بھری دنیا میں رہ رہے ہیں۔ چاہے ہم موبائل فون استعمال کر رہے ہوں، گوگل پر تلاش کر رہے ہوں، آن لائن شاپنگ کر رہے ہوں، یا یہاں تک کہ فٹنس ٹریکر کے ساتھ چل رہے ہوں، ڈیٹا اکٹھا کیا جا رہا ہے۔ سوال یہ پیدا ہوتا ہے کہ یہ ڈیٹا کس طرح مفید ہے؟

ڈیٹا سائنس وہ شعبہ ہے جہاں ہم نمونے (Pattern) تلاش کرنے، سوالات کے جواب دینے، اور لوگوں یا کاروباری اداروں کو بہتر فیصلے کرنے میں مدد کے لیے ڈیٹا کا استعمال کرتے ہیں۔ اس میں ڈیٹا کو اکٹھا کرنا، منظم کرنا، تجزیہ کرنا، اور ویژولائیزیشن (visualizing) شامل ہے۔

فیصلہ سازی میں ڈیٹا کی اہمیت

ڈیٹا لوگوں اور تنظیموں کو بہتر فیصلے کرنے میں بہت بڑا کردار ادا کرتا ہے۔ ڈیٹا کے بغیر، فیصلے اندازوں یا ذاتی آراء پر مبنی ہو سکتے ہیں جو غلطیوں کا باعث بن سکتے ہیں۔ مثال کے طور پر، اگر سکول انتظامیہ کوئی نیا سپورٹس کلب شروع کرنا چاہتی ہے، تو وہ پہلے اس بارے میں ڈیٹا اکٹھا کر سکتے ہیں کہ طلباء کون سے کھیل سب سے زیادہ پسند کرتے ہیں۔ یہ انہیں ایک سمارٹ اور منصفانہ انتخاب کرنے میں مدد کرتا ہے۔

ڈیٹا سائنس کے حقیقی زندگی میں اطلاقات

ڈیٹا سائنس ہماری روزمرہ کی زندگی کے کئی پہلوؤں میں اکثر ہمارے علم میں آئے بغیر استعمال ہوتی ہے،:

- **سوشل میڈیا:** فیس بک، یوٹیوب اور انسٹاگرام جیسی ایپلی کیشنز ڈیٹا کا استعمال کرتے ہوئے ہمیں ایسی پوسٹس دکھاتی ہیں جو ہمیں پسند آ سکتی ہیں۔
- **آن لائن شاپنگ:** ای-کامرس ویب سائٹس ہماری پچھلی خریداریوں کے ڈیٹا کا استعمال کرتے ہوئے مصنوعات تجویز کرتی ہیں۔
- **صحت کی دیکھ بھال:** ہسپتال بیماریوں کی پیش گوئی کرنے اور مریضوں کی دیکھ بھال کو زیادہ مؤثر طریقے سے منظم کرنے کے لیے ڈیٹا سائنس کا استعمال کرتے ہیں۔
- **نقل و حمل (Transport):** گوگل میپس جیسی ایپلی کیشنز ٹریفک ڈیٹا کا استعمال کرتے ہوئے تیز ترین راستوں کی تجویز دیتی ہیں۔
- **تعلیم:** سکولوں میں، طلباء کی حاضری اور ٹیسٹ کے نتائج سے حاصل کردہ ڈیٹا کو تدریسی حکمت عملیوں کو بہتر بنانے اور پیش رفت کو ٹریک کرنے کے لیے استعمال کیا جاتا ہے۔
- **ڈیٹا سائنس:** یوٹیوب ویڈیوز دیکھنے سے لے کر صحت کے مشورے حاصل کرنے یا بہترین راستہ منتخب کرنے تک، ہماری زندگی کو زیادہ سمارٹ (smart)، آسان اور زیادہ شخصی نوعیت (personalized) کا بنانے میں مدد کرتی ہے۔

5.2 ڈیٹا سائنس لائف سائیکل (DSLC) کا ایک جائزہ

ڈیٹا سائنس صرف ایک مرحلہ میں مکمل نہیں ہوتی۔ یہ مسائل کو حل کرنے میں مددگار اقدامات کا ایک سلسلہ ہے، جسے ہم ڈیٹا سائنس لائف سائیکل (DSLC) کہتے ہیں۔ ڈیٹا سائنس لائف سائیکل (DSLC) عام طور پر درج ذیل مراحل پر مشتمل ہوتا ہے:

ڈیٹا سائنس لائف سائیکل (DSLC) عام طور پر درج ذیل چھ (6) مراحل پر مشتمل ہوتا ہے:

- **مرحلہ 1:** مسئلہ کو سمجھنا (Understanding the Problem)
- **مرحلہ 2:** ڈیٹا اکٹھا کرنا (Collecting Data)
- **مرحلہ 3:** ڈیٹا کو کلین کرنا اور تصدیق کرنا (Cleaning and Validating Data)
- **مرحلہ 4:** ڈیٹا کا تجزیہ کرنا (Analyzing Data)
- **مرحلہ 5:** ڈیٹا کی تشریح کرنا (Interpreting Data)
- **مرحلہ 6:** نتائج کو بصری شکل دینا اور ابلاغ کرنا / ویژولائیزنگ اور کمیونیکیٹنگ رزلٹ

مسئلہ کو سمجھنا (Understanding the Problem)

اس سے پہلے کہ ہم ڈیٹا اکٹھا کرنے کے بارے میں بات کرنا شروع کریں، زیرِ غور مسئلہ کو واضح طور پر سمجھنا ضروری ہے۔ مسئلہ کو سمجھنا DSLC میں سب سے اہم مرحلہ ہے۔ اگر ہم مسئلہ کو صحیح طریقے سے نہیں سمجھتے، تو ہم غلط ڈیٹا اکٹھا کر سکتے ہیں جس کے نتیجے میں غیر مفید یا غلط نتائج برآمد ہوں گے۔

اس ابتدائی مرحلے میں، ہمیں کچھ سوالات (کیا، کیوں، کون) پوچھنے کی ضرورت ہوتی ہے، جیسے کہ:

- ہم کیا معلوم کرنے کی کوشش کر رہے ہیں؟
- یہ مسئلہ کیوں اہم ہے؟
- ان نتائج کو کون استعمال کرے گا؟

مندرجہ ذیل ذیلی حصے DSLC کے پہلے مرحلے کا مزید تفصیل سے احاطہ کرتے ہیں۔

مسئلہ کی شناخت اور تعریف کرنا

سب سے پہلے، ہمیں مسئلے کی شناخت کرنے کی ضرورت ہے جسے ہم ڈیٹا کا استعمال کرتے ہوئے حل کرنا چاہتے ہیں۔ یہ مسئلہ ہمارے اسکول، گھر، شہر، یا زندگی کے کسی بھی شعبے سے متعلق ہو سکتا ہے جہاں معلومات کی بنیاد پر فیصلے کیے جاتے ہیں۔ ایک بار جب کسی مسئلے کی شناخت ہو جائے، تو ہمیں اسے واضح اور سادہ طریقے سے بیان کرنا چاہیے۔

مثال: صرف یہ کہنے کے بجائے کہ، "طالب علم خوش نہیں ہیں،" ہم مسئلہ کو بہتر طریقے سے یوں بیان کرسکتے ہیں:
"طالب علموں کو اسکول کی کون سی سرگرمیوں میں سب سے زیادہ لطف آتا ہے؟"
کسی مسئلے کی تعریف کرتے وقت، ہمیں مندرجہ ذیل باتوں کو مدنظر رکھنے کی ضرورت ہے:

i) واضح اور سادہ الفاظ کا استعمال کریں،
ii) ایک مخصوص سوال پر توجہ مرکوز کریں، اور
iii) غیر واضح یا عام شرائط سے گریز کریں۔

کیا آپ جانتے ہیں؟
ایک اچھا مسئلہ وہ ہے جس کا جواب ڈیٹا اکٹھا کرنے اور تجزیہ کرنے سے دیا جاسکے۔

واضح مقاصد کا تعین کرنا

ایک بار جب ہم مسئلے کی شناخت اور تعین کر لیتے ہیں، تو ہمیں یہ فیصلہ کرنا چاہیے کہ ہم کیا حاصل کرنا چاہتے ہیں۔ انہیں ہمارے مقاصد (Objectives) کہا جاتا ہے جو ڈیٹا سائنس کے باقی عمل میں رہنمائی کرتے ہیں۔ وہ بتاتے ہیں کہ:

I) ہمیں کس قسم کے ڈیٹا کی ضرورت ہے؟
ii) ہم کیا سوالات پوچھیں گے؟
iii) ہمیں کن نتائج کی تلاش ہے؟

مثال: اوپر بیان کردہ مسئلہ "طالب علموں کو اسکول کی کون سی سرگرمیوں میں سب سے زیادہ لطف آتا ہے؟" کے لیے، ہم اپنے مقاصد کو درج ذیل طریقے سے تشکیل دے سکتے ہیں:

- معلوم کریں کہ ہر سرگرمی میں کتنے طالب علم حصہ لیتے ہیں؟
- معلوم کریں کہ کون سی سرگرمی کو طالب علموں کی طرف سے سب سے زیادہ دلچسپ قرار دیا جاتا ہے؟
- سفارش کریں کہ کن سرگرمیوں کو بہتر یا تبدیل کیا جانا چاہیے؟

کیا آپ جانتے ہیں؟
اپنے مقاصد کو ایک چیک لسٹ کی طرح لکھیں۔ یہ آپ کے پراجیکٹ کو مرکوز رکھنے میں مدد کرتا ہے۔

مثالیں: اسکول، صحت، کاروبار سے متعلق مسائل اور مقاصد

یہ ذیلی سیکشن تصور کو بہتر طریقے سے سمجھنے کے لیے کچھ حقیقت پسندانہ مثالوں کا احاطہ کرتا ہے۔ جدول 5.1 مختلف ڈومینز جیسے اسکول، صحت اور کاروبار سے متعلق مثال کے مسائل اور ان کے مقاصد فراہم کرتا ہے۔

مثال: مسائل اور مقاصد

یہ جدول مختلف شعبوں میں ڈیٹا سائنس کے اطلاق کے لیے حقیقی مسائل کی مثالیں اور ان سے متعلقہ واضح مقاصد کی وضاحت کرتا ہے۔

جدول:5.1

ڈومین	مثالی مسئلہ	مقاصد
سکول	بہت سے طالب علم اسکول دیر سے پہنچتے ہیں۔	ایک مہینے میں کتنے طالب علم دیر سے پہنچتے ہیں،اس کا پتہ لگانا۔
		دیر سے آنے کی وجوہات کی شناخت کرنا (مثلاً، ٹرانسپورٹ، فاصلہ، ہنگامی صورتحال،یاموسمی مسائل)۔
		دیر سے آنے والے کیسز کو کم کرنے کے لیے حل تجویز کرنا۔
صحت	سکول کے اوقات کے دوران بہت سے طالب علم کافی زیادہ پانی نہیں پیتے ہیں۔	طالب علم روزانہ کتنا پانی پیتے ہیں،اس پر ڈیٹا اکٹھا کرنا۔
		پانی کی کم مقدار پینے کی وجوہات معلوم کرنا۔
		صحت مند عادات کو فروغ دینے کے لیے اقدامات کی سفارش کرنا۔
کاروبار	سکول کی کینٹین یہ جاننا چاہتی ہے کہ طالب علم کون سے سنیکس سب سے زیادہ پسند کرتے ہیں۔	طالب علموں کے پسندیدہ اسنیکس کے بارے میں سروے کرنا۔
		مقبولیت اور فروخت کے ڈیٹا کا موازنہ کرنا۔
		کینٹین مینیجر کو سمارٹ آرڈرنگ کے فیصلے کرنے میں مدد فراہم کرنا۔

یادرکھیں:

1. کسی بھی ڈیٹا سائنس پراجیکٹ کا پہلا مرحلہ مسئلہ کو سمجھنا ہے۔
2. ہمیں مسئلہ کو واضح طور پر بیان کرنا چاہیے اور مخصوص اہداف کا تعین کرنا چاہیے۔
3. کلیئر (Clear) مسائل وہ ہوتے ہیں جنہیں ڈیٹا کا استعمال کر کے حل کیا جاسکتا ہے۔
4. واضح مقاصد صحیح ڈیٹا اکٹھا کرنے اور مفید نتائج تک پہنچنے میں مدد کرتے ہیں۔

ڈیٹا اکٹھا کرنا

ایک بار جب ہم مسئلہ کو سمجھ لیتے ہیں اور واضح اہداف مقرر کر لیتے ہیں، تو DSLC میں اگلا مرحلہ مطلوبہ ڈیٹا اکٹھا کرنا ہے۔ ڈیٹا خام مال (raw material) کی طرح ہے، اور ہمیں اسے کلین کرنے، ذخیرہ کرنے یا تجزیہ کرنے سے پہلے صحیح قسم کا ڈیٹا جمع کرنا چاہیے۔

مرحلہ 1 : ہم نے سوالات پوچھے جیسے "طالب علموں کو اسکول کی کون سی سرگرمیوں میں سب سے زیادہ لطف آتا ہے؟" یا "طالب علم سکول دیر سے کیوں آرہے ہیں؟"

مرحلہ 2: ان سوالات کے جواب دینے میں مدد فراہم کرنے والی معلومات اکٹھا کرنے کے بارے میں ہے۔

ہم مندرجہ ذیل چیزوں کا جائزہ لیں گے:

i. ڈیٹا اکٹھا کرنے کے طریقے
ii. ڈیٹا کہاں سے اکٹھا کیا جائے (ذرائع)؟
iii. یہ کیسے یقینی بنایا جائے کہ یہ ڈیٹا مفید اور منصفانہ ہے (ڈیٹا کا معیار)؟

(i) ڈیٹا اکٹھا کرنے کے طریقے

مسئلے اور دستیاب ٹولز پر منحصر، ڈیٹا اکٹھا کرنے کے کئی طریقے ہیں۔ آیئے اسکولوں، گھروں اور کاروباروں میں استعمال ہونے والے سب سے عام طریقوں پر بات کرتے ہیں۔

سروے (آن لائن / پیپر بیسڈ): سروے سب سے آسان اور عام طریقہ ہے۔ ہم لوگوں سے سوالات پوچھتے ہیں اور ان کے جوابات ریکارڈ کرتے ہیں۔

مثال: یہ جاننے کے لیے کہ طالب علم کن کھیلوں سے لطف اندوز ہوتے ہیں، ہم سوالات کے ساتھ ایک سروے بنا سکتے ہیں۔ جیسے

I) آپ کو کون سا کھیل سب سے زیادہ پسند ہے؟
ii) آپ اسے کتنی بار کھیلتے ہیں؟
iii) آپ دن کے کس وقت کھیلنا پسند کرتے ہیں؟

کیا آپ جانتے ہیں؟

سروے گوگل فارمز (آن لائن) اور پرنٹ شدہ کاغذی فارمز (مینوئل) کا استعمال کرتے ہوئے کیے جاسکتے ہیں۔

- APIs اور آن لائن ذرائع: ایپلیکیشن پروگرامنگ انٹرفیس (APIs) سافٹ ویئر کو ویب سائٹس یا ایپلی کیشنز (مختصراً ایپس) سے خودکار طور پر ڈیٹا اکٹھا کرنے کی اجازت دیتے ہیں۔
مثال: ایک اسکول روزانہ کی حاضری کا ڈیٹا کی رپورٹ بنانے کے لیے حاضری کا API استعمال کر سکتا ہے (یہ ڈویلپرز زیادہ استعمال کرتے ہیں)۔
- IoT اور سینسر ڈیٹا: انٹرنیٹ آف تھنگز (IoT) ڈیوائسز، جیسے سمارٹ واچز، واٹر ڈسپنسر، یا درجہ حرارت کے سینسر، خود بخود ڈیٹا اکٹھا کر سکتے ہیں۔
مثال: پانی پینے کی عادات کا مطالعہ کرنے کے لیے، سکول پانی کے فواروں (پوائنٹس) کے قریب سمارٹ واٹر میٹر لگا سکتے ہیں۔

ویب سکریپنگ (Web Scraping):

SMART TIPS
سمارٹ ٹپس
IoT ڈیوائسز صحت، حفاظت، اور ماحولیاتی نگرانی میں مفید ہیں۔

یہ ویب سائٹس سے عوامی ڈیٹا کو خودکار طور پر اکٹھا کرنے کا ایک طریقہ ہے۔
مثال: ایک کاروباری طالب علم سکول کی کینٹین کی قیمتوں سے موازنہ کرنے کے لیے آن لائن اسٹورز سے سنیکس کی قیمتیں اکٹھی کر سکتا ہے۔

SMART TIPS
سمارٹ ٹپس
ویب سکریپنگ صرف اساتذہ کی رہنمائی میں اور ان ویب سائٹس کے لیے کی جانی چاہیے جو اس کی اجازت دیتی ہیں۔

(ii) درست ڈیٹا سورس کا انتخاب

ہر قسم کا ڈیٹا مفید نہیں ہوتا۔ ہمیں قابل اعتماد، درست اور متعلقہ ذرائع کا انتخاب کرنا چاہیے۔
ڈیٹا سورس کا انتخاب کرتے وقت، چند سوالات پوچھیں جیسے:

i) کیا یہ ڈیٹا ایک بھروسہ مند جگہ سے ہے؟
ii) کیا یہ حالیہ ہے یا پرانا؟
iii) کیا یہ اس مسئلے سے مطابقت رکھتا ہے جسے میں حل کرنا چاہتا ہوں؟

مثال: اگر ہم 2025 میں طالب علموں کے پسندیدہ سنیکس جاننا چاہتے ہیں، تو ہمیں 2020 میں کیے گئے سروے کا استعمال کرنے سے گریز کرنا چاہیے۔

(iii ڈیٹا کے معیار اور جانبداری کا جائزہ لینا Assessing Data Quality and Bias

ڈیٹا کا معیار کا مطلب ہے کہ ڈیٹا

(i مکمل ہے (کوئی غیر موجود ڈیٹا نہیں)،

(ii درست ہے (کوئی غلطیاں نہیں)،

(iii متعلقہ ہے (ہمارے مسئلے سے مطابقت رکھتا ہے)۔

SMART TIPS سمارٹ ٹپس

ہمیشہ جدید اور مسئلے سے متعلق ڈیٹا اکٹھا کریں۔

جانبداری (Bias) کا مطلب ہے کہ ڈیٹا صرف حقیقت کا ایک حصہ دکھاتا ہے اور پوری تصویر کو نظر انداز کر دیتا ہے۔

مثال: اگر ہم صرف لڑکوں کا ان کے پسندیدہ کھیلوں کے بارے میں سروے کرتے ہیں، تو ہم لڑکیوں کی رائے سے محروم ہو جائیں گے۔ نتائج غیر منصفانہ یا لڑکوں کی طرف جانبدار ہوں گے۔

جانبداری سے بچنے کے لیے درج ذیل باتوں کو یاد رکھیں:

- لوگوں کے مختلف گروپس سے پوچھیں۔
- سوالات کو غیر جانبدار رکھیں اور رہنمائی کرنے والے (لیڈنگ) نہ ہوں۔
- جوابات میں عجیب و غریب پیٹرنز کی جانچ کریں۔

سرگرمی:

ایک سادہ سروے ڈیزائن کرنا

مقصد: اپنے ہم جماعتوں کی فارغ وقت کی عادات کے بارے میں ایک مختصر سروے بنا کر ڈیٹا اکٹھا کرنے اور ذخیرہ کرنے کا طریقہ سیکھیں۔

ہدایات:

1. **اپنا سروے ڈیزائن کریں:** اپنے ہم جماعتوں سے پوچھنے کے لیے 3 سے 5 سوالات سوچیں۔
2. **جوابات اکٹھا کرنے کی تیاری کریں:** ایک طریقہ کا انتخاب کریں (مثلاً، کاغذی فارم یا گوگل فارم)۔
3. **ڈیٹا اکٹھا کریں:** کم از کم 5 ہم جماعتوں سے اپنا سروے مکمل کرنے کو کہیں اور ان کے جوابات کو واضح طور پر لکھیں۔
4. **ڈیٹا کو ذخیرہ کریں:** (i) اگر آپ کاغذ استعمال کر رہے ہیں، تو اسے صاف ستھرا اور کالموں میں منظم رکھیں۔

   (ii) اگر ڈیجیٹل ٹولز استعمال کر رہے ہیں، تو فائل کو ایک واضح نام جیسے FreeTime_your name.xlsx سے محفوظ کریں۔
5. (اختیاری طور پر) اپنے استاد یا ہم جماعتوں کو دکھائیں کہ آپ نے ڈیٹا کیسے اکٹھا اور ذخیرہ کیا۔

MEMORY BOX یاد رکھیں:

1. کسی مسئلے کی تعریف کرنے کے بعد، ہمیں وہ ڈیٹا اکٹھا کرنے کی ضرورت ہے جو مسئلہ حل کرنے میں مدد کر سکے۔
2. ڈیٹا اکٹھا کرنے کے مختلف طریقے ہیں جیسے سروے، APIs، سینسرز، اور ویب سکریپنگ۔
3. اگلے مرحلے پر جانے سے پہلے ہمیں ہمیشہ ڈیٹا کے معیار اور غیر جانبداری جانچنا چاہیے۔
4. وہ ڈیٹا جو نامکمل یا متعصب ہو، غلط فیصلوں کا باعث بن سکتا ہے۔

ڈیٹا کلیننگ اور تصدیق (ویلیڈیٹنگ)

ڈیٹا اکٹھا کرنے کے بعد، اگلا اہم مرحلہ اسے کلین کرنا اور تصدیق کرنا ہے۔ خام ڈیٹا بے ترتیب ہوتا ہے۔ اس میں غلطیاں، غائب ویلیوز، یا یہاں تک کہ غلط اندراجات بھی شامل ہو سکتے ہیں۔ اگر ہم اسے جانچے بغیر استعمال کرتے ہیں، تو ہمارا تجزیہ غلط نتائج دے گا۔
یاد رہے کہ اس سے پہلے ہم نے سروے اور دیگر طریقوں سے ڈیٹا اکٹھا کیا تھا۔ اب، ہمیں یہ جانچنے کی ضرورت ہے کہ آیا وہ ڈیٹا درست، مکمل اور استعمال کے لیے تیار ہے۔ یہ مرحلہ یقینی بناتا ہے کہ جس ڈیٹا کا ہم بعد میں تجزیہ کریں گے، وہ قابل اعتماد ہے۔

خام ڈیٹا (Raw Data)

خام ڈیٹا اصل معلومات ہے جو ہم لوگوں، سینسرز، ویب سائٹس، یا ایپس سے اکٹھا کرتے ہیں۔ اسے ابھی تک جانچا یا کلین نہیں کیا گیا ہے۔
مثال کے طور پر، فرض کریں کہ 20 طالب علموں نے آپ کے پسندیدہ کھیلوں کے سروے کو پُر کیا۔ کچھ اندراجات جدول میں دکھائے گئے ہیں:

جدول 5.2: طالب علموں کے پسندیدہ کھیل سے متعلق خام ڈیٹا

سیریل نمبر	نام	پسندیدہ کھیل	فریکوئینسی (Frequency)	ایک دن میں پڑھنے کا وقت
1	Ahmad	cricket	Daily	Evening
2	Ali		Weekly	Afternoon
3	Sara	Footbal	Daily	
4	Sana	Football	Daliy	Evening

جدول 5.2 میں مندرجہ ذیل چیزیں شامل ہیں:

(i) سپیلنگ کی غلطیاں : (Spelling Errors) مثلاً، Footbal سیریل نمبر 3 اور Daliy سیریل نمبر 4۔

(ii) (Missing Entries): میسنگ اینٹریز سیریل نمبر 2 اور 3 میں خالی خانے موجود ہیں۔

مکسڈ کیپیٹلائزیشن : (Mixed Capitalizations) مثلاً، cricket سیریل نمبر 1 اور Football سیریل نمبر 4۔ اس طرح کے ڈیٹا کو کلیننگ (Cleaning) کی ضرورت ہوتی ہے۔

عمومی ڈیٹا ایررز (غائب، نقل ڈپلیکیٹ، غیر معمولی ویلیوز)

جدول 5.3 خام ڈیٹا میں پائے جانے والے کچھ عام مسائل کو ظاہر کرتا ہے۔

جدول: 5.3 خام ڈیٹا میں عام مسائل

سیریل نمبر	غلطی کی قسم	مثال
1	مِسنگ ڈیٹا	خالی خانے (مثلاً، کھیل کا نام یا وقت درج نہیں)۔
2	ڈوپلیکیٹ اندراجات	ایک ہی طالب علم کو دو بار درج کیا جانا۔
3	غیر مستحکم فارمیٹس (Inconsistent formats)	"Cricket، cricket"، "CRICKET" (حروف کی کیپیٹلائزیشن میں فرق)۔
4	ہجوں کی غلطیاں (Spelling mistakes)	"Football" کی بجائے "Footbal" یا "Daily" کی بجائے "Daliy"۔
5	غیر معمولی ویلیوز (Outliers) (عجیب و غریب اقدار)	فریکئیونسی = 1000 بار فی ہفتہ۔

ڈیٹا کی تصدیق کی تکنیکیں (Data Validation Techniques)

ڈیٹا کی تصدیق کا مطلب یہ جانچنا ہے کہ آیا ڈیٹا:

i. درست قواعد کی پیروی کرتا ہے (مثلاً، صرف درست کھیلوں کے نام)،

ii. ویلیوز کی صحیح قسم رکھتا ہے (مثلاً، جہاں نمبرز کی ضرورت ہو وہاں نمبرز ہوں)، اور

iii. فیک یا غیر سنجیدہ (fake or silly) نہیں ہے۔

مثال کے طور پر، ویلیڈیشن رولز:

"پسندیدہ کھیل" کے لیے صرف ان ویلیوز میں سے کسی ایک کی اجازت دیں: فٹ بال، کرکٹ، بیڈ منٹن، ہاکی۔

سروے میں ویلیڈیشن: اس مقصد کے لیے، مندرجہ ذیل باتوں پر غور کریں:

- خالی ٹیکسٹ باکسز کے بجائے کثیر الانتخابی سوالات کا استعمال کریں۔
- اہم سوالات کو لازمی قرار دیں، تاکہ کوئی بھی انہیں نظر انداز نہ کرے۔
- ہجوں کی غلطیوں کو کم کرنے کے لیے ڈراپ ڈاؤنز یا چیک باکسز استعمال کریں۔

SMART TIPS

سمارٹ ٹپس

گوگل شیٹس، ایکسل، یا آن لائن فارمز جیسے ٹولز میں مہیا کئے گئے سادہ قواعد بھی ڈیٹا کی تصدیق میں مدد کر سکتے ہیں۔

ڈیٹا کلیننگ کے طریقے

ایک بار جب ہمیں غلطیاں مل جاتی ہیں، تو ہمیں انہیں درست کرنے یا ہٹانے کی ضرورت ہوتی ہے۔ اسے ڈیٹا کلیننگ کہتے ہیں۔

ڈیٹا کلین کرنے کے لیے چند بنیادی اقدامات درج ذیل ہیں:

- سپیلنگ کی غلطیاں درست کریں: "Footbal" → "Football"
- فارمیٹس کو معیاری بنائیں: "CRICKET" → "Cricket"
- نقل کو ہٹائیں (تکرار-Redundancy): ایک ہی طالب علم کے لیے بار بار آنے والے اندراجات کو حذف کر دیں۔
- گمشدہ اقدار کو ہینڈل کرنا: اس شخص سے دوبارہ پوچھیں یا اس قطار (row) کو ہٹا دیں۔
- غلط ڈیٹا کو درست کریں: اگر فریکوئینسی (frequency) "دس سو" (ten hundred) کے طور پر لکھا گیا ہے، تو اسے درست کریں یا ہٹا دیں۔

ڈیٹا کلیننگ کیوں ضروری ہے؟

فرض کریں کہ ہم اس مرحلے کو چھوڑ دیتے ہیں اور ایسے ڈیٹا کا تجزیہ کرتے ہیں جس میں شامل ہیں:

- سپیلنگ کی غلطیوں والے کھیل ("Cricket")
- گمشدہ اوقات / سٹڈی ٹائم
- ڈوپلیکیٹ اندراجات

ہمارا نتیجہ شاید کہے:

"فٹ بال: 3 ووٹ، cricket: 1 ووٹ، Cricket: 2 ووٹ"

یہ غلط تصویر دیتا ہے، کیونکہ "cricket" اور "Cricket" ایک ہی کھیل ہیں!

