



तत् त्वं पूषन् अपावृणु
केन्द्रीय विद्यालय संगठन
KENDRIYA VIDYALAYA SANGATHAN

KENDRIYA VIDYALAYA SANGATHAN

A Step Ahead in Quality Education



CBSE
CENTRAL BOARD OF
SECONDARY EDUCATION

CLASS **XI**

COMPUTER SCIENCE

QUESTION BANK

SESSION 2026-27

As per CBSE Syllabus



Exam Oriented • Concept Based • Comprehensive Practice



Chapterwise
Questions



Objective &
Subjective
Questions



Competency
Based Questions



Programming
Practice



Higher Order
Thinking
Questions (HOTS)



Conceptual
Clarity



Aligned with
Latest CBSE
Syllabus



NCERT Based
& KVS Oriented



Builds Logical
Thinking &
Problem Solving



Perfect for
Formative &
Summative
Assessments

KVS AGRA REGION

KENDRIYA VIDYALAYA SANGATHAN
(AGRA REGION)



तत् त्वं पूषन् अपावृणु
केन्द्रीय विद्यालय संगठन

QUESTION BANK/VIDEO LECTURE LINKS COMPUTER SCIENCE

CLASS XI (SESSION-2026-27)