خلاصہ: ہم کہہ سکتے ہیں کہ کلین ڈیٹا سمارٹ فیصلوں کی طرف لے جاتا ہے۔ ڈرٹی (Dirty) ڈیٹا الجھن کا باعث بنتا ہے۔

ڈیٹا کا تجزیہ کرنا

اب جب کہ ہمارا ڈیٹا صحیح طریقے سے اکٹھا اور کلین ہو چکا ہے، ہم تجزیہ کے مرحلے کی طرف بڑھ سکتے ہیں۔ یہ وہ جگہ ہے جہاں ہم پیٹرن (patterns)، رجحانات (trends)، اور سوالات کے جوابات تلاش کرنے کے لیے ڈیٹا کا جائزہ لیتے ہیں جو ہم نے مرحلہ 1 میں پوچھے تھے۔ فرض کریں کہ آپ کے پاس ڈیٹا کا ایک کلین اور منظم جدول ہے، مثال کے طور پر، طالب علموں کے پسندیدہ کھیل یا پڑھائی کے اوقات کے بارے میں جوابات۔ اب آپ پوچھ سکتے ہیں، "یہ ڈیٹا ہمیں کیا بتا رہا ہے؟" یہ مرحلہ ہمیں ڈیٹا میں جو کچھ بھی ہم دریافت کرتے ہیں، اس کی بنیاد پر فیصلے کرنے میں مدد کرتا ہے۔

ڈیٹا کا تجزیہ کرنے کا کیا مطلب ہے؟

ڈیٹا کا تجزیہ کرنے کا مطلب ہے احتیاط سے ڈیٹا کی جانچ کرنا تاکہ:

(i) اس کا خلاصہ کیا جائے (مثلاً، کتنے طالب علموں نے ہر آپشن کا انتخاب کیا)،

(ii) مختلف گروپوں کا موازنہ کیا جائے (مثلاً، لڑکے بمقابلہ لڑکیاں صبح بمقابلہ شام)،

(iii) نمونے تلاش کیے جائیں (مثلاً، زیادہ تر طالب علم فٹ بال کو ترجیح دیتے ہیں اور شام کو کھیلتے ہیں)۔

آپ شمار (counts-کتنے؟)، اوسط (averages)، فریکوئنسی (frequency-کتنی بار؟) وغیرہ جیسے پیمانوں کا استعمال کر سکتے ہیں۔

تجزیہ کے لیے سادہ ٹولز کا استعمال

آپ سادہ ٹولز جیسے گوگل شیٹس، مائیکروسافٹ ایکسل، یا نوٹ بک میں لکھتے ہوئے ڈیٹا کا تجزیہ شروع کر سکتے ہیں۔

مثال: آپ نے 20 طالب علموں کا سروے کیا۔ جدول 5.4 دکھاتا ہے کہ آپ کو کیا ملا ہو گا۔

اس ڈیٹا سے، ہم یہ مشاہدہ کر سکتے ہیں کہ زیادہ تر طالب علم کرکٹ کو ترجیح دیتے ہیں۔

جدول 5.4: طالب علم کے سروے سے ماخوذ ڈیٹا

پسندیدہ کھیل	فریکیونسی
کرکٹ	10
فٹ بال	5
ہاکی	4
بیڈ منٹن	1

سرگرمی: ڈیٹا کلیننگ

مقصد: ڈیٹا انٹری کی عام غلطیوں (جیسے ہجوں کی غلطیاں، Null اقدار، اور غیر مستحکم فارمیٹنگ) کی شناخت اور درستگی کا طریقہ سیکھنا۔

ہدایات:

1. بے ترتیب جدول کا جائزہ:

ذیل میں طالب علموں کے پسندیدہ کھیل سے متعلق خام ڈیٹا کا جدول دیا گیا ہے جس میں غلطیاں موجود ہیں۔

جدول 5.5: طالب علموں کے پسندیدہ کھیل سے متعلق خام ڈیٹا (بے ترتیب ڈیٹا)

سیریل نمبر	نام	پسندیدہ کھیل	فریکئونسی (Frequency)	دن کا وقت
1	Sadiq	cricket	Daliy	Evening
2	Noor	Cricket	Weekly	
3	Maryam	Footbal		morning
4	Zara	Football	Daliy	Afternoon

.2 گروپس (pair) میں کام کریں

اپنے ساتھی کے ساتھ بیٹھیں اور جدول کا ایک ساتھ مطالعہ کریں۔

.3 غلطیوں کی نشاندہی اور فہرست سازی:

کم از کم 5 غلطیاں تلاش کریں۔ ان میں شامل ہو سکتی ہیں:

- سپیلنگ کی غلطیاں: مثلاً، Daliy، یا ممکنہ طور پر Cricket۔
- NULL ڈیٹا: مثلاً، خالی خانے۔
- غیر مستحکم فارمیٹنگ: مثلاً، Football بمقابلہ football، یا morning کو کیپیٹلائزنہ کرنا۔

4۔ غلطیوں کی درستگی:

i) ہر غلطی کو منطقی اور منصفانہ طریقے سے ٹھیک کرنے پر تبادلہ خیال کریں۔

ii) کلین شدہ جدول کو اپنی نوٹ بک میں دوبارہ لکھیں۔

.5 کلاس میں اشتراک (اختیاری)

I) اپنے کلین شدہ جدول کا دوسرے گروپ کے ساتھ موازنہ کریں۔

ii) دیکھیں کہ آیا کہ انہوں نے بھی وہی غلطیاں تلاش کیں یا کچھ مختلف غلطیاں۔

MEMORY BOX

یاد رکھیں:

.1 ڈیٹا کا تجزیہ ہمیں نمونے تلاش کرنے اور فیصلے کرنے میں مدد کرتا ہے۔

.2 ڈیٹا کو سمجھنے کے لیے ہم گنتی (counts) اور موازنہ (comparisons) کا استعمال کرتے ہیں۔

.3 یہاں تک کہ سادہ ٹولز جیسے جدول (Table) یا اسپریڈ شیٹس بھی ڈیٹا کا تجزیہ کرنے میں ہماری مدد کر سکتے ہیں۔

سرگرمی:

ڈیٹا کا تجزیہ (Data Analysis)

مقصد: ٹرینڈ کی شناخت اور فریکیونسی جدول سے بنیادی نتائج اخذ کرنے کے ذریعے ڈیٹا کا تجزیہ کرنے کا طریقہ سیکھنا۔

ہدایات:

اپنا ذاتی سروے ڈیٹا (اگر دستیاب ہو) یا ذیل میں جدول 5.6 میں دیا گیا نمونہ ڈیٹا استعمال کریں۔

احتیاط سے اس جدول کو پڑھیں جو ظاہر کرتا ہے کہ کتنے طالب علم ہر کھیل کو پسند کرتے ہیں۔

جدول 5.6: سروے ڈیٹا سے ماخوذ نمونہ ڈیٹا

کھیل	فریکیونسی یا طلباء کی تعداد
کرکٹ	7
فٹ بال	6
ہاکی	5
بیڈ منٹن	2

شناخت کریں:

i) سب سے زیادہ مقبول کھیل

ii) سب سے کم مقبول کھیل

ایک وجہ لکھیں کہ ہر کھیل آپ کی کلاس میں سب سے زیادہ یا سب سے کم مقبول کیوں ہو سکتا ہے۔

اپنے نتائج کو سادہ جملوں کا استعمال کرتے ہوئے اپنی نوٹ بک میں پیش کریں۔

انٹرپریٹنگ ڈیٹا

اس مرحلے کا مقصد ڈیٹا سے معنی تلاش کرنا ہے (یعنی رجحانات اور پیٹرنز کی شناخت کرنا)۔ ڈیٹا کا خلاصہ کرنے کے بعد، ہم خود سے یہ سوال پوچھ سکتے ہیں:

- یہ نمبرز ہمیں کیا بتا رہے ہیں؟
- کچھ آپشنز زیادہ مقبول کیوں ہیں؟
- کیا کوئی حیرت انگیز بات یا کوئی واضح پیٹرن ہے؟

مثال: اگر ہم یہ معلوم کرنا چاہتے ہیں کہ دیر سے آنے والے زیادہ تر طالب علم اسکول سے دور رہتے ہیں، تو ہم ایک حل کے طور پر نقل و حمل (transport) کی مدد فراہم کرنے کی سفارش کر سکتے ہیں۔

ویژولائزیشن اور ابلاغ کے نتائج

ڈیٹا کا تجزیہ کرنے کے بعد، آخری مرحلہ نتائج کو اس طرح سے شیئر کرنا ہے کہ دوسرے لوگ اسے آسانی سے سمجھ سکیں۔ لوگ عام طور پر نمبروں کے مقابلے میں تصاویر کو زیادہ بہتر سمجھتے ہیں، اسی لیے ہم ڈیٹا ویژولائزیشن کا استعمال کرتے ہیں۔

یاد رہے کہ پچھلے مرحلے میں، ہم نے پایا کہ کرکٹ سب سے زیادہ مقبول کھیل تھا۔ اس مرحلے میں، ہم ایک ایسا چارٹ بنائیں گے جو اسے واضح طور پر ظاہر کرے گا۔ یہ ہمارے نتائج کو ہم جماعتوں، اساتذہ یا دوسروں کے سامنے پیش کرنا آسان بنا دیتا ہے۔

ڈیٹا ویژولائزیشن کی اہمیت

ڈیٹا ویژولائزیشن اہم ہے کیونکہ یہ:

i) ڈیٹا کو آسانی سے سمجھنے کے قابل بناتا ہے،

ii) پیٹرنز کو ایک نظر میں دکھاتا ہے،

iii) پریزنٹیشنز یا رپورٹس میں خیالات کو واضح طور پر شیئر کرنے میں مدد کرتا ہے۔

مثال: یہ کہنے کے بجائے کہ "10 طالب علم کرکٹ پسند کرتے ہیں،" ہم اسے بار چارٹ میں دکھا سکتے ہیں۔

چارٹس اور گرافس کی عام اقسام (بار چارٹ، پائی چارٹ، لائن چارٹ وغیرہ)

ہمیشہ ایک ایسا چارٹ منتخب کریں جو آپ کے ڈیٹا سے مطابقت رکھتا ہو۔ جدول 5.7 عام طور پر استعمال ہونے والے چارٹس کی اقسام، ان کے استعمال کی صورتحال اور مثالیں دکھاتا ہے۔

جدول 5.7: چارٹس کی عام اقسام

چارٹ / گراف کی قسم	استعمال کا مقصد	مثال
بار چارٹ	گروپس کا موازنہ کرنا	ہر کھیل کو پسند کرنے والے طالب علموں کی تعداد۔
پائی چارٹ	پوری چیز کے حصے دکھانا	پسندیدہ اسنیکس کا فیصد۔
لائن چارٹ	وقت کے ساتھ تبدیلی دکھانا	ہر ہفتے دیر سے آنے والے طالب علموں کی تعداد۔

SMART TIPS

سمارٹ ٹپس

اپنے چارٹس کو سادہ، کلین اور درست طریقے سے لیبل کریں۔

کمیونیکیٹنگ رزلٹ:

ایک بار جب آپ کا چارٹ تیار ہو جائے تو، اپنے نتائج کی وضاحت کریں۔ آپ 2 سے 3 سطریں لکھ سکتے ہیں:

مثال: "زیادہ تر طالب علم 10 کرکٹ پسند کرتے ہیں، جس کے بعد فٹ بال 6 آتا ہے۔ بہت کم طالب علم 1 بیڈ منٹن کو پسند کرتے ہیں۔"

SMART TIPS

سمارٹ ٹپس

ڈیٹا کی کہانی سنانے کے لیے بصری معاونتوں اور الفاظ کو ایک ساتھ استعمال کریں (واضح، مختصر اور مفید)۔

MEMORY BOX

یاد رکھیں:

1. ڈیٹا ویژولائیزیشن کا مطلب نتائج دکھانے کے لیے چارٹس اور گرافس کا استعمال کرنا ہے۔
2. یہ ڈیٹا کو سمجھنے اور شیئر کرنے میں آسان بناتا ہے۔
3. اپنے ڈیٹا کی قسم کے لیے صحیح چارٹ استعمال کریں۔
4. اپنی بصری معاونتوں کو ہمیشہ لیبل لگائیں اور وضاحت کریں۔

5.3 ویژولائزیشن کے لیے ٹولز

مرحلہ 5 جیسے ہی آپ جان لیں کہ کس قسم کا چارٹ استعمال کرنا ہے۔ اگلا مرحلہ اسے مختلف ٹولز کی مدد سے تیار کرنا ہوتا ہے۔ مرحلہ 6 میں، ہم نے سیکھا کہ کس طرح چارٹس ڈیٹا کو سمجھنے اور پیش کرنے میں آسانی پیدا کرتے ہیں۔ اس سیکشن میں، آپ سیکھیں گے کہ سافٹ ویئر کا استعمال کرتے ہوئے درحقیقت وہ چارٹس کیسے بنائے جاتے ہیں۔

آیئے سب سے زیادہ مفید اور ابتدائی افراد کے لیے دوستانہ ٹولز کا جائزہ لیتے ہیں۔

اسپریڈ شیٹس (Excel/Google Sheets) میں چارٹس بنانا

گوگل شیٹس (Google Sheets) اور مائیکروسافٹ ایکسل (Microsoft Excel) ٹیبلز سے چارٹس اور گرافس بنانے کے لیے سادہ مگر طاقتور ٹولز ہیں۔

گوگل شیٹس میں چارٹ بنانے کے اقدامات:

i. اپنا ڈیٹا قطاروں (rows) اور کالموں (columns) میں درج کریں۔

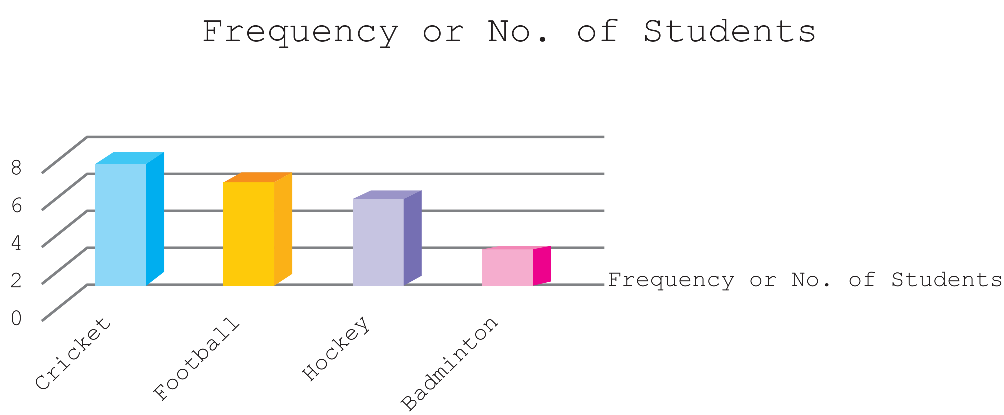
ii. ڈیٹا کی حد (data range) کو منتخب کریں۔

iii. "Insert" مینیو پر کلک کریں پھر "Chart" کو منتخب کریں۔

iv. چارٹ کی قسم (بار، پائی، لائن) کا انتخاب کریں۔

v. چارٹ کو ایک عنوان (title)، لیبل (label)، اور رنگ کے ساتھ حسب ضرورت (Customize) بنائیں۔

مثال: فرض کریں کہ آپ کے پاس گوگل شیٹس میں ڈیٹا ہے (براہ کرم جدول 5.6 سے رجوع کریں)۔ آپ کی حوصلہ افزائی کی جاتی ہے کہ آپ جدول 5.6 کے ڈیٹا کو بار چارٹ میں تبدیل کرنے کے لیے ان اقدامات کا استعمال کریں۔



شکل 5.1: طالب علموں کے پسندیدہ کھیل (جدول 5.6 کے ڈیٹا پر مبنی)

ہم اسپریڈ شیٹس کا استعمال اس لیے کرتے ہیں کیونکہ وہ:

i. استعمال میں آسان ہیں،
ii. انسٹالیشن کی ضرورت نہیں (گوگل شیٹس کے لیے)،
iii. چھوٹے سے درمیانے سائز کے پراجیکٹس کے لیے بہترین ہیں۔
iv. شکل 5.1: ہمارے جدول 5.6 کے مطابق ایک سادہ بار گراف دکھاتی ہے۔

سرگرمی:

سروے ڈیٹا جدول 5.4 سے بار چارٹ، پائی چارٹ اور لائن چارٹ بنانا

مقصد: یہ سیکھنا کہ سروے ڈیٹا ویژولائیزیشن پیش کرنے کے لیے فریکوئنسی ٹیبل کو بار چارٹ، پائی چارٹ، اور لائن چارٹ میں کیسے تبدیل کیا جاتا ہے۔

ہدایات:

- جدول 5.4 میں دیے گئے نمونہ ڈیٹا کا استعمال کریں۔
- اپنی نوٹ بک میں درج ذیل تینوں چارٹس کو بنائیں (بار چارٹ، پائی چارٹ، لائن چارٹ)۔
  اپنے نتائج کو سادہ جملوں کا استعمال کرتے ہوئے اپنی نوٹ بک میں پیش کریں۔

MEMORY BOX

یاد رکھیں:

1. گوگل شیٹس اور ایکسل چارٹس بنانے کے لیے آسان ٹولز ہیں۔
2. ایک ڈیش بورڈ ایک ہی جگہ پر متعدد ویژوالز (multiple visuals) دکھاتا ہے تاکہ ایک واضح ڈیٹا کہانی بیان کی جا سکے۔

نوٹ: معیاری DSLC چھ اہم مراحل پر مشتمل ہوتا ہے، تاہم، حقیقی دنیا کے کچھ اطلاقات / منظر ناموں میں، کچھ اضافی مراحل بھی شامل کیے جا سکتے ہیں۔ ایسا ہی ایک مرحلہ ڈیٹا کو محفوظ کرنا اور منظم کرنا ہو سکتا ہے، خاص طور پر بڑے منصوبوں یا ٹیم کے ماحول میں جہاں ڈیٹا بیسز اور سٹرکچرڈ سٹوریج اہم ہوتی ہے۔ انہیں عام طور پر DBMS اور SQL جیسے نظاموں کے ذریعے منظم کیا جاتا ہے۔

5.4 ڈیٹا بیس اور ڈیٹا کو محفوظ کرنا

ایک بے ترتیب سروے کے جدول کو کلین کرنے کے بعد، کلین شدہ ڈیٹا کو ایک مناسب فارمیٹ میں (یا تو اسپریڈ شیٹ میں یا ایک ڈیٹا بیس میں) ذخیرہ کرنے کی ضرورت ہوتی ہے، تاکہ یہ منظم، قابل تلاش، اور تجزیہ کے لیے تیار ہو۔ آیئے دیکھتے ہیں کہ ڈیٹا کو کیسے اور کہاں ذخیرہ کیا جاتا ہے۔

سٹرکچرڈ ڈیٹا کو ذخیرہ کرنے کے دو اہم طریقے ہیں:

i. اسپریڈ شیٹس (Spreadsheets)

ii. ڈیٹا بیسز (Databases)

دونوں کے درمیان فرق جدول 5.8 میں فراہم کیا گیا ہے۔

جدول 5.8 اسپریڈ شیٹ بمقابلہ ڈیٹا بیس

طریقہ	وضاحت	مثالی ٹول
اسپریڈ شیٹس	چھوٹے ڈیٹا سیٹس کے لیے قطاروں اور کالموں پر مشتمل ایک گرڈ۔	ایکسل (Excel)، گوگل شیٹس (Google Sheets)
ڈیٹا بیسز	بڑے ڈیٹا کو ٹیبلز میں محفوظ کرنے اور منظم کرنے کے لیے ایک سٹرکچرڈ نظام۔	مائی ایس کیو ایل (MySQL)، ایس کیو لائٹ (SQLite)

SMART TIPS

سمارٹ ٹپس

اسپریڈ شیٹس چھوٹے پراجیکٹس کے لیے اچھی ہیں۔ ڈیٹا بیسز بڑے یا ملٹی یوزر ڈیٹا سسٹمز (multi-user data systems) کے لیے بہتر ہوتے ہیں۔

ڈیٹا بیس کیا ہے؟

ایک ڈیٹا بیس (Database) ڈیٹا کا ایک منظم مجموعہ ہے۔ یہ ڈیٹا کو ٹیبلز کی شکل میں محفوظ کرتا ہے۔ ہر ٹیبل قطاروں (rows) اور کالموں (columns) کے ساتھ ایک چھوٹی اسپریڈ شیٹ کی طرح ہوتا ہے۔ ڈیٹا بیسز ڈیٹا کو مؤثر طریقے سے محفوظ کرنے، بازیافت کرنے، اور اپ ڈیٹ کرنے میں مدد دیتے ہیں۔ جدول 5.9 ڈیٹا بیس سے متعلق اصطلاحات کی وضاحت کرتا ہے۔

جدول 5.9 ڈیٹا بیس اصلاحات: معنی اور مثالیں

اصطلاح	معنی	مثال
ٹیبل	متعلقہ ڈیٹا کا ایک گروپ۔	اسپورٹس سروے
ریکارڈ	ٹیبل میں ایک واحد اندراج (قطار-row)۔	علی کے ڈیٹا کی مکمل قطار۔
فیلڈ	معلومات کا ایک مخصوص حصہ (کالم-column)۔	نام، کھیل، فریکوئنسی

سرگرمی:

ڈیٹا کو منظم کرنا

مقصد: مستقل فارمیٹنگ استعمال کرتے ہوئے ڈیٹا کو منظم اور بہتر ساخت والے جدول میں ترتیب دینا سیکھیں۔

ہدایات: پہلے سے حاصل کیے گئے کلین ڈیٹا کو استعمال کریں۔

- اپنی نوٹ بک پر ایک نیا جدول بنائیں جس کے کالم ہیڈنگز درج ذیل ہوں۔

Frequency, Time of day, Favorite sport Name, Roll No

- کم سے کم 5 قطاروں میں ڈیٹا لکھیں۔
- اپنا ڈیٹا صحیح طریقے سے کالمز میں لکھیں۔
- اس بات کو یقینی بنائیں کہ تمام انٹریز کے صحیح لکھے گئے ہوں اور غلطیوں سے پاک ہوں۔

ڈی بی ایم ایس اور ایس کیو ایل (DBMS and SQL)

ڈیٹا بیس مینجمنٹ سسٹم (DBMS) وہ سافٹ ویئر ہے جو ڈیٹا بیسز کو بنانے، منظم کرنے اور ان کے ساتھ تعامل کرنے کے لیے استعمال ہوتا ہے۔ اس کی مثالوں میں MySQL، SQLite، اور مائیکروسافٹ ایکسیس (Microsoft Access) شامل ہیں۔

ڈیٹا بیس میں ڈیٹا کے ساتھ کام کرنے کے لیے، ہم ایک لینگوئج استعمال کرتے ہیں جسے SQL یا سٹرکچرڈ قیوری لینگوئج (Structured Query Language - SQL) کہا جاتا ہے۔

SQL ہمیں سادہ کمانڈز کا استعمال کرتے ہوئے ڈیٹا کو شامل کرنے (add)، اپ ڈیٹ کرنے، حذف کرنے، یا بازیافت کرنے (retrieve) میں مدد کرتی ہے۔

چند بنیادی SQL قیوریز (Queries) درج ذیل ہیں:

1. SELECT * FROM Sports

یہ سادہ قیوری (Query) Sports نامی ٹیبل سے تمام ڈیٹا دکھاتی ہے۔

2.SELECT Name FROM Sports WHERE Favorite_Sport = 'Cricket'

یہ قیوری Sports ٹیبل میں موجود ان طلباء کے نام دکھاتی ہے جو کرکٹ پسند کرتے ہیں۔

3.INSERT INTO Sports (Name, Favorite_Sport) VALUES ('Ali', 'Football')

یہ قیوری ایک نیا اندراج (entry) شامل کرتی ہے۔

یہ بنیادی SQL اسٹیٹمنٹس بڑے ڈیٹا بیسز سے مخصوص ڈیٹا کو مؤثر طریقے سے منظم کرنے اور بازیافت کرنے میں مدد کرتی ہیں۔

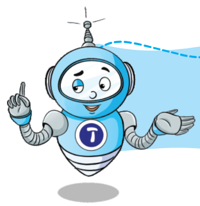
MEMORY BOX

یاد رکھیں:

1. آسان رسائی اور تجزیہ کے لیے ڈیٹا کو منظم طریقے سے محفوظ کیا جانا چاہیے۔
2. اسپریڈ شیٹس چھوٹے ڈیٹا سیٹس کے لیے مفید ہیں۔
3. ڈیٹا بیسز بڑے اور زیادہ پیچیدہ ڈیٹا کے لیے استعمال ہوتے ہیں۔
4. ایک ٹیبل ریکارڈز (قطاروں) اور فیلڈز (کالموں) کا استعمال کرتے ہوئے ڈیٹا کو ذخیرہ کرتا ہے۔
5. منظم ڈیٹا غلطیوں کو کم کرتا ہے اور تجزیہ کو آسان بناتا ہے۔

خلاصہ (Summary)

- اسپریڈ شیٹس (Spreadsheets): چھوٹے ڈیٹا سیٹس کے لیے قطاروں اور کالموں کا گرڈ (جیسے ایکسل، گوگل شیٹس)۔
- ڈیٹا بیس (Database): ٹیبلز میں ذخیرہ شدہ ڈیٹا کا ایک منظم مجموعہ۔
- ٹیبل (Table): متعلقہ ڈیٹا کا ایک گروپ (ایک چھوٹی اسپریڈ شیٹ کی طرح)۔
- ریکارڈ (Record): ایک ٹیبل میں ایک واحد اندراج (قطار)۔
- فیلڈ (Field): ایک ٹیبل میں معلومات کا ایک مخصوص حصہ (کالم)۔
- SQL (سٹرکچرڈ قیوری لینگوئج): وہ لینگوئج جو ڈیٹا بیسز میں ڈیٹا کے ساتھ کام کرنے کے لیے استعمال ہوتی ہے۔



مشق

کثیر انتخابی سوالات

i ڈیٹا سائنس لائف سائیکل (DSLC) میں پہلا مرحلہ کیا ہے؟

الف ڈیٹا اکٹھا کرنا
ب ڈیٹا کی بصری پیشکش کرنا
ج مسئلہ کو سمجھنا
د ڈیٹا کا تجزیہ کرنا

ii بہت سے طالب علموں کی آراء اکٹھا کرنے کا بہترین طریقہ کون سا ہے؟

الف IoT آلات
ب سروے
ج ویب سکریپنگ
د ڈیٹا بیس

iii ڈیٹا میں جانبداری (Bias) کیا ہے؟

الف ڈیٹا کا بہت بڑا سائز
ب غیر منصفانہ یا یک طرفہ ڈیٹا
ج اچھی طرح کلین شدہ ڈیٹا
د تیز رفتار ڈیٹا

iv چارٹ بنانے کے لیے ابتدائی افراد کے لیے مثالی ٹول کون سا ہے؟

(الف) Tableau
(ب) Python
(ج) Google Sheets
(د) Hadoop

v ایک ڈیٹابیس معلومات کو-------------------کی شکل میں محفوظ کرتا ہے۔

(الف) سلائیڈز
(ب) ٹیبلز
(ج) بلاکس
(د) لئیرز

vi درج ذیل چارٹس میں سے کون سا گروہوں کا موازنہ کرنے کے لیے بہترین ہے؟

(الف) پائی چارٹ
(ب) بار چارٹ
(ج) لائن چارٹ
(د) ایریا چارٹ

vii ڈیٹا کا تجزیہ کرنے کا بنیادی مقصد کیا ہے؟

(الف) پرانا ڈیٹا حذف کرنا
(ب) ڈیٹا کو سجانا
(ج) پیٹرنز تلاش کرنا اور فیصلے کرنا
(د) انٹرنیٹ پر اپ لوڈ کرنا

viii درج ذیل میں سے کون سا ٹول ایک جدید ڈیٹا ویژولائزیشن پلیٹ فارم ہے؟

(الف) MS Paint
(ب) Google data Studio
(ج) Note Pad
(د) WhatsApp

ix ڈیٹا کو ذخیرہ کرنے سے پہلے کیا کرنا چاہیے؟

(الف) اینیمیشنز شامل کرنا
(ب) اسے کلین اور تصدیق کرنا
(ج) اسے انکرپٹ کرنا
(د) اس کا ترجمہ کرنا

x درج ذیل میں سے کون سی ایک ڈیٹا سورس کی مثال ہے؟

(الف) یوٹیوب
(ب) سروے فارم
(ج) وائٹ بورڈ
(د) سلائیڈ شو

مختصر سوالات

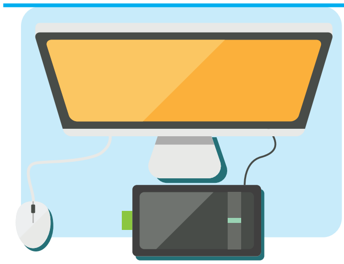
1. ڈیٹا اکٹھا کرنے سے پہلے مسئلہ کو سمجھنے کی کیا اہمیت ہے؟
2. ڈیٹا اکٹھا کرنے کے کوئی دو طریقوں کے نام بتائیں اور ہر ایک کی ایک مثال دیں۔
3. "ڈیٹا کلیننگ" سے ہمارا کیا مطلب ہے؟
4. ڈیٹا استعمال کرنے سے پہلے اسے تصدیق (validate) کرنا کیوں ضروری ہے؟
5. چارٹس کا استعمال کرتے ہوئے ڈیٹا کو بصری شکل دینے کا کیا مقصد ہے؟
6. ڈیٹا میں جانبداری نتائج کو کیسے متاثر کرتی ہے؟
7. ایک بے ترتیب سروے کو کلین کرنے کے لیے کوئی سے دو مراحل / اقدامات لکھیں۔
8. چارٹ بنانے کے لیے استعمال ہونے والے کوئی سے دو ٹولز کے نام بتائیں۔

تفصیلی سوالات

1. ڈیٹا سائنس لائف سائیکل (DSLC) کے چھ مراحل کو ہر ایک کے لیے ایک سطر میں بیان کریں۔
2. بار چارٹ، پائی چارٹ، اور لائن چارٹ کا مثالوں کے ساتھ تفصیلی موازنہ کریں۔
3. DBMS اور SQL کے درمیان فرق واضح کریں۔
4. SQL میں ایک کیوری (Query) لکھیں جس کے ذریعے ایک طالب علم کا تمام ڈیٹا حاصل کیا جائے جس کا رول نمبر 15 ہے۔

جوابات

1. ج – مسئلہ کو سمجھنا
2. ب – سروے
3. ب– غیر منصفانہ یا یک طرفہ ڈیٹا
4. ج – Google Sheets
5. ب– ٹیبلز
6. ب – بار چارٹ
7. ج – نمونے تلاش کرنا اور فیصلے کرنا
8. ب – Google data Studio
9. ب – اسے کلین اور تصدیق کرنا
10. ب – سروے فارم



یونٹ

6 مصنوعی ذہانت (AI) اور مشین لرننگ (ML) کا تعارف

حاصلاتِ تعلم: اس باب کے اختتام تک، آپ اس قابل ہو جائیں گے کہ:

- مصنوعی ذہانت (AI) اور مشین لرننگ (ML) کی تعریف کر سکیں۔
- AI اور ML میں فرق کو سمجھ سکیں، ان کی منفرد خصوصیات اور اطلاق کو جان سکیں۔
- سپروائزڈ لرننگ (Supervised Learning) کے تصور کو سمجھ سکیں اور پیش گوئیاں کرنے کے لیے سسٹمز کو سکھانے میں اس کے اطلاق کو جان سکیں۔
- ان سپروائزڈ لرننگ (Unsupervised Learning) کو سمجھ سکیں، جو ڈیٹا کو گروپ کرنے اور پیٹرن کی شناخت کے لیے استعمال ہوتی ہے۔
- مختلف سپروائزڈ اور ان سپروائزڈ لرننگ الگورتھمز سے واقفیت حاصل کر سکیں۔
- مختلف الگورتھمز کو سمجھیں اور مخصوص کاروباری مسائل کی بنیاد پر متعلقہ الگورتھمز کا انتخاب کر سکیں، جیسے کہ سادہ لینیئر ریگریشن (Simple Linear Regression) اور ڈیسیژن ٹری (Decision Trees)۔
- ماڈل کی کارکردگی کی اہم پیمائشوں (Metrics) جیسے کنفیوژن میٹرکس (Confusion Matrix) اور درستگی (Accuracy) سے واقفیت حاصل کر سکیں۔
- سمجھیں کہ AI اور ML روزمرہ کی زندگی، کاروبار اور تعلیم میں کہاں استعمال ہوتے ہیں۔
- AI کے استعمال میں تعصب یا طرفداری (Bias)، رازداری (Privacy)، اور ذمہ داری (Responsibility) پر بات کر سکیں۔

تعارف

موجودہ ڈیجیٹل دور میں، ٹیکنالوجی صرف مشینوں کے بارے میں نہیں ہے، یہ ان سمارٹ سسٹمز کے بارے میں ہے جو سوچ سکتے ہیں، سیکھ سکتے ہیں، اور بہتر فیصلے کرنے میں ہماری مدد کر سکتے ہیں۔ یہ باب مصنوعی ذہانت (AI) اور مشین لرننگ (ML) کے دلچسپ شعبوں کا تعارف کراتا ہے۔ یہ وہ طاقتور ٹیکنالوجیز ہیں جو وائس اسسٹنٹ (Voice Assistants) جیسے سری (Siri)، یوٹیوب پر ویڈیو کی تجاویز ChatGpt اور یہاں تک کہ موسم کی پیش گوئی کے پیچھے کارفرما ہیں۔ ان شعبوں کو سمجھنا آپ کو یہ دیکھنے میں مدد دیتا ہے کہ کمپیوٹر کس طرح ذہین بن رہے ہیں اور حقیقی زندگی کے مسائل کو حل کرنے میں معاون ثابت ہو رہے ہیں۔

اس باب میں، طلباء یہ جانیں گے کہ کس طرح ڈیٹا کو مشین پیٹرن پہچاننے، پیش گوئیاں کرنے، اور معلومات کو گروپ کرنے کی تربیت دینے کے لیے کا استعمال، بالکل ایک انسانی دماغ کی طرح، لیکن زیادہ تیزی اور درستی سے کرتی ہے۔

6.1 AI اور ML کا عمومی جائزہ

جدید دنیا میں، ہر سیکنڈ بہت بڑی مقدار میں ڈیٹا جمع کیا جا رہا ہے (جسے انسان اکیلے سنبھال نہیں سکتے)۔ یہیں پر AI اور ML کام آتے ہیں۔ یہ ٹیکنالوجیز ڈیٹا سے سیکھتی ہیں اور مشینوں کو سوچنے، فیصلہ کرنے، اور خود کو بہتر بنانے میں مدد کرتی ہیں (بالکل انسانوں کی طرح لیکن زیادہ تیزی اور مستقل مزاجی سے۔

مصنوعی ذہانت (AI) کیا ہے؟

- AI سے مراد ہے مشینوں کو وہ کام کرنا سکھانا جن کے لیے عام طور پر انسانی ذہانت کی ضرورت ہوتی ہے، جیسے تقریر کو سمجھنا، تصاویر کو پہچاننا، یا فیصلے کرنا۔
- تعریف: AI ایک مشین یا کمپیوٹر پروگرام کی وہ صلاحیت ہے جو ایسے کام سرانجام دے جو عام طور پر انسانی سوچ اور صلاحیت کا تقاضا کرتے ہیں۔

حقیقی زندگی کی مثالیں:

- تعلیمی اداروں بشمول سکولوں میں استعمال ہونے والی اردو سپیچ ٹو ٹیکسٹ ایپس (Urdu speech-to-text apps)۔
- بینکوں کی طرف سے کسٹمر سپورٹ کے لیے استعمال ہونے والے چیٹ بوٹس (Chatbots)۔
- بڑے شہروں میں ٹریفک کنٹرول کے نظام، جیسے لاہور میں سیف سٹی اتھارٹی کی طرف سے متعارف کرایا گیا نظام۔

مشین لرننگ (ML) کیا ہے؟

مشین لرننگ AI کا ایک ذیلی شعبہ ہے۔ اس سے مراد ہے ایک مشین کو ڈیٹا سے سیکھنے اور ہر بار پروگرام کیے بغیر خود سے فیصلے کرنے کی تربیت دینا۔

تعریف: ML ایک ایسا فیلڈ ہے جہاں کمپیوٹر کسی چیز کی پیش گوئی یا فیصلہ کرنے کے لیے ماضی کے ڈیٹا سے سیکھتا ہے۔

مثالیں:

- نیٹ فلکس (Netflix) یا یوٹیوب (YouTube) صارفین کی دیکھنے کی ہسٹری کی بنیاد پر ویڈیوز کی تجویز کرتے ہیں۔
- موسم کی ایک ایپ اگلے دن کے موسم کی پیش گوئی کرنے کے لیے ماضی کے درجہ حرارت کا استعمال کرتی ہے۔

AI اور ML کے درمیان فرق

جدول 6.1 : AI اور ML کے درمیان فرق کو ظاہر کرتا ہے۔

خصوصیت	AI	ML
تعریف	مشینوں کو انسانی رویے کی نقل کرنے کے لیے کافی ہوشیار بنانا۔	مشینوں کو ڈیٹا سے سیکھنے اور بہتر ہونے کی تربیت دینا۔
بنیادی مقصد	انسانوں جیسے سمارٹ کام انجام دینا۔	ڈیٹا سے پیش گوئیاں یا فیصلے کرنا۔

ڈیٹا کی ضرورت؟	کبھی کبھار (ذہانت سے کام کرنے کے لیے ڈیٹا استعمال کرتا ہے)۔	ہمیشہ (سیکھنے کے لیے ڈیٹا کی ضرورت ہوتی ہے)۔
انسانی رویہ؟	ہاں (جیسے بولنا، دیکھنا، یا سمجھنا)۔	نہیں (صرف پیٹرن سیکھتا ہے)۔
حقیقی زندگی کی مثال	جاز کیش پر چیٹ بوٹ جو گاہکوں کو جواب دیتا ہے۔	وہ ایپ جو اگلے ہفتے بجلی کے استعمال کی پیش گوئی کرتی ہے۔

کیا آپ جانتے ہیں؟

- AI کو دماغ سمجھیں، اور ML کو وہ تجربہ سمجھیں جو دماغ کو سکھاتا ہے کہ کس طرح بہتر ہونا ہے۔
- AI مشینوں کے ذہانت سے کام کرنے کا وسیع تصور ہے، جبکہ ML وہ ہے جب مشینیں پیش گوئیاں یا فیصلے کرنے کے لیے ڈیٹا سے سیکھتی ہیں۔

6.2 مشین لرننگ کی اقسام

پچھلے سیکشن میں، آپ نے سیکھا کہ ML مشینوں کو پیش گوئیاں یا فیصلے کرنے کے لیے ڈیٹا سے سیکھنے میں مدد کرتا ہے۔ جس طرح طلباء ایک استاد کی مدد سے یا خود سے تلاش کر کے سیکھ سکتے ہیں، اسی طرح مشینیں بھی دو اہم طریقوں سے سیکھ سکتی ہیں، یعنی سپروائزڈ لرننگ (Supervised Learning) اور اَن سپروائزڈ لرننگ (Unsupervised Learning)۔

سپروائزڈ لرننگ (Supervised Learning)

سپروائزڈ لرننگ میں، مشین کو درست جوابات (لیبلز) کے ساتھ ڈیٹا دیا جاتا ہے۔ یہ اپنے اندازوں کا درست جوابات سے موازنہ کر کے سیکھتی ہے اور وقت کے ساتھ اس کا سیکھنا بہتر ہوتا ہے۔

تعریف: سپروائزڈ لرننگ وہ ہے جب مشین کو لیبل والے ڈیٹا کا استعمال کرتے ہوئے سکھایا جاتا ہے، جہاں اِن پٹ اور درست آؤٹ پٹ دونوں فراہم کیے جاتے ہیں۔

مثال: فرض کریں کہ آپ ایک کمپیوٹر کو یہ پیش گوئی کرنے کی تربیت دے رہے ہیں کہ آیا کوئی پھل سیب ہے یا آم۔ آپ مشین کو پھلوں کی بہت سی تصاویر دکھاتے ہیں اور اسے بتاتے ہیں کہ

- "یہ ایک سیب ہے"
- "یہ ایک آم ہے"

کمپیوٹر ان مثالوں سے سیکھتا ہے۔ بعد میں، نئی تصاویر دکھانے پر یہ زیادہ درستی کے ساتھ پیش گوئی کر سکتا ہے۔

مثال: ایک سسٹم کو طلباء کے نمبروں اور حاضری کی بنیاد پر سکھایا جاتا ہے تاکہ یہ پیش گوئی کی جا سکے کہ کون فیل ہو سکتا ہے یا کسے اضافی مدد کی ضرورت ہو سکتی ہے۔

ان سپروائزڈ لرننگ (Unsupervised Learning)

ان سپروائزڈ لرننگ میں، مشین کو بغیر کسی لیبل کے صرف ڈیٹا دیا جاتا ہے۔ اسے خود ہی پیٹرن تلاش کرنے یا ملتی جلتی چیزوں کو گروپ کرنے کی ضرورت ہوتی ہے۔

تعریف: ان سپروائزڈ لرننگ میں مشین کو یہ بتائے بغیر کہ درست جواب کیا ہے، لیبل کے بغیر، ڈیٹا میں پیٹرن یا گروپس میں تلاش کرتی ہے۔

مثال: لاہور میں ایک کپڑوں کی دکان گاہکوں کا ڈیٹا (عمر، مقام، خریداری کی ہسٹری) جمع کرتی ہے۔ یہ گاہکوں کو لیبل نہیں کرتی، لیکن مشین پیٹرن تلاش کرتی ہے جیسے:

- ایک گروپ آرام دہ لباس خریدتا ہے۔
- دوسرا روایتی لباس کو ترجیح دیتا ہے۔

یہ دکان کو مارکیٹنگ میں مدد کرتا ہے، یہاں تک کہ گاہکوں کی صحیح اقسام جانے بغیر بھی۔

مثال: ایک سکول طلباء کو سٹڈی گروپس میں تقسیم کرنا چاہتا ہے۔ سسٹم ان کی دلچسپیوں اور مضامین کو چیک کرتا ہے اور صرف مشابہتوں کی بنیاد پر گروپس بناتا ہے، بغیر یہ جانے کہ کون سا گروپ "بہترین" ہے۔

سپروائزڈ بمقابلہ ان سپروائزڈ لرننگ

دونوں قسم کی لرننگ کا موازنہ جدول 6.2 میں فراہم کیا گیا ہے۔

جدول 6.2 سپروائزڈ بمقابلہ اَن سپروائزڈ لرننگ

خصوصیت	سپروائزڈ لرننگ	ان سپروائزڈ لرننگ
لیبلڈ ڈیٹا؟	ہاں (مشین کو درست جوابات بتائے جاتے ہیں)۔	نہیں (مشین خود سے پیٹرن تلاش کرتی ہے)۔
بنیادی کام	پیش گوئی کرنا یا درجہ بندی کرنا (Classify)۔	گروپ کرنا یا ملتی جلتی ڈیٹا کو اکٹھا کرنا (Cluster)۔
مثال	نمبروں اور مطالعاتی کے گھنٹوں کا استعمال کرتے ہوئے طالب علم کے نتائج کی پیش گوئی کرنا۔	خریداری کے رویے کی بنیاد پر گاہکوں کو گروپ کرنا۔
سکول میں استعمال	ان طلباء کی شناخت کرنا جنہیں مدد کی ضرورت ہے۔	پروجیکٹس کے لیے طلباء کے خودکار گروپس بنانا۔

سرگرمی:

بحث کریں اور میچ کریں (سپروائزڈ یا ان سپروائزڈ لرننگ)

مقصد: حقیقی زندگی کے حالات کے لیے صحیح قسم کی شناخت کر کے سپروائزڈ اور ان سپروائزڈ مشین لرننگ کے درمیان فرق کو سمجھنا۔

ہدایات:

درج ذیل ہر صورت حال کو غور سے پڑھیں (جدول 6.3 سے رجوع کریں)۔

جدول 6.3: سپروائزڈ (S) یا ان سپروائزڈ (U) لرننگ سے میچ کریں

سیریل نمبر	صورت حال کا بیان	S یا U
1	پیش گوئی کرنا کہ آیا کوئی طالب علم پاس ہو گا یا فیل۔	
2	مضمون کی ترجیحات کی بنیاد پر طلباء کو گروپ کرنا۔	
3	ان باکس سے سپیم ای میلز کی شناخت کرنا۔	
4	مصنوعات کو خصوصیات کی بنیاد پر اقسام میں تقسیم کرنا۔	

- اپنی نوٹ بک میں، ہر ایک کے لیے صحیح ML قسم لکھیں: آیا ہر صورت حال سپروائزڈ یا ان سپروائزڈ لرننگ کا اطلاق ہوتا ہے۔
- اپنے جوابات کو اپنی نوٹ بک میں ہر ایک کے لیے ایک مختصر وجہ کے ساتھ لکھیں۔
- ایک ہم جماعت اور گروپس کے ساتھ اپنے جوابات پر بات کریں۔
- کلاس میں بحث کے دوران اپنی وجوہات کی وضاحت کے لیے تیار رہیں۔

کیا آپ جانتے ہیں؟

DO YOU KNOW

سپروائزڈ لرننگ کو ایک استاد کے ساتھ سیکھیں، اور ان سپروائزڈ لرننگ کو خود سے تلاش کرنا سمجھیں۔

MEMORY BOX

یاد رکھیں:

- سپروائزڈ لرننگ پیش گوئی کرنے یا درجہ بندی کرنے کے لیے لیبل والے ڈیٹا کا استعمال کرتی ہے۔
- ان سپروائزڈ لرننگ چھپے ہوئے پیٹرن کو تلاش کرنے کے لیے لیبل کے بغیر ڈیٹا کا استعمال کرتی ہے۔
- دونوں پاکستان کے سکولوں، کاروباروں، ایپس، اور سرکاری نظاموں میں استعمال ہوتے ہیں۔
- اگر سسٹم درست جوابات والی مثالوں سے سیکھ رہا ہے، تو یہ سپروائزڈ ہے۔ اگر یہ دیئے گئے جوابات کے بغیر گروپ بنا رہا ہے، تو یہ ان سپروائزڈ ہے۔

6.3 AI/ML کے عملی اطلاقات

اب جب کہ آپ سمجھ چکے ہیں کہ AI اور ML کیا ہیں، اور مشینیں کیسے سیکھتی ہیں، اب یہ جاننے کا وقت ہے کہ یہ ٹیکنالوجیز ہمارے ارد گرد کہاں استعمال ہو رہی ہیں۔ ہم ان ٹیکنالوجیز کو ہر روز سکولوں، گھروں، کاروبار اور ایپس میں استعمال کرتے ہیں، حالانکہ ہم ہمیشہ انہیں نوٹس نہیں کرتے۔

سمارٹ اسسٹنٹس اور تجاویز

AI بہت سے سمارٹ ٹولز اور ایپس کو سپورٹ کرتا ہے جو ہماری زندگیوں کو آسان بنانے میں مدد کرتے ہیں۔

سمارٹ اسسٹنٹس (Smart Assistants): گوگل اسسٹنٹ، سری، اور یہاں تک کہ اردو وائس ٹائپنگ ایپس جیسے سمارٹ اسسٹنٹ، آواز کی کمانڈز کو سمجھنے، ترجمہ کرنے، یاد دہانیاں (reminders) سیٹ کرنے، اور سوالات کے جواب دینے کے لیے AI کا استعمال کرتے ہیں۔

مثال: گوگل اسسٹنٹ کو اردو میں "کل 8 بجے مجھے جگانا" کہیں، یہ اسپیچ ریکگنیشن اور نیچرل لینگویج پروسیسنگ تکنیکوں کا استعمال کرتے ہوئے ایک الارم سیٹ کرے گا، جو دونوں AI کو استعمال کرتے ہیں۔

ریکمنڈیشن سسٹمز (Recommendation systems): یہ سسٹم آپ کے ماضی کے رویے کی بنیاد پر تجویز کرتے ہیں کہ آپ کیا پسند کرسکتے ہیں۔

مثالیں:

- یوٹیوب (Youtube) آپ کے چند کلپس دیکھنے کے بعد ویڈیوز کی تجویز کرتا ہے۔
- دراز (Daraz) آپ کی براؤزنگ ہسٹری کی بنیاد پر کپڑوں کی تجویز کرتا ہے۔
- انسٹاگرام (Instagram) آپ کی پہلے دیکھی ہوئی ویڈیوز سے ملتی جلتی ویڈیوز دکھاتا ہے۔

یہ ایپس ML کا استعمال کرتے ہوئے تجزیہ کرتی ہیں کہ آپ کو کیا پسند ہے اور وقت کے ساتھ اپنی تجاویز کو بہتر بناتی ہیں۔

SMART TIPS

سمارٹ ٹپس

اگر کوئی ایپ تجاویز دیتی ہے، آپ کی آواز کو سمجھتی ہے، آپ کو جواب دیتی ہے، یا آپ سے سیکھتی ہے، تو یہ غالباً AI یا ML کا استعمال کر رہی ہے۔

سرگرمی:

ہماری روزمرہ کی زندگی میں AI

مقصد: روزمرہ کے ٹولز اور ایپلی کیشنز میں AI اور ML کے استعمال کو پہچاننا۔

ہدایات:

- 3 ایپس یا ٹولز کی فہرست بنائیں جو آپ نے پچھلے ہفتے استعمال کیے ہیں اور جو AI یا ML کا استعمال کرسکتے ہیں۔ مثال کے طور پر، یوٹیوب، گوگل میپس، چیٹ جی پی ٹی (ChatGPT)، سنیپ چیٹ (Snapchat)۔
- ایک کلاس میٹ کے ساتھ، ہر ایپ میں AI کیا کرتا ہے اس پر بات کریں۔ کیا یہ ویڈیوز کی تجویز کرتا ہے؟ کیا یہ آواز کے کمانڈز پر ردعمل دیتا ہے؟ کیا یہ آپ کو تیزی سے تلاش کرنے میں مدد کرتا ہے؟

اپنی مثالیں اور AI کا فنکشن اپنی نوٹ بک میں لکھیں۔

کسٹمر کا تجزیہ اور چیٹ بوٹس کا استعمال کرتے ہوئے کاروباری استعمال کے معاملات

AI اور ML کاروبار کے گاہکوں کو سمجھنے، سوالات کے جواب دینے، اور ان کی خدمات کو بہتر بنانے میں مدد کرتے ہیں۔

کسٹمر کا تجزیہ (Customer Analysis):

کمپنیاں صارفین کے بارے میں ڈیٹا اکٹھا کرتی ہیں تاکہ یہ معلوم کریں:
کون سی پروڈکٹ بیچنے کے لیے بہترین ہے۔
مثال: ایک گروسری ڈیلیوری ایپ ML کا استعمال ان اشیاء کی تجویز کے لیے کرتی ہے جو آپ اکثر خریدتے ہیں، اور پیش گوئی کرتی ہے کہ کس شہر میں کن اشیاء کی ضرورت ہے۔

چیٹ بوٹس اور خودکار مدد (Chatbots & Automated Help)

بہت سے پاکستانی بینک، ٹیلی کام، اور ای کامرس سائٹس اب چیٹ بوٹس کا استعمال کرتی ہیں جو سمارٹ سسٹمز ہیں جو انسانی ایجنٹ کی ضرورت کے بغیر سوالات کا جواب دے سکتے ہیں۔

DO YOU KNOW کیا آپ جانتے ہیں؟

AI ڈیٹا کی طاقت کا استعمال کرتے ہوئے کاروبار کو تیزی سے کام کرنے، پیسے بچانے، اور گاہکوں کو خوش رکھنے میں مدد کرتا ہے۔

بائس (Bias)، رازداری (Privacy)، اور ذمہ داری (Responsibility)

اگرچہ AI اور ML طاقتور ہیں، انہیں ذمہ داری کے ساتھ استعمال کیا جانا چاہیے۔
AI میں طرفداری (Bias in AI): بعض اوقات، مشینیں اس ڈیٹا سے غیر منصفانہ پیٹرن سیکھتی ہیں جس پر انہیں تربیت دی جاتی ہے۔ اسے بائس (Bias) یا طرفداری کہتے ہیں۔ مثال کے طور پر، اگر کسی AI سسٹم کو صرف انگریزی یا مردانہ آوازوں پر تربیت دی جاتی ہے، تو یہ اردو بولنے والوں یا خواتین کی آوازوں کے ساتھ اچھی طرح کام نہیں کر سکتا۔

MEMORY BOX **یاد رکھیں:** بائس (bias) یا طرفداری کو کم کرنے کے لیے ہمیشہ متنوع اور بیلنسڈ ڈیٹا استعمال کریں۔

رازداری اور ڈیٹا کا استعمال (Privacy and Data Use):

AI سسٹمز اکثر ذاتی ڈیٹا (جیسے نام، مقام، یا انتخاب) استعمال کرتے ہیں، جو رازداری کے خدشات کو جنم دیتا ہے۔ مثال کے طور پر، اگر کوئی ایپ آپ کی نقل و حرکت کو ٹریک کرتی ہے یا اجازت کے بغیر آپ کے مائیکروفون کو سنتی ہے، تو یہ خطرناک ہو سکتا ہے۔

بہترین عمل:

- صارف سے اجازت طلب کریں۔
- انکرپشن(encryption)کا استعمال کرتے ہوئے ڈیٹا کی حفاظت کریں۔
- ضرورت سے زیادہ ڈیٹا جمع نہ کریں۔

انسانوں کی ذمہ داری (Responsibility of Humans):

انسانوں کو AI سمجھداری سے استعمال کرنا چاہیے کیونکہ یہ صرف ایک ٹول ہے۔ اگر AI سے چلنے والا کوئی سسٹم غلط فیصلہ کرتا ہے، تو ذمہ داری ان انسانوں پر عائد ہوتی ہے جنہوں نے اسے بنایا یا استعمال کیا۔ مثال کے طور پر، اگر چہرے کی شناخت کا نظام غلطی سے کسی پر الزام لگاتا ہے، تو ڈویلپر کو مسئلہ ٹھیک کرنا چاہیے اور ذمہ داری لینی چاہیے۔

MEMORY BOX
یاد رکھیں:

- ML کا استعمال سمارٹ معاونین، شاپنگ ایپس، اور چیٹ بوٹس میں ہوتا ہے۔
- کاروبار انہیں کسٹمر سروس اور سیلز کے تجزیہ کے لیے استعمال کرتے ہیں۔
- ہمیں ہمیشہ اخلاقیات، انصاف، رازداری، اور ذمہ داری کو مد نظر رکھنا چاہیے۔

6.4 سپروائزڈ اور ان سپروائزڈ الگورتھمز

جیسا کہ پچھلے باب میں بیان کیا گیا ہے، ڈیٹا سائنس پیٹرن تلاش کرنے، پیش گوئیاں کرنے یا بڑے ڈیٹا کا تجزیہ کرنے کے لیے مختلف الگورتھمز کا استعمال کرتی ہے۔ صحیح الگورتھم کا انتخاب ڈیٹا کی قسم اور جس مسئلہ کو آپ حل کرنے کی کوشش کر رہے ہیں اس پر منحصر ہوتا ہے۔ کچھ عام ڈیٹا سائنس الگورتھمز یہ ہیں:

لینئر ریگریشن (Linear Regression): ایک سیدھے لائن کے تعلق کی بنیاد پر مستقبل کی ویلیوز کی پیش گوئی کرتا ہے (مثلاً، مطالعہ کے گھنٹوں سے طالب علم کے نمبروں کی پیش گوئی کرنا)۔

ڈیسیژن ٹری (Decision Trees): ہاں / نہیں سوالات کی بنیاد پر فیصلے کرنے کے لیے ایک ٹری (Tree-Like) جیسے ماڈل کا استعمال کرتا ہے (مثلاً، پیش گوئی کرنا کہ آیا کوئی گاہک خریدے گا یا نہیں)۔

کے-مینز (K-Means) (کلسٹرنگ): ملتی جلتی اشیاء کو ایک ساتھ گروپ کرتا ہے (مثلاً، ملتی جلتی دلچسپیوں والے طلباء کو گروپ کرنا)۔

یہ الگورتھمز تنظیموں کو حقیقی دنیا کے مسائل کو حل کرنے میں مدد کرتے ہیں، جیسے امتحانی نتائج کی پیش گوئی کرنا، بہتر مارکیٹنگ کے لیے گاہکوں کو گروپ کرنا، اور دھوکہ دہی یا صحت کے ڈیٹا میں پیٹرن کی شناخت کرنا۔

DO YOU KNOW
کیا آپ جانتے ہیں؟

سمجھیں۔ یہ کمپیوٹر کو بتاتا ہے کہ کون سے اجزاء (ڈیٹا) استعمال کرنے ہیں اور ایک الگورتھم کو باورچی خانے میں ایک ترکیب (Recipe) حتمی ڈش (پیش گوئی یا گروپ) حاصل کرنے کے لیے کن اقدامات پر عمل کرنا ہے۔

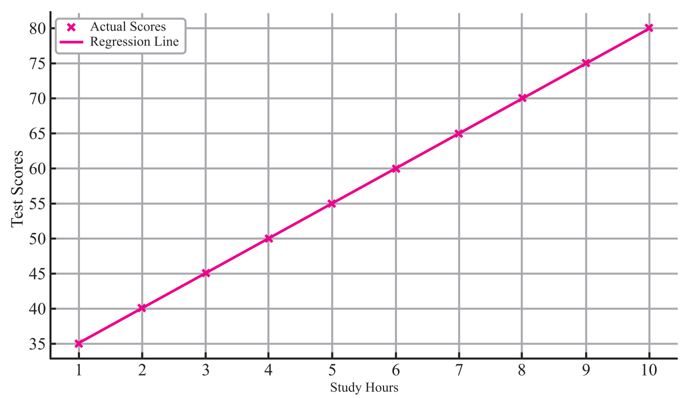
لینئر ریگریشن (Linear Regression)

سادہ لینئر ریگریشن الگورتھم ہمیں دو چیزوں کے درمیان تعلق کی بنیاد پر ایک ویلیو کی پیش گوئی کرنے میں مدد کرتا ہے۔ مثال کے طور پر، ایک استاد یہ پیش گوئی کرنا چاہتا ہے کہ طلباء ایک ٹیسٹ میں کتنی اچھی کارکردگی کا مظاہرہ کریں گے اس کی بنیاد پر کہ انہوں نے کتنے گھنٹے مطالعہ کیا۔ یہاں، مطالعہ کے گھنٹوں کو اِن پٹ (X) کے طور پر لیا جاتا ہے جبکہ ٹیسٹ سکور کو آؤٹ پٹ (Y) کے طور پر استعمال کیا جاتا ہے۔ لینئر ریگریشن ایک ایسی لائن تلاش کرتا ہے جو ڈیٹا پر بہترین طریقے سے فٹ ہو اور اگر طلباء نے، مثلاً، 2 گھنٹے مطالعہ کیا ہو تو نمبروں کی پیش گوئی کرنے میں مدد کرتا ہے۔ ماضی کے ڈیٹا کی بنیاد پر فروخت، درجہ حرارت، یا طالب علم کی کارکردگی کی پیش گوئی کرنا لینئر ریگریشن کے چند استعمال کے معاملات میں سے ہیں۔

منظر نامہ (Scenario): ایک استاد جدول 6.5 میں فراہم کردہ ڈیٹا کا استعمال کرتے ہوئے اپنے مطالعہ کے گھنٹوں کی بنیاد پر طالب علم کے ٹیسٹ سکور کی پیش گوئی کرنا چاہتا ہے۔ لینئر ریگریشن مستقبل کی ویلیوز کی پیش گوئی کرنے کے لیے ایک بہترین سیدھی لائن کھینچتا ہے جیسا کہ شکل 6.1 میں دکھایا گیا ہے۔ اس گراف میں، نیلے نقطے طالب علم کے اصل سکور ہیں جبکہ نیلی لائن پیش گوئی شدہ رجحان کو ظاہر کرتی ہے۔ اگر کوئی طالب علم 6.5 گھنٹے مطالعہ کرتا ہے، تو ماڈل لائن کا استعمال کرتے ہوئے اس کے سکور کی پیش گوئی کر سکتا ہے۔

جدول 6.5: طالب علم کا ڈیٹا

مطالعہ کے گھنٹے	ٹیسٹ سکور
01	35
02	40
03	45
04	50
05	55
06	60
07	65
08	70
09	75
10	80



شکل 6.1: لینئر ریگریشن سٹڈی کے گھنٹے اور ٹیسٹ کے سکور

کیا آپ جانتے ہیں؟

سادہ لینئر ریگریشن صرف ایک آزاد متغیر (اِن پٹ) کا استعمال ایک آوٹ کم کی پیش گوئی کرنے کے لیے کرتا ہے۔ تاہم، حقیقی زندگی میں، نتائج عام طور پر بہت سی چیزوں پر منحصر ہوتے ہیں۔
متعدد لینئر ریگریشن (Multiple linear regression) کمپیوٹر کو زیادہ پیچیدہ پیٹرن کو سمجھنے اور پیش گوئی کی درستگی کو بہتر بنانے میں مدد کرتا ہے۔

6.5 ماڈل کی کارکردگی کا جائزہ

ایک بار جب ML ماڈل کو تربیت دی جاتی ہے (جیسے پیش گوئی کرنا کہ کون ایک پروڈکٹ خریدے گا یا کوئی پیغام سپیم ہے یا نہیں)، تو ہمیں ماڈل کی کارکردگی کی پیمائش کرنے کی ضرورت ہوتی ہے۔ یہ فیصلہ کرنے میں مدد کرتا ہے کہ آیا ماڈل درست اور قابل اعتماد ہے۔ اس سیکشن میں، ہم سادہ تعریفوں اور مثالوں کے ساتھ چار اہم کارکردگی کی پیمائشوں کو دیکھیں گے۔

کنفیوژن میٹرکس (Confusion Matrix)

کنفیوژن میٹرکس ایک سادہ ٹیبل ہے جو ظاہر کرتا ہے کہ کتنی پیش گوئیاں / درجہ بندیاں درست تھیں یا غلط۔
جدول 6.6 ایک سادہ کنفیوژن میٹرکس دکھاتا ہے جس میں یہ غور کیا گیا ہے کہ آیا کوئی طالب علم پاس ہوگا یا فیل ہوگا۔ (درجہ بندی کی پرابلم)۔

جدول 6.6: کنفیوژن میٹرکس

	پیش گوئی شدہ: پاس	پیش گوئی شدہ: فیل
اصل: پاس	True Positive (TP) (درست مثبت)	False Negative (FN) (غلط منفی)
اصل: فیل	False Positive (FP) (غلط مثبت)	True Negative (TN) (درست منفی)

- TP = ماڈل نے پاس کی پیش گوئی کی اور طالب علم اصل میں بھی پاس تھا۔
- TN = ماڈل نے فیل کی پیش گوئی کی اور طالب علم اصل میں بھی فیل تھا۔
- FP = ماڈل نے پاس کی پیش گوئی کی لیکن طالب علم اصل میں فیل تھا۔
- FN = ماڈل نے فیل کی پیش گوئی کی اور طالب علم اصل میں پاس تھا۔

کنفیوژن میٹرکس یہ سمجھنے میں مدد کرتا ہے کہ آیا ماڈل درجہ بندی یا پیش گوئی میں غلطیاں کر رہا ہے اور ماڈل کی درستگی (Accuracy) (ایک اہم پیمائش) کا حساب لگانے میں مدد کرتا ہے۔

درستگی (Accuracy)

درستگی (A) ہمیں بتاتی ہے کہ کل کی گئی پیش گوئیوں میں سے کتنی پیش گوئیاں درست تھیں۔
اس کا حساب اس طرح لگایا جاتا ہے:

$$A = \frac{TP + TN}{TP + TN + FP + FN}$$

فرض کریں کہ ہمارے پاس 15 طلباء کا ڈیٹا ہے۔ اصل میں، 8 طلباء پاس تھے جبکہ 7 فیل تھے۔ ماڈل نے درج ذیل نتائج دیئے:

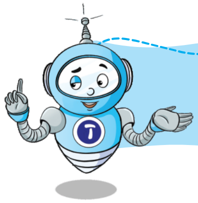
- TP = درست طور پر پیش گوئی شدہ پاس = 7
- TN = درست طور پر پیش گوئی شدہ فیل = 5
- FP = غلط طور پر پیش گوئی شدہ پاس = 2
- FN = غلط طور پر پیش گوئی شدہ فیل = 1

مندرجہ بالا فارمولے اور ماڈل کے نتائج کا استعمال کرتے ہوئے، ہم درستگی کا حساب اس طرح لگاتے ہیں:

$$A = \frac{7+5}{7+5+2+1} = \frac{12}{15} = 0.80 \text{ or } 80\%$$

خلاصہ (Summary)

- AI مشینوں کو سمارٹ کام کرنے کے قابل بناتا ہے۔ ML، جو AI کا ایک حصہ ہے، مشینوں کو ڈیٹا سے سیکھنے کی اجازت دیتا ہے تاکہ ہر بار خصوصی طور پر پروگرام کیے بغیر پیش گوئیاں کر سکیں یا پیٹرن کی شناخت کر سکیں۔
- سپروائزڈ لرننگ (مثلاً، طالب علم کے گریڈز یا کسٹمر کی خریداریوں کی پیش گوئی کرنا)۔
- ان سپروائزڈ لرننگ (مثلاً، لوگوں کو ترجیحات یا رویے کے لحاظ سے گروپ کرنا)۔
- ویلیوز کی پیش گوئی کے لیے لینئر ریگریشن۔
- فیصلے کرنے کے لیے ڈی سیژن ٹری (Decision Tree)۔
- ڈیٹا میں پیٹرن تلاش کرنے کے لیے کے مینز (عمومی جائزہ)۔



مشق

کثیر انتخابی سوالات

i مصنوعی ذہانت (AI) کیا ہے؟

الف ہاتھ سے ریاضی کے مسائل حل کرنا
ب مشینوں کا انسانوں جیسے کام کرنا
ج کمپیوٹر کا استعمال کرتے ہوئے ای میل بھیجنا
د آن لائن ویڈیوز دیکھنا

ii درج ذیل میں سے کون سی AI کی مثال ہے؟

الف آپ کے سوالات کا جواب دینے والا چیٹ بوٹ
ب وائٹ بورڈ کا استعمال کرنا
ج ایک TV ریموٹ
د کی بورڈ پر ٹائپ کرنا

iii مشین لرننگ (ML) کیا ہے؟

الف انسانوں کو مشینیں استعمال کرنا سکھانا
ب مشینوں کو ڈیٹا سے سیکھنا سکھانا
ج کمپیوٹر ہارڈ ویئر کی بنیادی باتیں سیکھنا
د انٹرنیٹ کے بغیر ایک مشین

iv کون سی ایپ ML کا استعمال کرتے ہوئے ویڈیوز کی تجویز کرتی ہے؟

الف گوگل میپس
ب مائیکروسافٹ پینٹ
ج یوٹیوب
د ونڈوز میڈیا پلیئر

v سپروائزڈ لرننگ میں، ڈیٹا ہوتا ہے:

الف غیر منظم
ب کسی جواب کے بغیر
ج درست جوابات کے ساتھ لیبل والا
د صرف ٹیکسٹ پر مبنی

vi درج ذیل میں سے کون سی ان سپروائزڈ لرننگ کی مثال ہے؟

(الف) طالب علم کے نمبروں کی پیش گوئی کرنا
(ب) سپیم ای میلز کی شناخت کرنا
(ج) کوئزز کو چیک کرنا
(د) بغیر لیبل کے دلچسپی کی بنیاد پر طلباء کو گروپ کرنا

vii گوگل اسسٹنٹ کی ایک مثال ہے:

(الف) ایک گیم ایپ
(ب) ایک ورڈ پروسیسر
(ج) AI کا استعمال کرتے ہوئے ایک سمارٹ معاون
(د) ایک کیلکولیٹر

viii ڈیٹا اینالسٹ کے لیے کون سی مہارت ضروری ہے؟

(الف) کھانا پکانا
(ب) بنیادی ڈرائنگ
(ج) ڈیٹا کی تشریح اور تصور
(د) ڈرونز اڑانا

ix AI زیادہ تر استعمال ہوتا ہے:

(الف) ہاتھ سے نوٹ لکھنے کے لیے
(ب) فون کال کرنے کے لیے
(ج) فون چارج کرنے کے لیے
(د) سمارٹ فیصلہ سازی اور آٹومیشن کے لیے

مختصر سوالات

1. AI اور ML کے درمیان کیا فرق ہے؟
2. پاکستان میں استعمال ہونے والی AI کی ایک حقیقی زندگی کی مثال دیں۔
3. سپروائزڈ لرننگ کیسے کام کرتی ہے؟
4. کنفیوژن میٹرکس کی تعریف کریں۔
5. کاروبار مشین لرننگ کا استعمال کیسے کر سکتے ہیں؟
6. ایک سمارٹ معاون اور ایک تجویز کے نظام (Recommendation system) کی ایک ایک مثال دیں۔
7. ان سپروائزڈ لرننگ میں کس قسم کا ڈیٹا استعمال ہوتا ہے؟

تفصیلی سوالات

1. مناسب مثالوں کے ساتھ سپروائزڈ اور ان سپروائزڈ لرننگ کی وضاحت کریں۔
2. مندرجہ ذیل پیمائشوں کا استعمال کرتے ہوئے مشین لرننگ میں ماڈل کی کارکردگی کا جائزہ کیسے لیا جاتا ہے وضاحت کریں؟
   a) کنفیوژن میٹرکس
   b) درستگی (Accuracy)
   c) پری سیژن (Precision)
3. ایک ماڈل پیش گوئی کرتا ہے کہ آیا طلباء پاس (1) ہوں گے یا فیل (0)۔ 20 طلباء میں سے:

True Positives (TP) = 8

True Negatives (TN) = 6

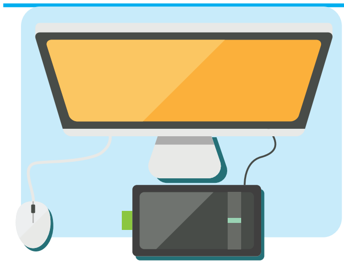
False Positives (FP) = 4

False Negatives (FN) = 2

آپ سے مطلوب ہے کہ (a) کنفیوژن میٹرکس پُر کریں، اور (b) درستی (Accuracy) کا حساب لگائیں۔

جوابات

1. ب — مشینوں کا انسانوں جیسے کام کرنا
2. الف — آپ کے سوالات کا جواب دینے والا چیٹ بوٹ
3. ب — مشینوں کو ڈیٹا سے سیکھنا سکھانا
4. ج — یوٹیوب
5. ج — درست جوابات کے ساتھ لیبل والا
6. د — بغیر لیبل کے دلچسپی کی بنیاد پر طلباء کو گروپ کرنا
7. ج — استعمال کرتے ہوئے ایک سمارٹ معاون
8. ج — ڈرونز اڑانا
9. د — فون چارج کرنے کے لیے



یونٹ 7

مصنوعی ذہانت (AI) کے اطلاقات

حاصلات تعلم: اس باب کے اختتام تک، آپ اس قابل ہو جائیں گے کہ:

- یہ وضاحت کر سکیں کہ (نیچرل لینگوئج پروسیسنگ) NLP، روبوٹکس، آواز کی شناخت (Voice Recognition)، اور دیگر شعبوں میں AI کو کیسے استعمال کیا جا سکتا ہے۔
- ایک ایسی ٹیکنالوجی کی شناخت کا تجربہ کر سکیں جن میں طلبہ NLP، روبوٹکس، آواز کی شناخت، اور دیگر پر مبنی مصنوعات کے ساتھ تعامل (interact) کریں۔
- NLP، روبوٹکس، آواز کی شناخت، اور دیگر فوائد اور حدود پر تبادلہ خیال کر سکیں۔
- یہ دکھا سکیں کہ ایسے مواقع موجود ہیں جہاں AI کا استعمال سماجی ناانصافیوں (social injustices) کا سبب بنتا ہے۔
- ان حالات کا جائزہ لینے کے لیے تنقیدی سوچ کی صلاحیتوں (critical thinking skills) کو فروغ دیں جہاں AI نے سماجی ناانصافیوں کو جنم دیا ہے اور بہتری کے ممکنہ شعبوں کی نشاندہی کریں۔
- AI کے سماجی اثرات (social implications)، خاص طور پر ناانصافیوں اور اخلاقی ذمہ داریوں کے حوالے سے، تحریری اور زبانی دونوں طرح سے مؤثر طریقے سے بات چیت (communicate) کر سکیں۔
- AI کی تیاری اور استعمال میں اخلاقی غور و فکر (ethical considerations) کے ساتھ ساتھ ان تحفظات کو نظر انداز کرنے کے ممکنہ نتائج سے آگاہی کا مظاہرہ کریں۔
- ایسے کیس اسٹڈیز یا منظر ناموں کا تجزیہ کر سکیں جہاں AI نے سماجی ناانصافیوں کو جنم دیا ہے، اور ان مسائل میں حصہ ڈالنے والے بنیادی عوامل پر تبادلہ خیال کر سکیں۔

تعارف

کمپیوٹر سائنس صرف کمپیوٹر استعمال کرنے کے بارے میں نہیں ہے؛ یہ مسائل حل کرنے، ذہین آلات بنانے، اور زندگی کو آسان بنانے کے لیے ہے۔ جدید کمپیوٹر سائنس کے سب سے اہم حصوں میں سے ایک مصنوعی ذہانت (Artificial Intelligence) ہے۔ نیچرل لینگوئج پروسیسنگ (NLP) کمپیوٹرز کو انسانی زبان سمجھنے میں مدد کرتی ہے۔ چیٹ بوٹس، ٹیکسٹ کے تجزیہ (analysis) کے آلات، اور خودکار ترجمہ (automatic translation) کی ایپس سب NLP کا حصہ ہیں۔ آواز کی ٹیکنالوجی وائس اسسٹنٹس جیسے الیکسا (Alexa) اور گوگل اسسٹنٹ کو چلاتی ہے۔ تجاویز کے نظام (Recommendation systems) صارف کی پسند کی بنیاد پر

آڈیوز، ویڈیوز، یا مصنوعات تجویز کرتے ہیں۔ خود مختار گاڑیاں (Autonomous vehicles)، جیسے کہ خود سے چلنے والی کاریں، ڈرائیور کے بغیر محفوظ طریقے سے حرکت کرنے کے لیے AI کا استعمال کرتی ہیں۔ روبوٹکس فیکٹریوں، ہسپتالوں، اور یہاں تک کہ خلائی منصوبوں میں بھی استعمال ہوتا ہے۔

AI کو منصفانہ اور محفوظ طریقے سے استعمال کیا جانا چاہیے۔ اخلاقیات (ethics) کا مطالعہ کرنا ضروری ہے، تاکہ AI لوگوں کو نقصان نہ پہنچائے یا سماجی ناانصافیاں پیدا نہ کرے۔ کچھ نظام، اگر انہیں احتیاط سے ڈیزائن نہ کیا جائے، تو تعصب / جانبداری (bias) دکھا سکتے ہیں یا غیر منصفانہ فیصلے کر سکتے ہیں۔ یہی وجہ ہے کہ طلبہ اور مستقبل کے ڈویلپرز کو ڈیٹا کی رازداری، احتساب (accountability)، اور AI آلات بنانے کے اخلاقی اصولوں کے بارے میں سیکھنا چاہیے۔ یہ باب یہ بھی سکھائے گا کہ AI کے بارے میں تنقیدی سوچ کیسے اپنائی جائے، اس کے فوائد اور خطرات دونوں کو کیسے پہچانا جائے، اور یہ سمجھا جائے کہ AI ملازمتوں، تعلیم، صحت، اور معاشرے کو کیسے متاثر کرتی ہے۔

7.1 مصنوعی ذہانت (AI) کا تعارف

مصنوعی ذہانت (Artificial Intelligence - AI) کمپیوٹر سائنس کا ایک جدید میدان ہے جو مشینوں کو یہ سکھاتا ہے کہ انسانوں کی طرح "سوچیں" اور "عمل" کریں۔ یہ کمپیوٹرز کو مسائل حل کرنے، ڈیٹا سے سیکھنے، اور مرحلہ وار بہترین فیصلے کرنے میں مدد کرتی ہے۔

AI بہت سے شعبوں میں استعمال ہوتا ہے جیسے کہ موبائل ایپس، آن لائن شاپنگ، صحت کی دیکھ بھال، اور نقل و حمل۔ مثال کے طور پر، جب آپ سری (Siri) جیسا کوئی وائس اسسٹنٹ یا کسی ویب سائٹ پر چیٹ بوٹ استعمال کرتے ہیں، تو آپ AI استعمال کر رہے ہوتے ہیں۔ یہ ڈاکٹرز کو بیماریوں کا پتہ لگانے، ڈرائیورز کو نقشے استعمال کرنے، اور طلبہ کو آن لائن جوابات تلاش کرنے میں مدد کرتا ہے۔ (جیسا کہ شکل 7.1 میں دکھایا گیا ہے)۔

شکل 7.1: (AI) لوگوں کی مدد کرتی ہوئی

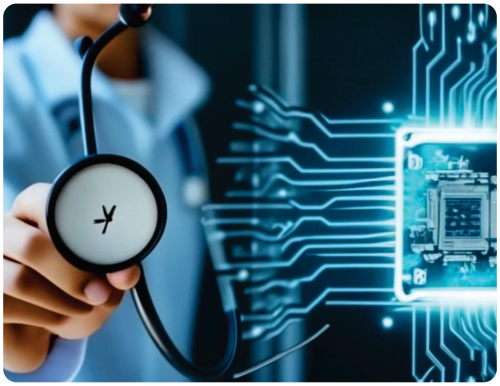
کیا آپ جانتے ہیں؟

AI اب بالکل انسان کی طرح کہانیاں لکھ سکتی ہے، ریاضی کے مسائل حل کر سکتی ہے، اور یہاں تک کہ آرٹ بھی تخلیق کر سکتی ہے!

سرگرمی:

"ہمارے ارد گرد کی AI" پوسٹر کا کام

طلبہ دو کے گروپ میں کام کریں گے اور ایک چھوٹا پوسٹر بنائیں گے جو یہ دکھائے کہ AI معاشرے کے مختلف حصوں جیسے ہسپتالوں، سکولوں، یا بینکوں میں کیسے استعمال ہوتی ہے۔ وہ تصاویر بنا سکتے ہیں، رسالوں سے کٹنگز استعمال کر سکتے ہیں، یا الفاظ سے وضاحت کر سکتے ہیں۔ AI کی مختلف شعبوں میں اہمیت کو سمجھنے کے لیے پوسٹرز جماعت میں دکھائے جائیں گے۔



شکل 7.2: AI بیماریوں کی تشخیص میں ڈاکٹروں کی مدد کرتی ہے۔

ملازمتوں اور کام پر AI کا اثر

AI کئی شعبوں میں لوگوں کے کام کرنے کے طریقے کو بدل رہی ہے۔ فیکٹریوں میں، روبوٹ اور مشینیں اب وہ کام کر رہے ہیں جو کبھی انسان کرتے تھے۔ یہ مشینیں زیادہ تیز، زیادہ دیر تک، اور کم غلطیوں کے ساتھ کام کر سکتی ہیں۔

اس کے ساتھ ہی، AI نئی قسم کی ملازمتیں بھی پیدا کر رہی ہے۔ AI سسٹمز کو ڈیزائن، ٹیسٹ، اور انتظام (manage) کرنے کے لیے لوگوں کی ضرورت ہے۔ ڈیٹا سائنٹسٹ، AI ڈویلپر، اور مشین لرننگ انجینئر جیسی ملازمتیں تیزی سے بڑھ رہی ہیں۔

سرگرمی:

"AI سے پہلے اور AI کے بعد" چارٹ

طلبہ کو دو کالموں والی ایک ورک شیٹ دی جائے گی: "AI سے پہلے" اور "AI کے بعد"۔ وہ ان ملازمتوں یا کاموں کی فہرست بنائیں گے جو انسان کیا کرتے تھے اور اب AI ان کاموں میں کس طرح مدد کرتا ہے یا انہیں تبدیل کر دیتا ہے۔

7.2 ڈیٹا اور AI

ڈیٹا AI کا سب سے اہم حصہ ہے۔ AI ڈیٹا کا مطالعہ کر کے سیکھتی ہے، بالکل اسی طرح جیسے کوئی طالب علم کتابیں پڑھ کر سیکھتا ہے۔ اگر ڈیٹا بڑا اور درست ہو، تو AI سسٹم زیادہ ذہین بن جاتا ہے اور بہتر نتائج دیتا ہے۔

AI سسٹم سیکھنے اور بہتر ہونے کے لیے ڈیٹا پر انحصار کرتے ہیں۔ جب وہ مختلف ذرائع سے بڑی مقدار میں معلومات حاصل کرتے ہیں، تو وہ پیٹرنز (patterns) تلاش کر سکتے ہیں اور بہتر فیصلے کرنا سیکھ سکتے ہیں۔

کیا آپ جانتے ہیں؟

ناقص معیار کے ڈیٹا پر تربیت یافتہ AI غلط جوابات دے سکتا ہے یا غیر منصفانہ فیصلے کر سکتا ہے۔

خود کار ڈیٹا اکٹھا کرنا

خود کار ڈیٹا اکٹھا کرنے کا مطلب ہے کسی شخص کی ہاتھ سے ڈیٹا داخل کرنے کی ضرورت کے بغیر معلومات جمع کرنا۔ بہت سے آلات اور سافٹ ویئر خود ہی ڈیٹا اکٹھا کر سکتے ہیں۔ مثال کے طور پر، موبائل فون خود بخود مقام (location)، استعمال، اور تلاش کی ہسٹری اکٹھا کرتے ہیں۔ آن لائن ویب سائٹس بھی صارف کے کلکس، ویوز، اور ترجیحات اکٹھا کرتی ہیں۔

کیا آپ جانتے ہیں؟

"ایپس کون سا ڈیٹا اکٹھا کرتی ہیں؟"

سروے: طلبہ اپنے فون پر 3-2 ایپس کی جانچ کریں گے اور نوٹ کریں گے کہ یہ ایپس کس قسم کا ڈیٹا اکٹھا کرتی ہے (مقام، آواز، کلکس، تلاش کی ہسٹری)۔

خود کار ڈیٹا کولیکشن کے ساتھ، AI تیزی سے ایڈجسٹ ہو سکتی ہے اور صارف کی ضروریات کا جواب دے سکتی ہے۔ یہ وقت بھی بچاتی ہے، غلطیوں کو کم کرتی ہے، اور سسٹمز کو ہر وقت انسانی مداخلت کی ضرورت کے بغیر کام کرنے کی اجازت دیتی ہے۔

ڈیٹا کی اقسام:

معیار (Quality) اور مقدار (Quantity)

مصنوعی ذہانت (AI) میں، استعمال ہونے والے ڈیٹا کی قسم بہت اہم ہے۔ "مقدار سے مراد یہ ہے کہ کتنا ڈیٹا دستیاب ہے"۔ جب ایک AI نظام کو بڑی تعداد میں مثالیں ملتی ہیں، تو وہ مختلف حالات سے زیادہ واقف ہو جاتا ہے۔ مثال کے طور پر، اگر کوئی مشین بہت سی مختلف تصاویر یا متن دیکھتی ہے، تو وہ نئے مواد کو سمجھنے میں بہتر ہو جاتی ہے۔

"معیار کا مطلب ہے کہ ڈیٹا کتنا مفید اور درست ہے"۔ اگر ڈیٹا میں غلطیاں، خالی / نامکمل ویلیوز، ابہام والے لیبل ہیں، تو AI غلط چیزیں سیکھ سکتی ہے۔ لہذا، ڈیٹا کی مقدار اور درستگی دونوں ہی مضبوط AI نظام بنانے میں بڑا کردار ادا کرتے ہیں۔

کیا آپ جانتے ہیں؟

AI دونوں قسم کے ڈیٹا کے ساتھ بہتر سیکھتی ہے: بڑا (مقدار) اور درست (معیار)۔

سرگرمی:

"کلین (Clean) بمقابلہ میسی (Messy) ڈیٹا" میچنگ ٹاسک

طلبہ کو دو چھوٹی ڈیٹا ٹیبلز دی جاتی ہیں: ایک واضح اور منظم معلومات کے ساتھ (اچھا معیار) اور ایک نامکمل یا غلط تفصیلات کے ساتھ (برا معیار)۔ وہ ہر ٹیبل سے سوالات کا موازنہ کرتے ہیں اور سمجھتے ہیں کہ ناقص ڈیٹا کے ساتھ AI کے لیے اچھے فیصلے کرنا کتنا مشکل ہے۔

7.3 AI کا حقیقی دنیا میں استعمال (Real-World Applications of AI)

AI اب روزمرہ زندگی کے بہت سے شعبوں میں استعمال ہو رہی ہے۔ یہ کاروباروں، سکولوں، ہسپتالوں، اور یہاں تک کہ گھروں کو بھی زیادہ ذہین اور زیادہ مؤثر بنانے میں مدد کرتی ہے۔ یہ AI کے حقیقی استعمال ہیں جو کام کو آسان اور تیز بناتے ہیں۔

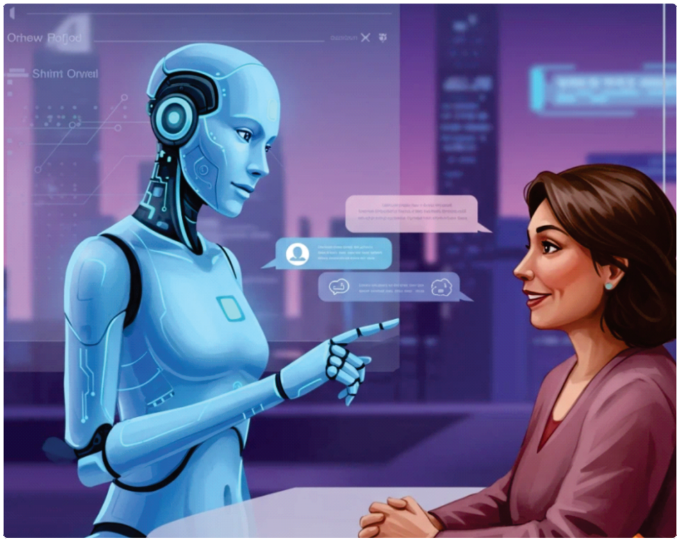
نیچرل لینگویج پروسیسنگ (NLP)

نیچرل لینگویج پروسیسنگ (NLP) AI کا ایک حصہ ہے جو کمپیوٹرز کو انسانی زبان سمجھنے میں مدد کرتا ہے۔ یہ مشینوں کو متن پڑھنے، آواز سننے، اور جوابات دینے کی اجازت دیتا ہے جس طرح لوگ سمجھ سکتے ہیں۔

- مثال کے طور پر، جب کوئی شخص سرچ بار میں کوئی سوال ٹائپ کرتا ہے یا وائس اسسٹنٹ سے بات کرتا ہے، تو NLP , AI سسٹم کو صحیح طریقے سے سمجھنے اور جواب دینے میں مدد کرتا ہے (جیسا کہ شکل 7.3 میں دکھایا گیا ہے)۔
- کسٹمر سروس میں، یہ انسانی مدد کے بغیر عام سوالات کا جواب دینے میں مدد کرتا ہے۔

کیا آپ جانتے ہیں؟ (DO YOU KNOW)

ویب سائٹس پر موجود چیٹ بوٹس آپ کے سوالات کا 24/7 جواب دے سکتے ہیں، یہ سب NLP کی بدولت ہے۔



شکل 7.3: بولنے والا چیٹ بوٹ لوگوں کی مدد کرتا ہے۔

روزمرہ کی زندگی میں روبوٹکس

روبوٹکس AI کا ایک اہم حصہ ہے جس میں انسانوں جیسے کام کرنے کے لیے ڈیزائن کی گئی مشینیں شامل ہیں۔ ان مشینوں کو روبوٹس کہا جاتا ہے۔ روبوٹس کئی جگہوں پر استعمال ہوتے ہیں جیسے فیکٹریاں، ہسپتال، گھر، اور یہاں تک کہ خلا میں بھی۔

- وہ بھاری سامان اٹھانے، سرجری میں مدد کرنے، یا فرش خود بخود صاف کرنے جیسے کام انجام دے سکتے ہیں۔
- کچھ روبوٹس بزرگوں یا معذور افراد کو روزمرہ کے سادہ کاموں میں مدد دے سکتے ہیں۔

چونکہ روبوٹ تھکتے نہیں ہیں اور عمدہ درستگی کے ساتھ کام کر سکتے ہیں، اس لیے وہ زندگی کے بہت سے شعبوں میں زیادہ عام ہوتے جا رہے ہیں۔

کیا آپ جانتے ہیں؟

DO YOU KNOW

روبوٹ اب کچھ ریستورانوں میں کھانا پیش کرنے اور میزیں صاف کرنے کے لیے استعمال ہوتے ہیں۔

آواز کی ٹیکنالوجی اور آواز سے کنٹرول

آواز کی ٹیکنالوجی A I کی ایک قسم ہے جو کمپیوٹرز کو بولی ہوئی باتوں کو سننے اور سمجھنے کی اجازت دیتی ہے۔ یہ مشینوں کو آواز کو متن میں بدلنے اور جوابات کے ساتھ ردِعمل ظاہر کرنے میں مدد کرتی ہے۔

- فون یا سمارٹ آلات میں استعمال ہونے والے وائس اسسٹنٹس اس ٹیکنالوجی کی مثالیں ہیں۔
- مثال کے طور پر، کسٹمر سروس میں، یہ کسی انسانی آپریٹر کے بغیر کالوں کا جواب دینے اور صارفین کی رہنمائی میں مدد کرتا ہے۔
- سمارٹ گھروں میں، لوگ صرف بول کر لائٹس یا آلات کو کنٹرول کر سکتے ہیں۔
- صحت کی دیکھ بھال میں، یہ ڈاکٹروں کو ٹائپ کیے بغیر نوٹس ریکارڈ کرنے میں مدد کر سکتا ہے۔

آواز کی ٹیکنالوجی لوگوں کے لیے صرف اپنی آواز سے مشینوں کو استعمال کرنا آسان اور تیز تر بناتی ہے۔

MEMORY BOX **یاد رکھیں:**

سمارٹ فون اب آپ کی آواز کا استعمال کرتے ہوئے کھل سکتے ہیں، کسی چیز کو چھونے کی ضرورت نہیں۔

کیا آپ جانتے ہیں؟

DO YOU KNOW

کچھ کاریں اب موسیقی چلانے، کال کرنے، یا سمتیں دینے کے لیے آواز کے احکامات (voice commands) کا جواب دیتی ہیں۔

سرگرمی:

مشق "بولو اور دیکھو"

طلبا فون یا کمپیوٹر کا استعمال کرتے ہوئے، طلبہ آواز کو متن میں بدلنے والے آلات (جیسے گوگل وائس ٹائپنگ) کو آزمائیں گے۔ وہ ایک جملہ بولیں گے اور دیکھیں گے کہ کیا وہ آلہ اسے صحیح طور پر ٹائپ کرتا ہے۔ پھر وہ مختلف لہجوں یا رفتار میں بولنے کی کوشش کریں گے اور نتائج کا موازنہ کریں گے۔

تجاویز کے نظام (Recommendation Systems)

تجاویز کے نظام AI کا ایک حصہ ہیں جو صارفین کو ان کی پسند اور پچھلے اقدامات کی بنیاد پر تجاویز دیتے ہیں۔ یہ نظام مطالعہ کرتے ہیں کہ کسی شخص نے پہلے کیا دیکھا، تلاش کیا، یا منتخب کیا ہے، اور پھر اسی طرح کے آپشنز (options) / تجاویز دکھاتے ہیں۔

- مثال کے طور پر، ایک آن لائن شاپنگ ویب سائٹ پر، اگر کوئی سکول کے بیگ دیکھتا ہے، تو AI دوسری قسم کے بیگ یا متعلقہ اشیاء تجویز کر سکتا ہے۔
- تعلیم میں، ایک ویب سائٹ طالب علم کی سیکھنے کی تاریخ کی بنیاد پر مددگار ویڈیوز یا مضامین تجویز کر سکتی ہے۔

تجاویز کے نظام مواد کو زیادہ ذاتی اور مفید بنا کر صارف کے تجربے کو بہتر بناتے ہیں۔ تاہم، ان نظاموں کو رازداری کی حفاظت اور بار بار ایک ہی طرح کا مواد دکھانے سے گریز کرنے کے لیے احتیاط سے استعمال کیا جانا چاہیے۔

SMART TIPS

سمارٹ ٹپس

جب یوٹیوب آپ کو "ویڈیوز جو آپ کو پسند آسکتی ہیں" دکھاتا ہے، تو یہ AI آپ کی دلچسپیوں کو سیکھ رہا ہے۔

سرگرمی:

سیمولیشن "میری پسند کا اندازہ لگاؤ"

ہر طالب علم خفیہ طور پر اپنی پسندیدہ قسم کی ویڈیو یا چیز کا انتخاب کرتا ہے۔ استاد مختلف انتخاب دکھاتا ہے (مثلاً، مزاح، کھیل، کھانا پکانا)۔ پچھلے انتخاب کی بنیاد پر، طلبہ ایک دوسرے کو اگلی چیز "تجویز" کرنے کی کوشش کرتے ہیں۔ یہ اس بات کی نقل کرتا ہے کہ AI پچھلے رویے کی بنیاد پر کیسے تجاویز دیتا ہے۔

خودمختار گاڑیاں (Self-driving Vehicles)

خودمختار گاڑیاں، جنہیں خود سے چلنے والی گاڑیاں بھی کہا جاتا ہے، وہ گاڑیاں ہیں جو انسانی ڈرائیور کے بغیر سڑک پر چل سکتی ہیں۔ یہ گاڑیاں اپنے ارد گرد کے ماحول کو سمجھنے اور خود سے فیصلے کرنے کے لیے AI کا استعمال کرتی ہیں۔

- وہ سمارٹ آلات جیسے سینسرز، کیمرے، گلوبل پوزیشننگ سسٹم (GPS)، اور ریڈارز سے لیس ہوتی ہیں تاکہ ٹریفک لائٹس، سڑک کے نشانات، دوسری کاروں، اور لوگوں کا پتہ لگا سکیں۔

- ان آلات اور AI کی مدد سے، کار ضرورت پڑنے پر محفوظ طریقے سے رفتار بڑھا سکتی ہے، آہستہ کر سکتی ہے، مڑ سکتی ہے، اور رک سکتی ہے (جیسا کہ شکل 7.4 میں دکھایا گیا ہے)۔
- یہ انسانی غلطیوں جیسے توجہ ہٹانے یا تھکاوٹ کی وجہ سے ہونے والے سڑک حادثات کی تعداد کو کم کر سکتی ہیں۔
- یہ ان لوگوں کی بھی مدد کر سکتی ہیں جو گاڑی چلانے کے قابل نہیں ہیں، جیسے بزرگ یا معذور افراد۔

تاہم، کچھ چیلنجز بھی ہیں۔ ان گاڑیوں کو کئی بار ٹیسٹ کرنا ضروری ہے تاکہ یہ یقینی بنایا جاسکے کہ وہ تمام حالات میں زیادہ محفوظ ہیں۔ ڈرائیوروں کے لیے ملازمت کے نقصان اور AI نظاموں کے لیے مضبوط ٹریفک قوانین کی ضرورت کے بارے میں بھی تشویش ہے۔ لوگ اس بارے میں بھی پریشان ہیں کہ اگر خود سے چلنے والی کار حادثے کا سبب بنتی ہے تو کون ذمہ دار ہوگا۔ یہی وجہ ہے کہ انجینئرز، حکومتیں، اور کمپنیاں مل کر حفاظت کو بہتر بنانے اور ان سمارٹ گاڑیوں کے لیے منصفانہ قوانین بنانے کے لیے کام کر رہے ہیں۔

سرگرمی:

منصوبہ بندی کا کام "سفر کی منصوبہ بندی کرو"

طلبہ اپنے سکول سے پارک یا بازار تک کا نقشہ کھینچتے ہیں۔ پھر وہ نشاندہی کرتے ہیں کہ ایک خود سے چلنے والی کار کو کن چیزوں کا پتہ لگانے کی ضرورت ہوگی (جیسے ٹریفک لائٹس، سڑک پار کرنے والے لوگ، یا سڑک کے نشانات)۔

شکل 7.4: خود سے چلنے والی کار

7.4: AI ایپس کے فوائد اور چیلنجز

AI کے اطلاقات لوگوں کو کاموں کو زیادہ تیزی اور زیادہ آسانی سے مکمل کرنے میں مدد کر رہے ہیں۔

- دفاتر میں، AI نظام فائلوں کا انتظام کرتے ہیں، سوالات کا جواب دیتے ہیں، اور بڑی مقدار میں ڈیٹا کو چھانٹتے ہیں۔
- تعلیم میں، AI سمارٹ لرننگ ایپس کے ذریعے طلبہ کی رہنمائی کر سکتا ہے جو ان کی ذہنی سطح کے مطابق ایڈجسٹ ہوتی ہیں۔
- یہ وقت بچاتا ہے اور لوگوں کو زیادہ اہم فیصلوں پر توجہ مرکوز کرنے میں مدد کرتا ہے۔

اس کے ساتھ ہی، کاروبار کو منصوبہ بندی، فروخت کی پیش گوئی، اور بہتر سروس دینے میں معاونت کرتا ہے۔ یہ سمارٹ آلات انسانی غلطیوں کو کم کر کے اور کارکردگی میں اضافہ کر کے کھیتی باڑی، بینکنگ اور نقل و حمل جیسے بہت سے شعبوں میں زندگیوں کو بہتر بنا رہے ہیں۔

AI ایپس کی حدود اور چیلنجز

AI سسٹم ہمیشہ درست نہیں ہوتے اور اگر ڈیٹا ناقص یا جانبدار (biased) ہو تو بعض اوقات غلط نتائج دیتے ہیں۔ مثال کے طور پر، اگر زبان واضح نہ ہو تو ایک چیٹ بوٹ سوال کو غلط سمجھ سکتا ہے۔

- قانون یا صحت جیسے سنجیدہ شعبوں میں، AI کی چھوٹی غلطیاں بھی بڑے مسائل پیدا کر سکتی ہیں۔ اسی لیے احتیاط سے جانچ کی ضرورت ہے۔
- ایک اور چیلنج یہ ہے کہ AI بہت سی ملازمتوں میں انسانی کارکنوں کی جگہ لے سکتی ہے، مثلاً کسٹمر سروسز، ڈیٹا انٹری اور دفتری کام، جس سے ملازمت کا نقصان ہو سکتا ہے۔
- اس کے علاوہ، بہت سے AI نظام منصفانہ فیصلے نہیں کر سکتے اگر انہیں احتیاط سے ڈیزائن نہ کیا جائے۔

کیا آپ جانتے ہیں؟

اگر AI سسٹمز کو نامکمل یا جانبدار ڈیٹا پر تربیت دی جائے تو وہ غلطیاں کر سکتے ہیں۔

7.5: AI کی اخلاقیات، انصاف، اور سماجی اثرات

AI میں اخلاقیات (Ethics in AI)

AI میں اخلاقیات کا مطلب ہے AI کو اس طرح استعمال کرنا جو سب کے لیے منصفانہ، ایماندار، اور محفوظ ہو۔ چونکہ AI فیصلے کر سکتا ہے، اس لیے یہ یقینی بنانا ضروری ہے کہ وہ فیصلے لوگوں کو نقصان نہ پہنچائیں یا کسی کے ساتھ ناانصافی نہ کریں۔ مثال کے طور پر، اگر کوئی AI سسٹم لوگوں کے ایک گروہ کو بہتر سروس دیتا ہے اور دوسروں کو نظر انداز کرتا ہے، تو یہ منصفانہ نہیں ہے۔ AI کو انصاف، حفاظت، اور انسانی حقوق کے احترام کو مدنظر رکھتے ہوئے احتیاط سے تیار کیا جانا چاہیے۔

SMART TIPS

سمارٹ ٹپس

اگر AI کو صحیح طریقے سے ڈیزائن نہ کیا جائے تو یہ غیر منصفانہ ہو سکتا ہے، اسی لیے ڈویلپرز کو صحیح اور غلط کے بارے میں سوچنا چاہیے۔

AI اور الگورتھم میں تعصب / جانبداری (Bias)

AI میں جانبداری (Bias) تب ہوتی ہے جب نظام اس ڈیٹا کی وجہ سے غیر منصفانہ نتائج دیتا ہے جس پر اسے تربیت دی گئی تھی۔ اگر ڈیٹا صرف لوگوں کے ایک گروہ کی نمائندگی کرتا ہے، تو AI دوسرے گروہوں کو سمجھ نہیں پائے گا یا ان کے ساتھ منصفانہ سلوک نہیں کرے گا۔

مثال کے طور پر، ایک چہرے کی شناخت کا آلہ تمام جلد کے رنگوں کے لیے اچھی طرح سے کام نہیں کر سکتا اگر اسے زیادہ تر صرف ایک قسم کی تصاویر پر تربیت دی گئی ہو۔ یہ ظاہر کرتا ہے کہ ڈیٹا میں مختلف لوگ، زبانیں، اور حالات شامل ہونے چاہیے۔ جانبداری (Bias) کو دور کرنا AI کو سب کے لیے زیادہ منصفانہ اور مددگار بنانے کی طرف ایک بڑا قدم ہے۔

سرگرمی:

کھیل "ایک ہی ان پٹ، مختلف آؤٹ پٹ؟"

طلبہ کو ایک فرضی AI سسٹم دیا جاتا ہے جو صارفین کے نام، رنگ، یا جنس کی بنیاد پر بے ترتیب یا غیر منصفانہ نتائج دیتا ہے۔ وہ اس بات کا تجزیہ کرتے ہیں کہ کیا یہ منصفانہ ہے اور حقیقی زندگی کے AI سسٹم احتیاط سے ڈیزائن نہ کیے جانے پر کیسے جانبدار (Biased) ہو سکتے ہیں۔

AI میں انصاف اور شفافیت

AI میں انصاف (Fairness) کا مطلب ہے کہ نظام کو تمام صارفین کے ساتھ برابری کا سلوک کرنا چاہیے، چاہے ان کی نسل، جنس، عمر، یا پس منظر کچھ بھی ہو۔ یہ ان نظاموں میں اہم ہے جو سکول میں داخلے، ملازمت کے انٹرویوز، یا قانونی معاملات کے بارے میں فیصلے کرتے ہیں۔ انصاف کو یقینی بنانے کے لیے، AI کو ایسے ڈیٹا پر تربیت دی جانی چاہیے جس میں ہر قسم کے لوگ اور حالات شامل ہوں۔

شفافیت (Transparency) کا مطلب ہے کہ لوگوں کو یہ سمجھنے کا حق ہونا چاہیے کہ ایک AI نظام کیسے کام کرتا ہے اور اس نے ایک خاص فیصلہ کیوں کیا۔

سرگرمی:

"اپنے انتخاب کی وضاحت کرو "AI" کا کردار

ایک طالب علم "AI" کا کردار ادا کرتا ہے، اور دوسرے اس سے کوئی انتخاب کرنے کے لیے کہتے ہیں (جیسے انعام کے لیے کسی طالب علم کا انتخاب)۔ AI کو وجہ بتانی ہو گی۔

اگر کوئی AI قرض کی درخواست کو مسترد کرتا ہے یا کسی طالب علم کو منتخب کرتا ہے، تو صارفین کو یہ جاننے کا حق ہے کہ نظام نے کن اصولوں یا اقدامات کی پیروی کی۔ شفافیت غلطیوں سے بچنے میں مدد کرتی ہے اور یہ اعتماد دیتی ہے کہ نظام ایمانداری سے کام کر رہا ہے۔

رازداری، ڈیٹا کی حفاظت، اور نگرانی (Surveillance)

رازداری (Privacy) کا مطلب ہے ذاتی معلومات کو محفوظ رکھنا اور اجازت کے بغیر اسے شیئر نہ کرنا۔ بہت سے AI نظام صارفین سے ڈیٹا اکٹھا کرتے ہیں، جیسے نام، پتے، مقام یا سرچ ہسٹری اگر اس ڈیٹا کو محفوظ نہیں رکھا جائے تو اسے چوری یا غلط استعمال کیا جا سکتا ہے۔

ڈیٹا کی حفاظت (Data security) کا مطلب ہے معلومات کو ہیکرز، وائرس، یا دیگر حملوں سے بچانا۔ AI سسٹمز کو صارف کے ڈیٹا کو محفوظ رکھنے کے لیے مضبوط حفاظتی طریقے استعمال کرنے چاہئیں۔

نگرانی (Surveillance) وہ ہے جب نظام یا کیمرے لوگوں کے عمل (Action) کی نگرانی کرتے ہیں۔ اگرچہ یہ حفاظت میں مدد کر سکتا ہے، لیکن اسے احتیاط سے استعمال کیا جانا چاہیے۔ لوگوں کو ہر وقت بغیر وجہ کے دیکھنا آزادی اور رازداری کو نقصان پہنچا سکتا ہے۔

AI اور سماجی مساوات (معاشرے میں انصاف)

AI کو اس طرح استعمال کیا جانا چاہیے کہ یہ تمام لوگوں کی یکساں مدد کرے، بشمول غریب، معذور، اور دور دراز علاقوں میں رہنے والے۔ بعض اوقات، AI جیسی نئی ٹیکنالوجیز صرف امیر یا تعلیم یافتہ گروہوں کے لیے دستیاب ہوتی ہیں، جبکہ دوسرے پیچھے رہ جاتے ہیں۔ یہ لوگوں کے درمیان ایک خلا (gap) پیدا کرتا ہے۔ معاشرے میں انصاف کی حمایت کے لیے، AI ٹولز کو اس طرح ڈیزائن کیا جانا چاہیے کہ ہر کوئی فائدہ اٹھا سکے، نہ کہ صرف چند لوگ۔

سرگرمی:

"سب کے لیے AI" پوسٹر ڈرائنگ

طلبہ ایک پوسٹر بناتے ہیں جو یہ دکھاتا ہے کہ AI کو سب کی مدد کیسے کرنی چاہیے؛ امیر ہو یا غریب، شہر ہو یا گاؤں۔ پوسٹرز کو انصاف اور مساوی رسائی دکھانی چاہیے۔

AI ڈیزائنرز کی ذمہ داری

AI خود سے اصول نہیں بناتے وہ انسانی ڈیزائنرز کی تخلیق کردہ ہدایات پر عمل کرتے ہیں۔ اس کا مطلب ہے کہ جو لوگ AI بناتے ہیں ان کی ایک بڑی ذمہ داری ہے کہ وہ یقینی بنائیں کہ ان کے نظام معاشرے کے لیے منصفانہ، محفوظ اور مددگار ہیں۔

اگر کوئی AI نظام غلط فیصلہ کرتا ہے یا کسی کے ساتھ غیر منصفانہ سلوک کرتا ہے، تو غلطی اکثر اس کے ڈیزائن کے طریقہ کار سے آتی ہے۔ اسی لیے AI ڈیزائنرز کو احتیاط سے سوچنا چاہیے کہ ان کے ٹولز حقیقی زندگی میں لوگوں کو کیسے متاثر کریں گے۔

ایسا AI بنانا جو مددگار اور قابل اعتماد ہو، یہ صرف ایک تکنیکی کام نہیں ہے بلکہ یہ ایک اخلاقی فرض ہے۔

کیا آپ جانتے ہیں؟

AI صرف اتنا ہی اچھا ہے جتنا کہ اسے بنانے والے لوگ، اسی لیے ڈیزائنرز کو ذمہ داری سے کام کرنا چاہیے۔

سرگرمی:

ذہنی مشق (Brainstorm) "ایک منصفانہ نظام بناؤ"

طلبہ چھوٹے گروپس میں ایک منصفانہ AI نظام ڈیزائن کریں۔ (مثلاً، سکول میں داخلے یا ملازمت کے انتخاب کے لیے)۔ وہ ان اصولوں کی فہرست بنائیں جن پر وہ جانبداری غلطیوں، یا نقصان سے بچنے کے لیے عمل کریں گے۔

AI میں اخلاقیات پر عمل کرنا

AI میں اخلاقیات کا مطلب ہے AI سسٹمز کو تخلیق کرتے یا استعمال کرتے وقت صحیح اور منصفانہ کام کرنا۔ AI ڈیزائنرز کو یقینی بنانا چاہیے کہ ان کے نظام لوگوں کو نقصان نہ پہنچائیں غلط معلومات نہ پھیلائیں، یا کسی کے ساتھ غیر منصفانہ سلوک نہ کریں۔ اخلاقیات ڈیزائنرز کو ایسے فیصلے کرنے میں مدد کرتی ہے جو صارفین کی حفاظت کرتے ہیں اور انصاف کی حمایت کرتے ہیں۔

اخلاقیات پر عمل کرنے میں یہ بھی شامل ہے کہ AI نظام کیسے کام کرتا ہے اس بارے میں ایماندار رہا جائے۔ صارفین کو معلوم ہونا چاہیے کہ کون سا ڈیٹا استعمال ہو رہا ہے اور فیصلے کیسے کیے جا رہے ہیں۔ اخلاقی ڈیزائن لوگوں اور ٹیکنالوجی کے درمیان اعتماد پیدا کرتا ہے۔

SMART TIPS

سمارٹ ٹپس

آیزک آسیموو (Isaac Asimov) (ایک سائنس فکشن مصنف) نے ایک بار کہا تھا۔ ایک روبوٹ کبھی بھی انسان کو نقصان نہیں پہنچاتا، یہ اصول آج بھی AI کی حفاظت سکھانے کے لیے استعمال ہوتا ہے!

AI بنانے کے اصول و ضوابط (Rules and Norms)

AI سسٹمز کو کچھ اصولوں اور ضوابط کے ساتھ بنایا جانا چاہیے تاکہ یہ یقینی بنایا جا سکے کہ وہ محفوظ، منصفانہ، اور مفید ہیں۔ یہ اصول ڈیزائنرز کو نظام بنانے میں رہنمائی کرتے ہیں۔ یہ اصول اس وقت اہم ہیں جب AI کا استعمال صحت، تعلیم، اور قانون جیسے شعبوں میں ہوتا ہے۔

حکومتیں، کمپنیاں، اور محققین اب AI کی تیاری کے لیے واضح رہنما اصول بنانے کے لیے مل کر کام کر رہے ہیں۔

ڈیزائنرز کو بھی یہ ترغیب دی جاتی ہے کہ وہ اپنے نظاموں کی مناسب طریقے سے جانچ کریں اور صارفین کو رائے دینے کی اجازت دیں۔

ان اصولوں پر عمل کرنے سے لوگوں اور ٹیکنالوجی کے درمیان اعتماد پیدا ہوتا ہے، اور یہ AI کو معاشرے میں ایک مثبت قوت بننے میں مدد دیتا ہے۔

سرگرمی:

ایک جملہ انگریزی میں گوگل ٹرانسلیٹ یا کسی ترجمہ کی ایپ میں ٹائپ کریں، اور اسے اردو میں تبدیل کریں۔ پھر چند الفاظ تبدیل کریں اور دوبارہ کوشش کریں۔

سرگرمی (روبوٹکس):

ایک روبوٹ کے کام کرتے ہوئے کی مختصر ویڈیو دیکھیں (جیسے پرزے جمع کرنا، کھانا پیش کرنا، یا صفائی کرنا)۔

دیکھنے کے بعد 3 سطریں لکھیں اس بارے میں کہ:

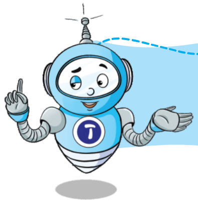
- o روبوٹ نے کیا کیا؟
- o AI نے اس کی کیسے مدد کی؟
- o اگر اس سے غلطی ہو جائے تو کیا ہوگا؟

· سرگرمی (آواز کی شناخت):

فون یا کمپیوٹر پر آواز سے ٹائپنگ کا استعمال کریں (مثلاً، گوگل ڈاکس وائس ٹائپنگ)۔ ایک جملہ بولیں اور متن کو ظاہر ہوتے ہوئے دیکھیں۔ پھر ایک مختلف جملہ بولیں اور دیکھیں کہ کیا متن درست ہے؟

خلاصہ (Summary)

- مصنوعی ذہانت (AI): کمپیوٹر سائنس کا وہ میدان ہے جو مشینوں کو انسانوں کی طرح "سوچنے" اور "عمل" کرنے کی تربیت دیتا ہے۔
- AI کی صلاحیتیں: مسائل حل کرنا، ڈیٹا سے سیکھنا، مرحلہ وار ہدایات کے بغیر سمارٹ فیصلے کرنا۔
- AI کے حقیقی دنیا کے اطلاقات: موبائل ایپس، آن لائن خریداری، صحت کی دیکھ بھال، نقل و حمل، آواز کے معاونین (voice assistants)۔
- ملازمتوں پر اثر: کچھ ملازمتوں (فیکٹری کے کام) کی جگہ لیتا ہے لیکن نئی ملازمتیں پیدا کرتا ہے (AI ڈویلپرز، ڈیٹا سائنٹسٹ)۔
- خودمختار گاڑیاں: خود سے چلنے والی کاریں جو سینسرز، کیمرے، گلوبل پوزیشننگ سسٹم (GPS) اور AI کا استعمال کرتی ہیں۔ یہ حادثات کو کم کرتی ہیں، اور معذور / بزرگ افراد کے لیے مدد فراہم کرتی ہیں۔
- انصاف (Fairness): نسل، جنس، عمر، پس منظر سے قطع نظر برابر کا سلوک۔
- شفافیت (Transparency): فیصلہ سازی کے عمل کا قابل فہم ہونا۔



مشق

کثیر انتخابی سوالات

i. AI کا کیا مطلب ہے؟

الف) خود کار ذہانت
ب) مصنوعی ذہانت
ج) فعال انٹر فیس
د) اطلاقی معلومات

ii. درج ذیل میں سے کون سا نیچرل لینگوئج کی پروسیسنگ (NLP) کی ایک مثال ہے؟

الف) آن لائن کیلکولیٹر
ب) آواز سے ٹائپنگ
ج) فائل کاپی کرنا
د) تصویر میں ترمیم

iii. کس قسم کا ڈیٹا AI کو بہتر فیصلے کرنے میں مدد کرتا ہے؟

الف) چھوٹا اور کمزور ڈیٹا
ب) صرف تصاویر
ج) بڑا اور اچھی کوالٹی کا ڈیٹا
د) لیبل کے بغیر ڈیٹا

iv. کون سا ٹول انسانی آواز کو سمجھنے کے لیے AI کا استعمال کرتا ہے؟

الف) ویڈیو پلیئر
ب) مترجم
ج) آواز کی پہچان
د) ماؤس

v. فیکٹریوں میں روبوٹس کا کردار کیا ہے؟

الف) ویڈیوز دیکھنا
ب) ویب سائٹس بنانا
ج) دہرائے جانے والے یا خطرناک کام کرنا
د) کہانیاں لکھنا

vi AI استعمال کرنے کا ایک چیلنج کیا ہے؟

الف یہ آہستہ کام کرتا ہے
ب اسے بند نہیں کیا جاسکتا
ج یہ انسانی نوکریوں کی جگہ لے سکتا ہے
د یہ ڈیٹا محفوظ نہیں کر سکتا

vii AI میں جانبداری (Bias) کیا ہے؟

الف کمپیوٹر کی خرابی
ب سب کے لیے مساوی نتائج
ج خراب ڈیٹا کی وجہ سے غیر منصفانہ نتائج
د تیز کارکردگی

viii AI میں انصاف (Fairness) کیوں اہم ہے؟

الف تاکہ سسٹم اچھا لگے
ب سست رفتار سے بچنے کے لیے
ج تمام صارفین کے ساتھ یکساں سلوک کرنے کے لیے
د زیادہ میموری استعمال کرنے کے لیے

ix درج ذیل میں سے کون سا AI ٹول نہیں ہے؟

الف چیٹ بوٹ
ب چہرے کی پہچان
ج ہجے کی جانچ پڑتال کرنے والا
د کاغذی نوٹ بک

مختصر سوالات

1. مصنوعی ذہانت (AI) کیا ہے؟
2. AI روزمرہ زندگی میں کیسے استعمال ہوتا ہے؟
3. AI کو بڑے اور اچھی کوالٹی کے ڈیٹا کی ضرورت کیوں ہوتی ہے؟
4. نیچرل لینگویج کی پروسیسنگ (NLP) کا کردار کیا ہے؟
5. روزمرہ زندگی میں روبوٹکس کی ایک مثال دیں۔
6. آواز کی پہچان (Speech recognition) ٹیکنالوجی کا مقصد کیا ہے؟
7. سفارشاتی نظام (Recommendation systems) کیسے کام کرتے ہیں؟

8. AI سسٹمز میں جانبداری (Bias) سے کیا مراد ہے؟

9. AI سسٹمز کو شفاف (transparent) اور منصفانہ (fair) کیوں ہونا چاہیے؟

تفصیلی سوالات

1. AI میں ڈیٹا کی اہمیت کیا ہے؟ وضاحت کریں کہ اچھے اور بڑے ڈیٹا کی ضرورت کیوں ہوتی ہے۔

2. AI کی کسی بھی تین حقیقی دنیا کی ایپلی کیشنز (جیسے NLP، روبو ٹکس، یا اسپیچ ٹیکنالوجی) کے بارے میں تفصیل سے لکھیں۔

3. AI میں اخلاقی مسائل (ethical issues) کی وضاحت کریں۔ انصاف (fairness)، پرائیویسی (privacy)، اور شفافیت (transparency) کیوں اہم ہیں؟

جوابات

1. ب — مصنوعی ذہانت

2. ب — آواز سے ٹائپنگ

3. ج — بڑا اور اچھی کوالٹی کا ڈیٹا

4. ج — آواز کی پہچان

5. ج — دہرائے جانے والے یا خطرناک کام کرنا

6. ج — یہ انسانی نوکریوں کی جگہ لے سکتا ہے

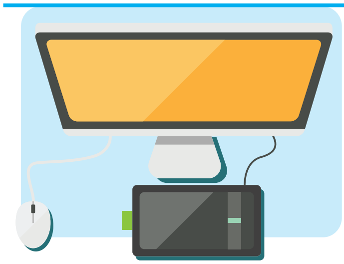
7. ب — سب کے لیے مساوی نتائج

8. ج — تمام صارفین کے ساتھ یکساں سلوک کرنے کے لیے

9. د — کاغذی نوٹ بک

یونٹ

8 ڈیجیٹل انٹرپرینیورشپ



حاصلاتِ تعلم: اس باب کے اختتام تک، آپ اس قابل ہو جائیں گے کہ:

- ڈیجیٹل انٹرپرینیورشپ میں قانونی تعمیل اور اخلاقی طریقوں کی اہمیت کی وضاحت کر سکیں۔
- ڈیٹا کی رازداری کے بنیادی قوانین کی شناخت اور وضاحت کر سکیں۔
- تضادِ مفادات (Conflicts of Interest) کو حل کرنے اور کاروباری سرگرمیوں میں انصاف کو یقینی بنانے کے لیے اخلاقی فیصلہ سازی کا اطلاق کر سکیں۔
- ذمہ دارانہ مارکیٹنگ کے طریقوں کو پہچانیں، ٹرو تھ فل اور ٹارگیٹڈ مارکیٹنگ کے لیے صارف کی رضامندی حاصل کرنے کی ضرورت کی وضاحت کر سکیں۔
- کاروباری چیلنجز سے نمٹنے کے لیے روٹ کاز اور SWOT تجزیہ جیسے مسئلہ حل کرنے کے منظم طریقوں کا استعمال کر سکیں۔
- ڈیٹا پر مبنی مارکیٹنگ کے تصور کی وضاحت کریں، بشمول قابل پیمائش اہداف کا تعین کرنا، کلیدی کارکردگی کے نقاط (KPIs) کی تعریف کرنا، تجزیاتی ٹولز کا استعمال اور سرمایہ کاری پر منافع (ROI) کی پیمائش کر سکیں۔
- کسٹمر ریلیشن شپ مینجمنٹ (CRM) سسٹمز کے کردار کی وضاحت کر سکیں، جو ذاتی مواصلات اور لائلٹی پروگرامز کے ذریعے طویل مدتی کسٹمر وفاداری پیدا کرنے میں مدد دیتے ہیں۔
- وضاحت کر سکیں کہ مارکیٹنگ آٹومیشن ٹولز کس طرح کارکردگی کو بہتر بنا سکتے ہیں، آپریشنل اخراجات کو کم کر سکتے ہیں، اور کسٹمر تک رسائی کو بڑھا سکتے ہیں۔

تعارف:

یہ باب قانونی، اخلاقی، اور حکمت عملی کی مہارتوں کی سمجھ فراہم کرتا ہے جو ایک کامیاب ڈیجیٹل کاروبار چلانے کے لیے اہم ہے۔ یہ وضاحت کرتا ہے کہ انٹرپرینیورز ڈیٹا پرائیویسی کے متعلقہ قوانین کی تعمیل کیسے کر سکتے ہیں؟ جیسے کہ پاکستان کا ڈیٹا پروٹیکشن بل، جنرل ڈیٹا پروٹیکشن ریگولیشن (GDPR)، اور بچوں کے آن لائن پرائیویسی پروٹیکشن ایکٹ (COPPA)۔ یہ مزید اخلاقی فیصلہ سازی کے اصولوں، تضادِ مفادات کی مینجمنٹ، اور منافع کو کسٹمر کی فلاح و بہبود کے ساتھ متوازن کرنے کے اصولوں کو بھی واضح کرتا ہے۔

یہ باب ڈیجیٹل ایتھکس کے کردار کا بھی جائزہ لیتا ہے جو کاروبار اور صارفین دونوں کے لیے محفوظ اور باوقار آن لائن ماحول کو فروغ دیتا ہے۔ حکمت عملی کے مارکیٹنگ تصورات، بشمول ڈیٹا کی بنیاد پر فیصلہ سازی، کسٹمر ریلیشن شپ مینجمنٹ (CRM)، مسابقتی تجزیہ (Competitive Analysis)، اور مارکیٹنگ آٹومیشن کو عملی مثالوں کے ساتھ زیر بحث لایا گیا ہے۔ اس باب کے اختتام تک، آپ ڈیجیٹل دور میں باخبر، ذمہ دار، اور حکمت عملی پر مبنی کاروباری فیصلے کرنے کا علم اور مہارت حاصل کر لیں گے۔

8.1 بنیادی سے اعلیٰ انٹر پرینیور شپ تک

نویں جماعت میں، آپ نے سیکھا تھا کہ انٹر پرینیور شپ اقدر پیدا کرنے کے لیے ایک کاروبار کو ڈیزائن کرنے، شروع کرنے اور منظم کرنے کا عمل ہے۔ ہم نے بحث کی تھی کہ یہ کس طرح اقتصادی ترقی کو فروغ دیتا ہے، روزگار کے مواقع پیدا کرتا ہے، اور ایسی نئی چیزیں متعارف کراتا ہے جو روزمرہ زندگی کو بہتر بناتی ہیں۔ آپ کو بنیادی ڈیجیٹل ٹولز، جیسے سوشل میڈیا پلیٹ فارمز، ای کامرس ویب سائٹس، اور مارکیٹ ریسرچ ایپلی کیشنز، سے بھی متعارف کرایا گیا تھا جو انٹر پرینیورز کو صارفین سے جڑنے اور اپنے آپریشنز کو مؤثر طریقے سے منظم کرنے کے قابل بناتے ہیں۔

دسویں جماعت میں، ہم اس بنیاد پر مزید مہارتیں حاصل کرتے ہیں تاکہ مسابقتی ڈیجیٹل کاروباری ماحول میں درکار اعلیٰ درجے کی مہارتوں کو تلاش کر سکیں۔ جیسا کہ قانونی تعمیل، تاکہ یہ یقینی بنایا جاسکے کہ کاروبار قانون کے دائرے میں کام کرتا ہے؛ ریگولیٹری آگاہی، تاکہ بدلتے ہوئے قواعد اور صنعت کی ضروریات کے ساتھ ہم آہنگ رہا جاسکے؛ اخلاقی فریم ورک، تاکہ منصفانہ اور ذمہ دارانہ فیصلہ سازی کی رہنمائی کی جا سکے؛ منظم مسئلہ حل کرنا، تاکہ چیلنجز سے مؤثر طریقے سے نمٹا جاسکے؛ اور حکمت عملی پر مبنی مارکیٹنگ، تاکہ ایک کاروبار کی موجودگی اور صارفین کے ساتھ اس کے تعلقات کو مضبوط بنایا جاسکے۔ یہ مہارتیں اس لیے اہمیت رکھتی ہیں کیونکہ آج کے انٹر پرینیورز سے توقع کی جاتی ہے کہ وہ اپنی دانشورانہ ملکیت کی حفاظت کریں، صارفین کے حقوق کا احترام کریں، ڈیٹا کی رازداری کو یقینی بنائیں، اور اپنی کمیونٹی کا اعتماد برقرار رکھیں۔

8.2 مسئلہ حل کرنا اور لچک پذیری:

کاروباری دنیا میں، چیلنجز کسی بھی وقت سامنے آسکتے ہیں۔ ان میں گاہک کا اعتماد کھونا، نئے حریفوں کا سامنا کرنا، یا ٹیکنالوجی میں تبدیلی شامل ہو سکتی ہے۔ کامیاب انٹر پرینیورز مشکلات کا سامنا کرنے پر ہار نہیں مانتے۔ اس کے بجائے، وہ ایک واضح طریقہ استعمال کرتے ہیں تاکہ مسئلہ کو سمجھیں، اس کی اصل وجہ تلاش کریں، اور اسے حل کرنے کے لیے مؤثر اقدامات کریں۔ وہ لچکدار (flexible) بھی رہتے ہیں اور حالات کا تقاضا ہونے پر تبدیلی کے لیے تیار رہتے ہیں۔ ان مہارتوں کو مسئلہ حل کرنا اور لچک پذیری (Adaptability) کہا جاتا ہے۔

مسئلہ کا منظم حل کرنا (Structured Problem Solving)

مسئلہ کا منظم حل کرنے کا مطلب ہے کسی مسئلے کی شناخت کرنا، یہ سمجھنا کہ وہ کیوں ہو رہا ہے، اور بہترین حل کا فیصلہ کرنے کے لیے مرحلہ وار طریقہ کار کا استعمال کرنا۔ اس میں اندازہ لگانے سے گریز کرنا شامل ہے۔ اس سے مسئلے کو دوبارہ ہونے سے روکنے میں مدد ملتی ہے۔ اس تصور کو نویں جماعت کی نصابی کتاب میں تفصیل سے بیان کیا جا چکا ہے۔

بنیادی وجہ کا تجزیہ (Root Cause Analysis)

ہر مسئلے کی ایک وجہ ہوتی ہے، جسے مسئلہ کی بنیادی وجہ (Root Cause) کہا جاتا ہے۔ اگر ہم صرف ظاہری مسئلے کو ٹھیک کریں گے تو وہ واپس آسکتا ہے۔ روٹ کاز کا تجزیہ اصل وجہ کو تلاش کرنے اور اس پر توجہ دینے میں مدد کرتا ہے، تاکہ حل دیرپا ہو۔
مثال: اگر آن لائن کاروبار میں مالک کو شکایات موصول ہو رہی ہیں تو مالک کو تحقیق کرنی چاہیے کہ آیا ڈیلیوری میں تاخیر، پروڈکٹ کی مبہم وضاحت، یا ناقص پیکیجنگ اصل وجہ ہے۔ ایک بار جب وجہ مل جاتی ہے، تو اسے درست کیا جا سکتا ہے۔

SWOT تجزیہ (SWOT Analysis)

SWOT ایک سادہ ٹول ہے جو چار اہم عوامل کو سمجھنے میں مدد دیتا ہے:

- S (Strengths) طاقت :- وہ چیزیں جو کاروبار اچھی طرح کرتا ہے (مثلاً، اعلیٰ معیار کی مصنوعات)۔
- W (Weaknesses) کمزوریاں :- وہ شعبے جن میں بہتری کی ضرورت ہے (مثلاً، سست کسٹمر سروس)۔
- O (Opportunities) مواقع :- ترقی یا بہتری کے امکانات (مثلاً، ایک نئی مصنوعات کی زیادہ مانگ)۔
- T (Threats) خطرات :- بیرونی عوامل جو کاروبار کو نقصان پہنچا سکتے ہیں (مثلاً، نئے حریف)۔

SWOT تجزیہ کاروبار کو اپنی موجودہ صورتحال کو سمجھنے اور مستقبل کے لیے دانشمندی سے منصوبہ بندی کرنے میں مدد کرتا ہے۔

SMART TIPS

سمارٹ ٹپس

ہر چیلنج ایک چھپا ہوا سبق ہوتا ہے۔ وہ کاروباری جو متجسس رہتے ہیں، اکثر "کیوں" پوچھتے ہیں، اور تیزی سے عمل کرتے ہیں، مسائل کو ترقی کے مواقع میں بدل دیتے ہیں۔

ایڈاپٹیبلیٹی اور رزیلیئنس (Adaptability and Resilience)

ایڈاپٹیبلیٹی کا مطلب ہے کہ ضرورت پڑنے پر تبدیل ہونے کے قابل ہونا اور تبدیلی کے لیے تیار رہنا ہے۔
رزیلیئنس (Resilience) کا مطلب ہے کہ ایک کاروبار کسی ناکامی یا مشکل کے بعد کتنی تیزی سے ری کور (Recover) ہو سکتا ہے۔ یہ دونوں مہارتیں کاروبار کو مشکل وقت میں بھی زندہ رہنے اور ترقی کرنے میں مدد دیتی ہیں۔

مارکیٹ یا ٹیکنالوجی میں تبدیلی کا ردِ عمل دینا:

کاروباری ماحول تیزی سے بدلتا ہے۔ صارفین کی نئی ضروریات، حریف، یا نئی ٹیکنالوجیز کسی بھی وقت ظاہر ہو سکتی ہیں۔ ایک کاروبار جو ان تبدیلیوں کے مطابق خود کو ڈھال لیتا ہے، کامیابی کا بہتر موقع رکھتا ہے۔

مثال: اگر صارفین آن لائن خریداری کو ترجیح دیتے ہیں۔ تو دوکاندار کو اشیاء کی گھروں میں ترسیل (Home Delivery) کی سہولت متعارف کروانی چاہیے۔

ناکامیوں سے نکلنا (Setbacks Recovery)

غلطیاں، ناکامیاں، یا نقصانات کسی بھی کاروبار میں ہو سکتے ہیں۔ ایک صلاحیت بحالی رکھنے والا کاروبار ناکامی کو قبول کرتا ہے، اس سے سیکھتا ہے اور بہتری کے لیے زیادہ محنت کرتا ہے۔

مثال: اگر کسی بیکری کو ذائقے کے بارے میں شکایات موصول ہوتی ہیں، تو وہ اپنی ترکیب (Reciep) کو بہتر بنا سکتی ہے، نئے ورژن کی جانچ کر سکتی ہے، اور صارفین کو اسے دوبارہ آزمانے کے لیے مدعو کر سکتی ہے۔

کیا آپ جانتے ہیں؟

دنیا کی کچھ بڑی کمپنیاں، جیسے نیٹ فلکس اور یوٹیوب، اس طرح شروع نہیں ہوئی تھیں جیسا کہ ہم انہیں آج جانتے ہیں۔ نیٹ فلکس نے پہلے ڈی وی ڈی (DVD) ڈاک کے ذریعے کرائے پر دی، اور یوٹیوب ویڈیو ڈیٹنگ کے لیے بنائی گئی تھی۔ انہوں نے اپنے خیالات کو بدلا اور پوری دنیا میں مشہور ہو گئے۔

8.3 اخلاقی کاروباری طریقے اور ڈیجیٹل ایتھکس

کاروبار میں ایتھکس کا مطلب ہے اخلاقی اصولوں پر عمل کرنا اور وہ کرنا جو درست، منصفانہ، اور ایماندار ہو۔ ڈیجیٹل دور میں، کاروبار کو یقینی بنانا چاہیے کہ ان کے فیصلے، مارکیٹنگ، اور آن لائن سرگرمیاں قانون اور صارفین کے حقوق دونوں کا احترام کریں۔ اخلاقی طریقے اعتماد پیدا کرتے ہیں اور ایک کاروبار کو ذمہ دارانہ طریقے سے بڑھنے میں مدد دیتے ہیں۔

اخلاقی فیصلہ سازی

اخلاقی فیصلہ سازی کا مطلب ہے کاروبار میں ایسے انتخاب کرنا جو منصفانہ، ایماندار، اور ذمہ دارانہ ہوں۔ یہ صحیح کام کرنے کے بارے میں ہے، یہاں تک کہ جب کوئی اور راستہ آسان یا زیادہ منافع بخش ہو۔ ایک کاروبار جو اخلاقی فیصلہ سازی پر عمل کرتا ہے، صارفین، ملازمین، اور برادری سے احترام اور اعتماد حاصل کرتا ہے۔ اخلاقی فیصلہ سازی ہمیشہ سب سے آسان راستہ نہیں ہوتا، لیکن یہ وہ راستہ ہے جو کاروبار کو سالمیت کے ساتھ ترقی کرنے، وفادار صارفین کو برقرار رکھنے، اور معاشرے میں ایک مثبت تاثر قائم رکھنے میں مدد دیتا ہے۔

مفادات کا ٹکراؤ (Conflict of interest)

مفادات کا ٹکراؤ تب ہوتا ہے جب کسی شخص کا ذاتی فائدہ یا تعلق ان کے کاروباری فیصلوں کو متاثر کر سکتا ہے۔ اس سے غیر منصفانہ سلوک یا تعصب پیدا ہو سکتا ہے۔

مثال: اگر ایک دکان کا مالک اپنے چچازاد بھائی کو نوکری پر رکھ لے اور کسی دوسرے ہنر مند شخص کو درخواست دینے کا موقع نہ دے، تو یہ غیر منصفانہ ہے۔ منصفانہ اور اخلاقی طریقہ یہ ہے کہ سب کو درخواست دینے کی اجازت دی جائے، ان کی مہارتوں کی جانچ کی جائے، اور پھر اس شخص کو بھرتی کیا جائے جو نوکری کے لیے بہترین ہو۔

منافع / کسٹمر کی فلاح و بہبود کے ساتھ متوازن کرنا:

ایک کاروبار کو چلانے کے لیے منافع کی ضرورت ہوتی ہے، لیکن منافع صارفین کو نقصان پہنچا کر یا ان کی حفاظت کو خطرے میں ڈال کر نہیں کمانا چاہیے۔

مثال: اگر کوئی کاروبار کم معیار کے اجزاء استعمال کرتے ہوئے کھانے کو زیادہ قیمت پر فروخت کرتا ہے، تو وہ ابتداء میں زیادہ پیسہ کما سکتا ہے۔ تاہم، صارفین کا اعتماد کھو جاتا ہے، اور کاروبار کو مستقبل میں نقصان ہوتا ہے۔ ایک اخلاقی کاروبار مناسب قیمت وصول کرتا ہے، محفوظ اور اچھی کوالٹی کی مصنوعات استعمال کرتا ہے، اور ایماندار معلومات فراہم کرتا ہے۔

ڈیجیٹل ایتھکس

ڈیجیٹل ایتھکس کا مطلب ہے انٹرنیٹ اور ڈیجیٹل آلات کو ذمہ داری اور دوسروں کی عزت کا خیال رکھتے ہوئے استعمال کرنا۔ کاروبار صارفین کی معلومات کو محفوظ رکھیں، درست تفصیلات فراہم کریں اور آن لائن شائستگی سے بات چیت کریں۔ صارفین بھی قواعد پر عمل کریں، دوسروں کے ساتھ احترام سے پیش آئیں اور نقصان دہ اقدامات سے گریز کریں۔ کاروبار واضح قواعد و ضوابط طے کر کے اور ہراساں کرنے یا دھمکانے کے خلاف فوری کارروائی کر کے محفوظ آن لائن خدمات مہیا کریں۔

ذمہ دارانہ مارکیٹنگ

ذمہ دارانہ مارکیٹنگ کا مطلب ہے مصنوعات اور خدمات کو ایمانداری اور انصاف کے ساتھ فروغ دینا، صارفین کو صحیح اور مکمل معلومات فراہم کرنا، اور جھوٹے وعدوں سے گریز کرنا۔ یہ طریقہ صارفین کو باخبر فیصلے کرنے میں مدد دیتا ہے اور کاروبار میں اعتماد پیدا کرتا ہے۔

- ہر کاروباری اشتہار میں سچی اور واضح معلومات ہونی چاہیے، جس میں بڑھا چڑھا کر بیان کی گئی یا چھپی ہوئی شرائط نہ ہوں جو صارفین کو گمراہ کر سکیں۔
- جب کوئی کاروبار اشتہارات دکھانے کے لیے صارفین کا ذاتی ڈیٹا جمع کرے، تو کاروبار کو صارفین سے واضح اجازت حاصل کرنی چاہیے اور وضاحت کرنی چاہیے کہ معلومات کا استعمال کیسے کیا جائے گا۔

سرگرمی:

مقصد:

طلباء منصفانہ کاروباری فیصلے کرنے، صارفین کو آن لائن محفوظ رکھنے، اور ایک ایماندار تشہیری آئیڈیا تیار کرنے میں اخلاقیات کا اطلاق کریں۔

طریقہ کار:

ا۔ نظر نامہ:

"آپ اور آپ کی ٹیم طلباء کے لیے ایک چھوٹی آن لائن دکان چلا رہے ہیں۔ آج، آپ کو تین فوری چیلنجز کا سامنا کرنا پڑے گا۔ کیا آپ کی دکان ایماندار، محفوظ، اور قابلِ اعتماد رہے گی؟"

۲۔ چیلنجز:

1. آپ کے پاس نوکری کے لیے دو امیدوار ہیں: ایک قریبی دوست اور ایک زیادہ ہنر مند اجنبی۔ فیصلہ کریں کہ کسے بھرتی کرنا ہے اور ایک جملے میں اس کی وجہ بتائیں۔
2. ایک صارف آپ کی دکان کے سوشل میڈیا پر غیر اخلاقی تبصرے پوسٹ کرتا ہے۔ فیصلہ کریں کہ شائستگی سے کیسے جواب دینا ہے اور اپنے پلیٹ فارم کو محفوظ رکھنا ہے۔
3. اپنے کاروبار کے لیے ایک مختصر تشہیری جملہ بنائیں جو سچا، واضح، اور گمراہ کن نہ ہو۔
4. ہر گروپ اپنے جوابات پیش کرے گا۔ کلاس سب سے زیادہ اخلاقی اور اعتماد سازی کے حل پر ووٹ دے گی۔

اعداد و شمار کی رازداری سے متعلق قوانین

اعداد و شمار کی رازداری کے قوانین وہ قواعد ہیں جو کاروباروں کے ذریعے جمع کی جانے والی ذاتی معلومات، جیسے نام، پتے، فون نمبر، یا ادائیگی کی تفصیلات کو محفوظ رکھتے ہیں۔ پاکستان میں، ذاتی اعداد و شمار کے تحفظ کے قانون کے مطابق اداروں کو چاہیے کہ:

- صرف وہی معلومات اکٹھی کریں جن کی انہیں ضرورت ہے۔
- معلومات کو صرف اس مقصد کے لیے استعمال کریں جو انہوں نے صارف کو بتایا تھا۔
- معلومات کو محفوظ رکھیں اور اجازت کے بغیر اسے کسی کے ساتھ شیئر نہ کریں۔
- کچھ دوسرے ممالک میں بھی مضبوط رازداری کے قوانین ہیں، جیسے: جنرل ڈیٹا پروٹیکشن ریگولیشن (GDPR): یہ یورپی ممالک کے لوگوں کے ذاتی اعداد و شمار کی حفاظت کے لیے ہے۔
- بچوں کے آن لائن رازداری کے تحفظ کا قانون (COPPA): یہ 13 سال سے کم عمر کے بچوں کے ذاتی اعداد و شمار کی حفاظت کے لیے ہے۔

یہ قوانین پاکستانی کاروباری افراد کے لیے اہم ہیں اگر ان کے صارف دوسرے ممالک سے تعلق رکھتے ہوں، کیونکہ انہیں ان ممالک کے قواعد پر بھی عمل کرنا پڑتا ہے۔

ضابطوں سے آگاہی (Regulatory Awareness)

ضابطوں سے آگاہی کا مطلب ہے یہ جاننا کہ آپ کے کاروبار پر کون سے قوانین اور قواعد لاگو ہوتے ہیں اور یہ یقینی بنانا کہ آپ ان کی پیروی کر رہے ہیں۔

اس میں شامل ہے:

- باخبر رہنا: باقاعدگی سے یہ جانچنا کہ آیا کوئی قانون تبدیل ہوا ہے جو آپ کے کاروبار کو متاثر کر سکتا ہے۔
- پابندی کا جائزہ: ایک ایسا عمل جس میں کاروبار کی جانچ کی جاتی ہے کہ وہ تمام مطلوبہ قوانین اور قواعد کی پیروی کر رہا ہے۔

مثال: اگر کوئی سکول ایک آن لائن پورٹل شروع کرنا چاہتا ہے جہاں طلباء اپنا ہوم ورک، نمبرز اور حاضری (اٹینڈنس) چیک کر سکیں۔

پورٹل استعمال کرنے کے لیے، طلباء کو اپنا نام، رول نمبر، اور رابطہ کی تفصیلات درج کرنی ہوں گی۔

سکول اعداد و شمار کی رازداری کے قواعد پر اس طرح عمل کرے گا:

- صرف وہ معلومات اکٹھی کرنا جو پورٹل کے لیے ضروری ہے۔
- معلومات کو صرف سکول کے مقاصد کے لیے استعمال کرنا، جیسے ہوم ورک کی تازہ ترین معلومات یا امتحانات کے نظام الاوقات بھیجنا۔
- معلومات کو پاس ورڈز کے ساتھ محفوظ کرنا تاکہ کوئی دوسرا اس تک رسائی نہ کر سکے۔

اگر اس اسکول کے طالب علم یورپ یا امریکہ سے ہوتے، تو اسے GDPR اور COPPA قانون کی پیروی بھی کرنی پڑتی، کیونکہ یہ ان ممالک کے لوگوں کے ذاتی اعداد و شمار کی حفاظت کرتے ہیں۔

کیا آپ جانتے ہیں؟

2019 میں، امریکہ میں گوگل اور یوٹیوب پر 170 ملین ڈالر کا جرمانہ عائد کیا گیا تھا کیونکہ انہوں نے بچوں کے آن لائن رازداری کے تحفظ کا قانون (COPPA) کے قواعد کی خلاف ورزی کرتے ہوئے والدین کی اجازت کے بغیر بچوں کا ذاتی ڈیٹا جمع کیا تھا۔ یہ ظاہر کرتا ہے کہ دنیا بھر میں اعداد و شمار کی رازداری کے قوانین کو کتنی سختی سے نافذ کیا جاتا ہے۔

8.4 لیگل فریم ورک اور کمپلائنس

لیگل فریم ورک اور کمپلائنس کا مطلب ہے وہ قوانین اور قواعد جن کی پیروی کسی کاروبار کو قانونی طور پر چلانے کے لیے کرنا ضروری ہے۔ اس میں کاروبار کی رجسٹریشن، ٹیکس کی ادائیگی، صارفین کے اعداد و شمار کا تحفظ، اور صنعت کے تقاضوں کو پورا کرنا شامل ہے۔

ذیل میں وہ قواعد ہیں جو کسی بھی کاروبار کے لیے قانونی مسائل سے بچاؤ کاروبار کی مارکیٹ میں اعتماد پیدا کرتے ہیں۔

پاکستان میں کاروباری رجسٹریشن

پاکستان میں کاروبار شروع کرنا صرف ایک اچھے آئیڈیا تک محدود نہیں ہے۔ اسے حکومت کے ساتھ رجسٹر کروا کر قانونی بنانا بھی ضروری ہے۔ رجسٹریشن سے کاروبار قانونی ہو جاتا ہے، اسے قانونی مسائل کے بغیر کام کرنے کی اجازت ملتی ہے، اور صارفین، بینکوں، اور دوسرے

کاروباروں کا اعتماد حاصل کرنے میں مدد ملتی ہے۔

ایس ای سی پی (SECP) سے رجسٹریشن

سیکیورٹیز اینڈ ایکسچینج کمیشن آف پاکستان (SECP) پاکستان میں کسی بھی نئے کاروبار کو رجسٹر کرنے والا ادارہ ہے۔ کاروبار کو رجسٹر کرنے کا عمل درج ذیل مراحل پر مشتمل ہے۔

- ایک منفرد کاروباری نام کا انتخاب کریں اور اسے ایس ای سی پی کی ویب سائٹ (https://secp.gov.pk/) کے ذریعے منظور کروائیں۔
- مطلوبہ دستاویزات تیار کریں، جیسے کمپنیوں کے لیے میمورنڈم اور آرٹیکلز آف ایسوسی ایشن یا شراکت داریوں کے لیے شراکت داری کا معاہدہ۔
- آن لائن درخواست دیں۔
- اندراج کا سرٹیفکیٹ حاصل کریں، جو ثابت کرتا ہے کہ کاروبار باضابطہ طور پر رجسٹر ہو چکا ہے۔

قومی ٹیکس نمبر (NTN) حاصل کرنا

NTN (نیشنل ٹیکس نمبر) پاکستان میں ٹیکس دہندگان کے لیے ایک شناختی نمبر ہے۔ یہ فیڈرل بورڈ آف ریونیو (FBR) کی طرف سے جاری کیا جاتا ہے۔ NTN حاصل کرنے کے لیے:

- ایف بی آر کے "آئرس" آن لائن پورٹل کے ذریعے درخواست دیں۔
- ایس ای سی پی کی رجسٹریشن دستاویزات، پتے کا ثبوت، اور کاروباری سرگرمی کی تفصیلات فراہم کریں۔
- منظوری کے بعد، NTN کاروبار کو بینک اکاؤنٹ کھولنے، رسیدیں جاری کرنے، اور ٹیکس کی ذمہ داریاں پوری کرنے کی اجازت دیتا ہے۔

ٹیکس گوشوارے جمع کروانا اور ریکارڈ رکھنا

رجسٹرڈ کاروباری اداروں کو چاہیے کہ:

- سالانہ انکم ٹیکس گوشوارے اور اگر قابل اطلاق ہو تو ماہانہ سیلز ٹیکس گوشوارے جمع کروائیں۔
- فروخت، خریداریوں، تنخواہوں اور اخراجات کا درست ریکارڈ رکھیں۔
- تاخیر یا غلط فائلنگ سے گریز کریں، کیونکہ ان کی وجہ سے جرمانے یا قانونی کارروائی ہو سکتی ہے۔

صنعت کے لیے مخصوص ضوابط

پاکستان میں کچھ کاروباری اداروں کو عام رجسٹریشن اور ٹیکس کے تقاضوں کے علاوہ خصوصی قواعد کی پیروی کرنی پڑتی ہے۔
مثال: اپنی کلاس کے دو طلباء، مسٹر ایکس اور مسٹر وائی، طلباء کے لیے ایک "آن لائن کتابوں کی دکان" شروع کرنا چاہتے ہیں۔

1. ایس ای سی پی (SECP) رجسٹریشن: انہوں نے اپنے کاروبار کے لیے ایک بامعنی اور پیشہ ورانہ نام جیسے "ای-بک ہب" (E-Book Hub) کا انتخاب کیا اور اسے ایس ای سی پی کی ویب سائٹ پر منظور کروالیا۔ وہ تمام مطلوبہ دستاویزات آن لائن جمع کرواتے ہیں اور اندراج کا سرٹیفکیٹ حاصل کرتے ہیں۔
2. NTN کا حصول: وہ اپنے ایس ای سی پی (SECP) کے دستاویزات، پتے، اور کاروباری تفصیلات کا استعمال کرتے ہوئے ایف بی آر (FBR) کے آئرس (IRIS) پورٹل کے ذریعے قومی ٹیکس نمبر (NTN) کے لیے درخواست دیتے ہیں۔
3. ٹیکس فائلنگ: وہ فروخت اور اخراجات کا مناسب ریکارڈ رکھتے ہیں اور اپنے سالانہ ٹیکس گوشوارے وقت پر جمع کرواتے ہیں۔
4. صنعتی قواعد: ایک ای-کامرس کاروبار کے طور پر، وہ پاکستان کی ای-کامرس پالیسی کے قواعد پر عمل کرنے کے لیے اپنی ویب سائٹ پر خدمات کی شرائط، رازداری کی پالیسی، اور واپسی / رقم کی واپسی کے قواعد شامل کرتے ہیں۔

ان مراحل کو مکمل کرکے، مسٹر ایکس (X) اور مسٹر (Y) وائی اپنی نئی آن لائن کتابوں کی دکان کو قانونی طور پر شروع کرتے ہیں، جسے حکومت نے منظور کرلیا ہے، جو صارفین اور سپلائرز کے لیے زیادہ قابل اعتماد ہے۔

کیا آپ جانتے ہیں؟

جب ہم کوئی نیا کاروبار رجسٹر کرتے ہیں، تو ہمیں اوپر بیان کی گئی دستاویزات کے مقاصد کو سمجھنے کی ضرورت ہوتی ہے:

i. میمورنڈم ایسوسی ایشن: کسی کمپنی کی بنیادی تفصیلات، جیسے اس کا نام، مقصد، اور مقام۔
ii. آرٹیکلز آف ایسوسی ایشن: کمپنی چلانے کے اندرونی قواعد۔
iii. شراکت داری کا معاہدہ: دو یا دو سے زیادہ افراد کے درمیان ایک معاہدہ جو مل کر کاروبار شروع کرتے ہیں، اور ہر فرد کا حصہ، کردار، اور ذمہ داریاں بیان کرتا ہے۔

ڈیجیٹل بزنس میں معاہدات

معاہدہ ایک تحریری یا زبانی رضامندی (Agreement) ہے جو دو یا دو سے زیادہ افراد یا تنظیموں کے درمیان ہوتا ہے اور قانونی طور پر لازمی ہوتا ہے۔ ڈیجیٹل بزنس میں معاہدات اہم ہیں کیونکہ وہ مل کر کام کرنے کی شرائط وضوابط طے کرتے ہیں، چاہے وہ کاروباری شراکت داروں، (سپلائرز یا صارفین) کے ساتھ ہو۔ یہ غلط فہمیوں کو روکنے میں مدد دیتے ہیں اور اگر کوئی اختلاف رائے پیدا ہو جائے تو قانونی تحفظ فراہم کرتے ہیں۔

ایک درست معاہدے کے بنیادی عناصر

کسی معاہدے کے درست ہونے کے لیے، اس میں درج ذیل شامل ہونا ضروری ہے:

- پیشکش: ایک فریق کچھ کرنے کا وعدہ کرتا ہے، جیسے کوئی مصنوعات فروخت کرنا یا خدمت فراہم کرنا۔
- قبولیت: دوسرا فریق بغیر کسی تبدیلی کے اس پیشکش سے متفق ہو جاتا ہے۔

- عوض / معاوضہ : کوئی قابلِ قدر چیز کا تبادلہ کیا جاتا ہے، جیسے رقم، سامان یا خدمات۔
- قانونی مقصد : معاہدہ ایک قانونی معاملے کے بارے میں ہونا چاہیے۔
- رضامندی : ہر کوئی اپنی مرضی سے راضی ہو، بغیر کسی دباؤ یا دھوکہ دہی کے۔

ڈیجیٹل بزنس میں اہمیت

ڈیجیٹل بزنس میں، معاہدات پر کاغذ یا ڈیجیٹل طور پر دستخط کیے جا سکتے ہیں۔ وہ ان مقاصد کے لیے استعمال ہوتے ہیں:

- شراکت داریاں : واضح کردار، ذمہ داریاں اور منافع کو کس طرح تقسیم کیا جائے گا، یہ طے کرنا۔
- سپلائرز (فراہم کنندگان): ترسیل کے وقت، ادائیگی کی شرائط اور معیار کے معیار کو مقرر کرنا۔
- آن لائن لین دین : خدمات کی شرائط، رقم کی واپسی کے قواعد، اور صارف کے حقوق کی وضاحت کرنا۔

واضح معاہدات اعتماد پیدا کرتے ہیں اور کاروباری معاملات کو آسان اور محفوظ بناتے ہیں۔

مثال : دسویں جماعت کے طلباء ایک الوداعی پارٹی کا انتظام کرنے کا منصوبہ بنا رہے ہیں۔ مسٹر (X) ایکس پارٹی کے انتظامات سنبھالے گا۔ اُسے ایک معاہدہ کرنے کا کہا جاتا ہے۔

- پیشکش : مسٹر (X) ایکس پارٹی کے لیے کھانے اور مشروبات کا انتظام کرنے کی پیشکش کرتا ہے۔
- قبولیت : مینجمنٹ اُس کی آفر قبول کرے گی۔
- عوض / معاوضہ : سکول کل اخراجات کے لیے 1,500 روپے دے گا۔
- قانونی مقصد : اسکول کی اجازت سے کلاس پارٹی کا اہتمام قانونی طور پر کرنا۔
- رضامندی : دونوں فریقین بغیر کسی دباؤ کے اپنی مرضی سے رضامندی کا اظہار کرتے ہیں۔

معاہدے کی شرائط:

- مسٹر (X) ایکس تقریب سے تین دن پہلے کھانے کا بندوبست کرے گا۔
- کھانے کی اشیاء وقت پر ڈیلیور کی جائیں گی۔

سرگرمی:

مقصد:

طلباء گروپس میں کام کریں گے تاکہ "اسکول کی کینٹین" کے لیے ایک چھوٹے کاروبار کا آئیڈیا بنائیں اور یقینی بنائیں کہ وہ قانونی ڈھانچے اور پابندی کے قواعد پر عمل کرتا ہے۔ وہ ہر رکن کی ذمہ داریوں کی وضاحت کے لیے ایک سادہ معاہدہ بھی تیار کریں گے۔

طریقہ کار:

i. گروپ بنانا : کلاس کو 5-4 طلباء کے گروپس میں تقسیم کریں۔

ii. کاروباری خیال : ہر گروپ یہ تصور کرے گا کہ وہ اسکول کی کینٹین میں ایک چھوٹا کاروبار شروع کر رہا ہے (مثلاً، سنیک کاؤنٹر، جوس بار، اسٹیشنری کا گوشہ وغیرہ)۔

iii. ایک سادہ معاہدہ تیار کرنا:

گروپ کے دو ارکان کے درمیان ایک بنیادی معاہدہ لکھیں جو درج ذیل کو ظاہر کرے:
- پیشکش : ہر رکن کیا کرے گا۔
- قبولیت : منصوبے پر رضامندی۔
- عوض / معاوضہ : ہر رکن کیا حصہ ڈالے گا (مثلاً، رقم، وقت، یا سامان)۔
- قانونی مقصد : اسکول میں قانونی سرگرمی۔
- رضامندی : بغیر کسی دباؤ کے باہمی رضامندی۔

8.5 فکری ملکیت کے حقوق (Intellectual Property Rights)

فکری ملکیت کے حقوق وہ قانونی حقوق ہیں جو کسی شخص یا کاروبار کی تخلیقات کو محفوظ رکھتے ہیں۔ ان تخلیقات میں ایجادات، کتابیں، گانے، آرٹ ورک، برانڈ کے نام، یا لوگو (Logo) شامل ہو سکتے ہیں۔ فکری ملکیت کے حقوق (IPR) یقینی بناتے ہیں کہ کوئی بھی ان تخلیقات کو اجازت کے بغیر استعمال یا نقل نہیں کر سکتا۔ یہ تحفظ لوگوں کو تخلیقی ہونے کی ترغیب دیتا ہے اور کاروباری اداروں کو اپنی منفرد شناخت برقرار رکھنے میں مدد کرتا ہے۔

پاکستان میں فکری ملکیت کا اندراج

پاکستان میں فکری ملکیت کو رجسٹر کرنے کے لیے درج ذیل بنیادی باتیں ضروری ہیں:

i. تجارتی نشانات (ٹریڈ مارکس): برانڈ کے ناموں، لوگو، یا نشانات کو محفوظ رکھتے ہیں جو سامان یا خدمات کی شناخت کے لیے استعمال ہوتے ہیں۔
- تنظیم برائے فکری ملکیت پاکستان (IPO Pakistan) میں درخواست دیں۔
- نام یا لوگو، سامان یا خدمات کی تفصیل، اور مطلوبہ فیس کے ساتھ درخواست دیں۔
- منظوری کے بعد، تجارتی نشان دوسروں کو اسی طرح کا نام یا لوگو استعمال کرنے سے روکتا ہے۔

ii. حقوقِ تصنیف (کاپی رائٹس): اصل تخلیقی کاموں جیسے کتابیں، گانے، کمپیوٹر پروگرام، یا ڈرائنگ کو محفوظ رکھتے ہیں۔
- آئی پی او (IPO) پاکستان کے تحت حقوقِ تصنیف کے دفتر میں درخواست دیں۔
- کام کی تفصیلات، یہ ثابت کرنے کا ثبوت کہ آپ نے اسے تخلیق کیا ہے، اور کام کی ایک نقل بھیجیں۔
- کام کی تخلیق کے وقت حقوقِ تصنیف خود بخود مل جاتے ہیں، لیکن اندراج زیادہ مضبوط قانونی تحفظ فراہم کرتا ہے۔

iii. پیٹنٹس (Patents): نئی مصنوعات، ڈیزائن، یا طریقہ کار کو قانونی تحفظ دیتے ہیں جسے کسی نے ایجاد کیا ہو۔ اس کا مطلب ہے کہ ایک مخصوص وقت کے لیے صرف موجد ہی اسے بنا، استعمال یا فروخت کر سکتا ہے۔
- آئی پی او (IPO) پاکستان کے تحت پیٹنٹ آفس میں درخواست دیں۔
- ایجاد کی تفصیلات، ڈرائنگ، اور وضاحتیں جمع کروائیں۔
- منظوری کے بعد، ایک مقررہ مدت کے لیے صرف موجد ہی ایجاد کو بنا، استعمال یا فروخت کر سکتا ہے۔

ڈیجیٹل دور میں تجارتی رازوں کا تحفظ

تجارتی راز وہ نجی کاروباری معلومات ہیں جو کاروباری افراد یا اداروں کو فائدہ دیتی ہیں، جیسے کوئی ترکیب، کوئی فارمولا، یا صارفین کی فہرست۔

- ان معلومات کو محفوظ رکھیں، خواہ وہ کاغذ پر ہو یا ڈیجیٹل شکل میں۔
- پاس ورڈز، خفیہ کاری (Encryption) کا استعمال کریں، اور صرف مجاز افراد تک رسائی کو محدود کریں۔
- ملازمین یا شراکت داروں سے ایک راز افشاء نہ کرنے کا معاہدہ (Non-Disclosure Agreement - NDA) پر

سرگرمی:

مقصد:

کاروباری لوگو (Logo) کی تخلیق کی مشق کرنا اور نقلی تجارتی نشان کی درخواست تیار کرنا، تاکہ یہ سمجھا جاسکے کہ قانونی تحفظ دوسروں کو کاروباری نام یا لوگو کی نقل کرنے سے کیسے روکتا ہے۔

طریقہ کار:

- اپنی کینٹین کے کاروبار کے لیے ایک منفرد لوگو (Logo) ڈیزائن کریں۔
- ایک نقلی تجارتی نشان کی درخواست تیار کریں جس میں شامل ہو:
- کاروبار کا نام۔
- کینٹین کی طرف سے فراہم کیے جانے والے کھانے یا خدمات کی ایک مختصر تفصیل۔
- لوگو (Logo) کی ایک ڈرائنگ یا چھپی ہوئی نقل۔
- درخواست کو کلاس میں پیش کریں اور وضاحت کریں کہ تجارتی نشان رکھنے سے کاروباری نام اور لوگو (Logo) کو دوسروں کی طرف سے نقل کیے جانے سے کیسے تحفظ ملے گا۔

SMART TIPS سمارٹ ٹپس

اپنے کاروباری نام اور لوگو (Logo) کو بازار میں اپنی شناخت سمجھیں؛ انہیں تخلیقی، آسانی سے پہچاننے کے قابل، اور بامعنی بنائیں۔ ایک مضبوط تجارتی نشان نہ صرف آپ کی شناخت کو نقل ہونے سے بچاتا ہے بلکہ صارفین کا اعتماد بھی بڑھاتا ہے، تشہیر میں معاونت کرتا ہے، اور آپ کے کاروبار کو دیرپا مسابقتی برتری (competitive edge) فراہم کرتا ہے۔

8.6 حکمت عملی پر مبنی مارکیٹنگ اور برانڈنگ

حکمت عملی پر مبنی مارکیٹنگ کسی کاروبار کو فروغ دینے کے لیے سرگرمیوں کی منصوبہ بندی اور ان پر عمل درآمد کا عمل ہے تاکہ صارفین کو اپنی طرف متوجہ کیا جاسکے اور انہیں وفادار رکھا جاسکے۔ آج کے ڈیجیٹل دور میں، کاروبار بہتر مارکیٹنگ کی حکمت عملیوں کو ڈیزائن کرنے، اپنے برانڈ کو مضبوط بنانے، اور طویل مدتی تعلقات استوار کرنے کے لیے ڈیٹا، ٹیکنالوجی اور کسٹمر کی بصیرتوں کا استعمال کرتے ہیں۔

ڈیٹا کی بنیاد پر مارکیٹنگ

ڈیٹا کی بنیاد پر مارکیٹنگ کا مطلب ہے کہ اندازہ لگانے کے بجائے حقائق، اعداد و شمار، اور تجزیہ کی بنیاد پر مارکیٹنگ کے فیصلے کرنا۔ کاروبار مؤثر مہمات (Campaigns) کی منصوبہ بندی کے لیے صارفین، ویب سائٹس، اور فروخت کے ریکارڈ سے معلومات جمع کرتے ہیں۔

قابل پیمائش اہداف اور KPIs (کلیدی کارکردگی کے اشارے) کا تعین

مثال: لاہور میں ایک آن لائن کپڑوں کی دکان تین ماہ کے اندر ٹی شرٹس کی فروخت میں 20% اضافہ کا ہدف مقرر کرتی ہے۔ وہ ہفتہ وار فروخت ہونے والی ٹی شرٹس کی تعداد، ویب سائٹ پر آنے والے صارفین، اور کسٹمر کے جائزوں (Reviews) جیسے KPIs کا استعمال کرتے ہوئے پیش رفت کو ٹریک کرتے ہیں۔

تجزیاتی ٹولز کا استعمال

مثال: کراچی کی ایک بیکری گوگل اینالیٹکس کا استعمال یہ دیکھنے کے لیے کرتی ہے کہ کون سے فیس بک اشتہارات ان کی ویب سائٹ پر سب سے زیادہ کسٹمرز لاتے ہیں۔ وہ دیکھتے ہیں کہ سالگرہ کے کیک دکھانے والے اشتہارات کو عام اشتہارات کے مقابلے میں زیادہ کلکس ملتے ہیں، لہٰذا وہ سالگرہ سے متعلق پروموشنز پر زیادہ توجہ دیتے ہیں۔

سرمایہ کاری پر واپسی کی پیمائش ROI (Return on Investment)

مثال: ایک چھوٹی کتابوں کی دکان انسٹاگرام اشتہارات پر 10,000 روپے خرچ کرتی ہے اور ان اشتہارات سے 25,000 روپے کماتی ہے۔ لاگت کا موازنہ منافع سے کر کے، وہ تصدیق کرتے ہیں کہ مہم کامیاب رہی اور اسی حکمت عملی کو دہرانے کا فیصلہ کرتے ہیں۔

کسٹمر ریلیشن شپ مینجمنٹ (CRM)

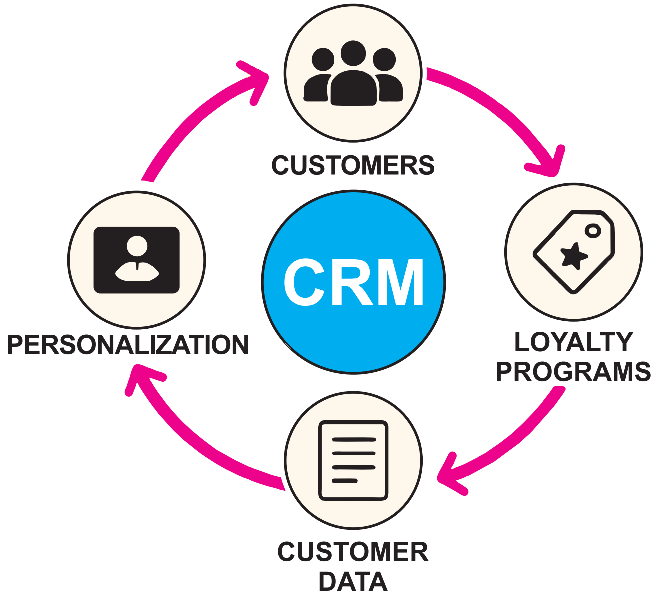
CRM ایک طریقہ ہے جس میں صارفین کے ساتھ تمام (میل جول/معملات) (Interactions) کو محفوظ کیا جاتا ہے تاکہ اعتماد پیدا کیا جاسکے اور طویل مدتی تعلقات قائم کیے جاسکیں۔ ایک اچھا CRM طریقہ کار کاروبار کو اپنے صارفین کو بہتر طریقے سے سمجھنے اور زیادہ بہتر انفرادی سے ان کی خدمت کرنے میں مدد کرتا ہے۔

CRM ٹولز کیسے کام کرتے ہیں؟

مثال: اسلام آباد میں ایک سپورٹس شاپ زوہو سی آر ایم (Zoho CRM) کا استعمال کرتی ہے تاکہ صارفین کے نام، رابطے کی تفصیلات، جوتے کے سائز، اور خریداری کی تاریخ کو ایک ہی سسٹم میں محفوظ کرے۔ جب اسی سائز کا کوئی نیا سپورٹس شوز (Shoes) آتا ہے جو گاہک عام طور پر خریدتا ہے، تو دکان اسے فوری پیغام یا ای میل بھیج سکتی ہے۔

صارفین کو برقرار رکھنا: (Retaining Customers) مثال: کراچی کا ایک کافی شاپ ایک لائلٹی پروگرام استعمال کرتا ہے جہاں صارفین کو ہر 10 خریداریوں کے بعد ایک مفت مشروب ملتا ہے۔ وہ صارف کا نام اور اس کے پسندیدہ مشروب کی پیشکش کے ساتھ ذاتی

نوعیت کی ای میلز بھی بھیجتے ہیں، جس سے صارفین کو قدر کی نگاہ سے دیکھا جاتا ہے اور ان کے واپس آنے کا امکان بڑھ جاتا ہے۔



شکل 8.1: سی آر ایم (CRM) کسٹمر ڈیٹا کو دیرپا تعلقات قائم کرنے کے لیے منظم کرتا ہے۔

کیا آپ جانتے ہیں؟

زوہو سی آر ایم (Zoho CRM) ایک کلاؤڈ پر مبنی سافٹ ویئر ہے جو 180 سے زائد ممالک میں 25,000 سے زیادہ کاروباروں کے ذریعے استعمال کیا جاتا ہے تا کہ چھوٹے سٹارٹ اپس سے لے کر بڑی کمپنیوں تک کسٹمر سے تعلقات کو منظم کیا جاسکے، جس سے انہیں سیلز بڑھانے اور کسٹمر کے اطمینان کو بہتر بنانے میں مدد ملتی ہے۔

مارکیٹنگ آٹومیشن

مارکیٹنگ آٹومیشن کا مطلب ہے کہ مارکیٹنگ کے کاموں کو انسانی ہاتھ سے کیے بغیر، خصوصی سافٹ ویئر کا استعمال کرتے ہوئے خودکار طریقے سے سنبھالنا۔ اس سے کاروباروں کو وقت بچانے، زیادہ مؤثر طریقے سے کام کرنے، اور صارفین کے ساتھ جڑے رہنے میں مدد ملتی ہے۔

مثالیں:

- میل چیمپ (Mailchimp) نئے صارفین کو خود بخود خوش آمدیدی ای میلز بھیج سکتا ہے، خصوصی پیشکشیں شیئر کر سکتا ہے اور سوشل میڈیا پر اپ ڈیٹس شائع کر سکتا ہے۔
- ہب سپاٹ (HubSpot) کسٹمر کی سرگرمی کو ٹریک کر سکتا ہے، فالو اپ پیغامات بھیج سکتا ہے اور مارکیٹنگ کی کارکردگی پر رپورٹس تیار کر سکتا ہے۔

چھوٹے کاروباروں کے لیے فوائد:

- معمول کے کاموں کو خودکار طریقے سے سنبھال کر وقت کی بچت کرتا ہے۔
- اضافی عملے کی ضرورت کو کم کر کے اخراجات میں کمی لاتا ہے۔
- صارفین کو بروقت اپ ڈیٹس بھیجتا ہے، جس سے کسٹمر انگیجمنٹ بہتر ہوتی ہے۔
- کم کوشش کے ساتھ ایک وسیع سامعین تک پہنچنے میں مدد کرتا ہے۔
- یہ دیکھنے کے لیے مہم کے نتائج کو ٹریک کرتا ہے کہ کیا سب سے بہتر کام کر رہا ہے۔

SMART TIPS

سمارٹ ٹپس

ایک برانڈ صرف ایک لوگو (Logo) نہیں ہے؛ یہ وہ احساس ہے جو صارفین آپ کے کاروبار کے بارے میں رکھتے ہیں۔ ہر پیغام، پراڈکٹ اور سروس اس احساس کو شکل دیتی ہے۔

سرگرمی:

مقصد:

آپ کے گروپ نے سکول کی ٹک شاپ میں فروخت کرنے کے لیے ایک نیا اسنیک تیار کیا ہے۔ آپ کے پاس اس کی برانڈنگ اور مارکیٹنگ کا منصوبہ بنانے کے لیے 20 منٹ ہیں۔

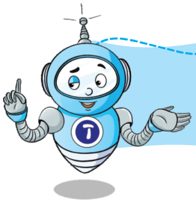
اقدامات:

1. نام اور لوگو (Logo): ایک پرکشش پراڈکٹ کا نام منتخب کریں اور ایک سادہ لوگو (Logo) بنائیں۔
2. برانڈ پیغام: ایک جملہ لکھیں جو وضاحت کرے کہ طلباء کو اسے کیوں خریدنا چاہیے۔
3. مارکیٹنگ کا طریقہ: اسے فروغ دینے کا ایک طریقہ منتخب کریں (پوسٹر، سلوگن، یا سوشل میڈیا پوسٹ)۔
4. کسٹمر انگیجمنٹ: صارفین کو واپس لانے کے لیے ایک طریقہ کار کی منصوبہ بندی کریں (رعایتیں، مفت نمونے، یا ایک کوئز)۔

نتائج: ہر گروپ اپنا منصوبہ 2 منٹ میں پیش کرے گا۔ کلاس پراڈکٹ کے سب سے پرکشش خیال کے لیے ووٹ دے گی۔

خلاصہ (Summary)

- کاروباری دنیا میں چیلنجز کسی بھی وقت سامنے آسکتے ہیں۔ ان میں گاہکوں کا کھو جانا، نئے حریفوں کا سامنا کرنا، یا ٹیکنالوجی میں تبدیلیاں شامل ہو سکتی ہیں۔
- موافقت پذیری (Adaptability) کا مطلب ضرورت پڑنے پر تبدیلی کے لیے تیار اور قابل ہونا ہے۔
- لچک (Resilience) کا مطلب یہ ہے کہ کوئی کاروبار کسی ناکامی یا دھچکے کے بعد تیزی سے بحال ہو سکے۔
- اخلاقی فیصلہ سازی کا مطلب کاروبار میں ایسے انتخاب کرنا ہے جو منصفانہ، ایماندارانہ اور ذمہ دارانہ ہوں۔
- مفادات کا ٹکراؤ (Conflict of interest) اس وقت ہوتا ہے جب کسی شخص کا ذاتی فائدہ یا تعلق اس کے کاروباری فیصلوں پر اثر انداز ہو سکتا ہو۔
- ڈیجیٹل اخلاقیات کا مطلب انٹرنیٹ اور ڈیجیٹل آلات کا ذمہ دارانہ اور احترام کے ساتھ استعمال کرنا ہے۔
- ذمہ دارانہ مارکیٹنگ سے مراد مصنوعات اور خدمات کی دیانتداری اور منصفانہ طور پر تشہیر کرنا، گاہکوں کو درست اور مکمل معلومات فراہم کرنا، اور جھوٹے وعدوں سے بچنا ہے۔
- ڈیٹا پرائیویسی کے قوانین وہ قواعد ہیں جو کاروبار کے ذریعے جمع کی گئی ذاتی معلومات جیسے نام، پتے، فون نمبر، یا ادائیگی کی تفصیلات کی حفاظت کرتے ہیں۔
- NTN پاکستان میں ٹیکس دہندگان کے لیے ایک شناختی نمبر ہے جو فیڈرل بورڈ آف ریونیو (FBR) جاری کرتا ہے۔
- پاکستان میں کچھ کاروباروں کو عام رجسٹریشن اور ٹیکس کی ضروریات کے علاوہ خصوصی قواعد پر عمل کرنا پڑتا ہے۔
- معاہدہ (Contract) دو یا دو سے زیادہ افراد یا تنظیموں کے درمیان ایک تحریری یا زبانی اتفاق ہے جو قانونی طور پر پابند ہوتا ہے۔
- ڈیٹا پر مبنی مارکیٹنگ کا مطلب محض اندازوں کے بجائے حقائق، اعداد و شمار اور تجزیے کی بنیاد پر مارکیٹنگ کے فیصلے کرنا ہے۔
- CRM گاہکوں کے ساتھ تمام معاملات کو سنبھالنے کا ایک طریقہ ہے تاکہ اعتماد پیدا کیا جا سکے اور طویل مدتی تعلقات استوار ہوں۔
- مارکیٹنگ آٹومیشن کا مطلب خاص سافٹ ویئر کا استعمال کرتے ہوئے مارکیٹنگ کے کاموں کو خودکار طریقے سے انجام دینا ہے، تاکہ ہر قدم دستی طور پر نہ کرنا پڑے۔



مشق

کثیر انتخابی سوالات

i دانشورانہ ملکیت (Intellectual Property) کی ایک مثال کون سی ہے؟

الف دکان کا فرنیچر
ب برانڈ کا لوگو (Logo)
ج خام مال
د دفتر کا کرایہ

ii مندرجہ ذیل میں سے کون سا قانون یورپی یونین میں ذاتی ڈیٹا کی حفاظت کرتا ہے؟

الف (COPPA) کوپا
ب (GDPR) جی ڈی پی آر
ج (DMCA) ڈی ایم سی اے
د (Cybercrime Act) سائبر کرائم ایکٹ

iii ٹریڈ مارک کو رجسٹر کرنے کے لیے کون سی دستاویز درکار ہے؟

الف ایک بینک اسٹیٹمنٹ
ب پروڈکٹ کی خریداری کا ثبوت
ج لوگو (Logo) اور کاروبار کے نام کی ایک نقل
د سوشل میڈیا پروفائل لنک

iv مندرجہ ذیل میں سے کون سا ڈیٹا کی ناقص پرائیویسی کا ایک خطرہ ہے؟

الف کسٹمر کا بڑھتا ہوا اعتماد
ب سائبر بلنگ (Bulleying)
ج شناختی چوری
د اشتہارات کی کم لاگت

v مندرجہ ذیل میں سے کون سی ذاتی ڈیٹا کی ایک مثال ہے؟

الف دفتر کا پتہ
ب ملازم کا شناختی کارڈ نمبر
ج اسٹور کھلنے کے اوقات
د ویب سائٹ تھیم کا رنگ

vi تضادِ مفادات (A conflict of interest) ہوتا ہے جب:

الف ملازمین کمپنی کے اہداف کا اشتراک کرتے ہیں
ب ذاتی فائدہ کاروباری فیصلے کو متاثر کرے
ج ایک کاروبار منافع میں اضافہ کرتا ہے
د کسٹمرز فیڈبیک دیتے ہیں

vii مندرجہ ذیل میں سے کون سا ڈیجیٹل ایتھکس (Ethics) کا حصہ ہے؟

الف اشتہارات میں جھوٹی معلومات کا استعمال
ب کسٹمر ڈیٹا کو احترام کے ساتھ محفوظ رکھنا
ج سائبر بلنگ (Bulleying)
د آن لائن موجودگی سے گریز کرنا

viii مندرجہ ذیل میں سے کون سا سی آر ایم (CRM) ٹول ہے؟

الف فوٹوشاپ
ب زوہو (Zoho)
ج گوگل میپس (Google Maps)
د کینوا (Canva)

ix مندرجہ ذیل میں سے کون سا ٹول مارکیٹنگ آٹومیشن کے لیے استعمال ہوتا ہے؟

الف میل چی مپ (Mailchimp)
ب پاور پوائنٹ
ج ایم ایس ورڈ
د پینٹ

x مارکیٹنگ میں، آر او آئی (RoI) کا مطلب ہے:

الف سرمایہ کاری پر واپسی
ب معلومات کی شرح
ج ایم ایس ورڈ
د پینٹ

مختصر سوالات

1 ۔ ڈیجیٹل دنیا میں کاپی رائٹ تخلیق کاروں کی حفاظت کیسے کرتا ہے؟

2 پیٹنٹ (Patent) اور ٹریڈمارک (Trademark) میں فرق واضح کریں۔

3 ڈیٹا پرائیویسی کی تعریف کریں۔

4 کاروبار میں اخلاقی فیصلہ سازی کی ایک مثال دیں۔

5. کاروباری فیصلوں میں ایمانداری کیوں ضروری ہے؟
6. ڈیٹا پر مبنی مارکیٹنگ کا مقصد بیان کریں۔
7. SWOT تجزیہ کس چیز کی شناخت میں مدد کرتا ہے؟
8. مارکیٹنگ میں KPIs متعین کرنے کا مقصد واضح کریں۔

تفصیلی سوالات

1. مسئلہ حل کرنے اور لچک پذیری کے عمل کی وضاحت کریں۔
2. اخلاقی کاروباری طریقوں اور ڈیجیٹل اخلاقیات پر ایک نوٹ لکھیں۔
3. بحث کریں کہ سی آر ایم (CRM) اور لائلٹی پروگرام کس طرح صارفین کو برقرار رکھنے میں مدد کرتے ہیں۔
4. مارکیٹنگ آٹومیشن کی وضاحت کریں اور چھوٹے کاروباروں کے لیے اس کے فوائد بیان کریں۔

جوابات

1. ب — برانڈ کا لوگو (Logo)
2. ب — (GDPR) جی ڈی پی آر
3. ج — لوگو (Logo) اور کاروبار کے نام کی ایک نقل
4. ج — شناختی چوری
5. ب — ملازم کا شناختی کارڈ نمبر
6. ب — ذاتی فائدہ کاروباری فیصلے کو متاثر کرے
7. ب — کسٹمر ڈیٹا کو احترام کے ساتھ محفوظ رکھنا
8. ب — (Zoho) زوہو
9. الف — میل چمپ (Mailchimp)
10. الف — سرمایہ کاری پر واپسی

امتحانی پرچہ کمپیوٹر سائنس اینڈ انٹر پرینور شپ

برائے جماعت دہم (X) کی تیاری کے

کمپیوٹر سائنس اینڈ انٹر پرینور شپ برائے جماعت دہم (X) کا پرچہ 50 نمبروں پر مشتمل ہو گا۔ پرچہ 30 حصوں (حصہ معروضی، حصہ مختصر سوالات اور حصہ تفصیلی سوالات) پر مشتمل ہو گا۔ حصہ معروضی طرز (کثیر الانتخابی سوالات) 10 نمبروں پر مشتمل ہو گا اور اس کا وقت 15 منٹ ہو گا۔ انشائی طرز دو حصوں (مختصر سوالات اور تفصیلی سوالات) پر مشتمل ہو گا۔ یہ حصہ 40 نمبر کا ہو گا اور اس کے لیے 1 گھنٹا 45 منٹ کا وقت مختص ہو گا۔ پرچہ مرتبین درج ذیل ہدایات کو ملحوظ رکھیں:۔

معروضی طرز (کثیر الانتخابی سوالات)	سوال نمبر 1: مکمل کتاب میں سے 10 کثیر الانتخابی سوالات کے جوابات درکار ہوں گے۔ -باب نمبر 1، 3، 4، 5، 6 اور 8 میں سے ایک ایک سوال پوچھا جائے گا۔ -باب نمبر 2 اور 7 میں سے دو ، دو سوالات پوچھے جائیں گے۔	$1 \times 10 = 10$
انشائی طرز	مختصر سوالات پرچہ میں مکمل کتاب میں سے مختصر سوالات کے حامل 30 سوالات (*سوال نمبر 2,3 اور 4*) ہوں گے۔	
	سوال نمبر 2: اس سوال کے تحت، طلبہ دیے گئے 2 مختصر سوالات میں سے کوئی سے 4 سوالات کے جوابات تحریر کریں گے۔ وضاحت: باب نمبر 1 میں سے 1، باب نمبر 2 میں سے 2 اور باب نمبر 3 میں سے 3 سوالات کا انتخاب کیا جائے۔	$2 \times 4 = 8$

	سوال نمبر3 : اس سوال کے تحت، طلبہ دیے گئے 6 مختصر سوالات میں سے کوئی سے 4 سوالات کے جوابات تحریر کریں گے۔ وضاحت: باب نمبر4 میں سے 2 اور باب نمبر5 میں سے 4 سوالات کا انتخاب کیا جائے۔	$2 \times 4 = 8$
	سوال نمبر4: اس سوال کے تحت، طلبہ دیے گئے 6 مختصر سوالات میں سے کوئی سے 4 سوالات کے جوابات تحریر کریں گے۔ وضاحت: باب نمبر6 میں سے 2 باب نمبر7 میں سے 2 اور باب نمبر8 میں سے 2 سوالات کا انتخاب کیا جائے۔	$2 \times 4 = 8$
تفصیلی سوالات	اس حصے کے تحت 30 تفصیلی سوالات دیے جائیں گے جن میں سے کوئی 20 سوالات کے جوابات درکار ہوں گے۔ وضاحت: سوال نمبر5: باب نمبر1 میں سے 1 سے 1 سوال کا انتخاب کیا جائے گا۔ سوال نمبر6: باب نمبر4 میں سے 1 سوال کا انتخاب کیا جائے گا۔ سوال نمبر7: باب نمبر6 یا 8 میں سے اسوال منتخب کیا جائے گا۔	$2 \times 8 = 16$

Model Paper

Computer Science and Entrepreneurship Class 10

Time Allowed: 15 minutes وقت:15منٹس

Multiple Choice Questions (MCQ درست آپشن کا انتخاب کریں۔

(i) ملٹی یوزر انوائرنمنٹ میں OS یقینی بناتا ہے:

(i) In multi-user environments, the OS ensures:

(A) Equal access and data privacy مساوی استعمال اور ڈیٹا پرائیویسی

(B) All users share the same files privacy تمام یوزر یکساں پرائیویسی فائل شیئر کرتے ہیں۔

(C) Hardware is never used ہارڈویئر کبھی استعمال نہیں ہوتا

(D) Internet speed is increased انٹرنیٹ کی سپیڈ بڑھ جاتی ہے۔

(ii) پاور آن سیلف ٹیسٹ (POST) کہاجاتا ہے:

(ii) The Power-On Self-Test (POST) is performed

(A) During system shutdown سسٹم کو بند کرتے وقت (B) After login لاگ اِن کے بعد

(C) At system startup سسٹم آن کرتے وقت (D) When installing software کوئی سافٹ ویئر انسٹال کرتے وقت

(iii) ونڈو ڈیفنڈر فراہم کرتا ہے:

(iii) Windows Defender provides

(A) File compression فائل کمپریشن (B) Virus protection وائرس سے بچاؤ

(C) Disk formatting ڈسک فارمیٹنگ (D) Music playback میوزک پلے بیک

(iv) اِن میں سے کونسا آپریٹر پائتھن میں قوت نمائی کے لیے استعمال کیا جاتا ہے:

(iv) Which of the following operators is used for exponentiation in Python?

(A) * (B) ** (C) // (D) #

(v) پائتھن میں لسٹ کے آخر پر ایک آئیٹم بڑھانے کے لیے اِن میں سے کون سا طریقہ استعمال کیا جاتا ہے:

(v) Which method is used to add an item at the end of the list in Python?

(A) insert() (B) append() (C) add() (D) extend()

(vi) بہت سے طلباء کی رائے لینے کے لیے بہترین طریقہ کون سا ہے؟

(vi) Which method is best for collecting opinions from many students?

(A) IoT Devices IoTڈیوائسز (B) Survey سروے

(C) Web Scraping ویب سکریپنگ (D) Database ڈیٹابیس

(vii) سپروائزرلرننگ میں ڈیٹا ہوتا ہے:

(vii) In supervised learning, the data is:

(A) Unorganized غیر منظم
(B) Without any answers بغیر کسی جواب کے
(C) Labeled with correct answers درست جواب کے لیبلز کے ساتھ
(D) Only text-based صرف ٹیکسٹ کی بنیاد پر

(viii) AI مخفف ہے:

(viii) What does AI stand for?

(A) Automatic Intelligence
(B) Artificial Intelligence
(C) Active Interface
(D) Applied Information

(ix) فیکٹریز میں ربوٹس کا کیا کردار ہے؟

(ix) What is the role of robots in factories?

(A) Watching videos ویڈیوز دیکھنا
(B) Making websites ویب سائٹ بنانا
(C) Doing repeated or dangerous tasks خطرناک یا دوہرائے جانے والے کام کرنا
(D) Writing stories کہانیاں لکھنا

(x) ٹریڈمارک کو رجسٹر کروانے کے لیے کون سا ڈاکومنٹ درکار ہوتا ہے؟

(x) Which document is required to register a trademark?

(A) A bank statement بنک سٹیٹمنٹ
(B) Proof of product purchase پراڈکٹ کی خریداری کا ثبوت
(C) A copy of the logo and business name بزنس کے نام اور لوگو کی کاپی
(D) Social media profile link سوشل میڈیا پروفائل کا لنک

سوال 2: کوئی سے چار سوالات کے مختصر جوابات تحریر کریں:

Question 2: Write Short answers to any Four (04) questions. (2x4=8)

(i) آپریٹنگ سسٹم (OS) کس طرح یوزر اور ہارڈویئر کے مابین مترجم کا کردار ادا کرتا ہے؟

(i) How does the OS act as a translator between the user and hardware?

(ii) ورچوئل میموری ریم سے سست رفتار کیوں ہے؟

(ii) Why is virtual memory slower than RAM?

(iii) باقاعدگی سے ڈسک کلین اپ چلانے کے دو فوائد تحریر کریں۔

(iii) List the two benefits of running regular disk cleanup.

(iv) پائتھن ڈویلپمنٹ انوائیرمنٹ سیٹ اپ کرنا کیوں اہم ہے؟

(iv) Why is it important to setup Python Development Environment?

(v) پائتھن میں ویری ایبلز کو نام دینے کے کوئی سے دو قوانین تحریر کریں۔

(v) Write down two variable naming rules in Python.

(vi) پائتھن میں ارتھمیٹک اور لاجیک آپریشنز میں فرق بیان کریں۔

(vi) Differentiate between arithmetic and logical operations in Python.

سوال 3: کوئی سے چار سوالات کے مختصر جوابات تحریر کریں:

Question # 3: Write Short answers to any Four (04) questions. (2x4=8)

(i) If۔ سٹیٹمنٹ، if-else۔ سٹیٹمنٹ سے کیسے مختلف ہے؟

(i) How does the if-statement differ from the if-else statement?

(ii) پائتھن لسٹس میں append آپریشن کا کیا مقصد ہے؟

(ii) What is the purpose of append operation in Python lists?

(iii) ڈیٹا سائنس کے حقیقی دُنیا میں کوئی سے دو اطلاق تحریر کریں۔

(iii) Write down two real world applications of Data Science.

(iv) ڈیٹا کو استعمال سے قبل ویلیڈیٹ کرنا کیوں ضروری ہے؟

(iv) Why is it important to validate data before using it?

(v) ڈیٹا اکھٹا کرنے کے لیے درست سورس کے انتخاب کی اہمیت بیان کریں۔

(v) Write the important to choose right source for data collection.

(vi) بذریعہ چارٹس ڈیٹا ویژولائزنگ کا کیا مقصد ہے؟

(vi) What is the purpose of visualizing data using charts?

سوال 4: کوئی سے چار سوالات کے مختصر جوابات تحریر کریں:

Question # 4: Write Short answers to any FOUR (4) questions. (2x4=8)

(i) سپروائزڈ لرننگ کس طرح کام کرتی ہے؟

(i) How does supervised learning work?

(ii) AI کو بہت زیادہ مقدار میں اچھی کوالٹی کا ڈیٹا کیوں درکار ہوتا ہے؟

(ii) Why does AI need large and good-quality data?

(iii) نیچرل لینگوئج پروسیسنگ (NCP) کا کردار بیان کریں۔

(iii) Write the role of Natural Language Processing (NLP)?

(iv) AI سسٹمز میں BIAS سے کیا مراد ہے؟

(iv) What is meant by bias in AI systems?

(v) بزنس میں اخلاقی فیصلہ سازی کی ایک مثال دیں۔

(v) Give one example of ethical decision-making in business.

(vi) SWOT تجزیہ کاری کس چیز کی شناخت میں مدد کرتی ہے؟

(vi) What does SWOT analysis help identify?

کوئی سے دو سوالات کے تفصیلی جوابات تحریر کریں:

Long Question Solve any Two Questions. (2x8=16)

سوال 5: آپریٹنگ سسٹم کے آرکیٹکچر کو تفصیل سے بیان کریں۔

Question # 5: Describe the architecture of an operating system.

سوال 6: 1 سے 20 کے درمیانی طاق اعداد بذریعہ لوپ پرنٹ کروانے کے لیے پائتھن پروگرام تحریر کریں۔

Question # 6: Write a Python program using a loop that prints all the odd numbers between 1 and 20.

سوال 7: مسائل کو حل کرنا اور اڈاپٹ ایبلٹی کے پراسس کی وضاحت کریں۔

Question # 7: Explain the process of problem solving and adaptability.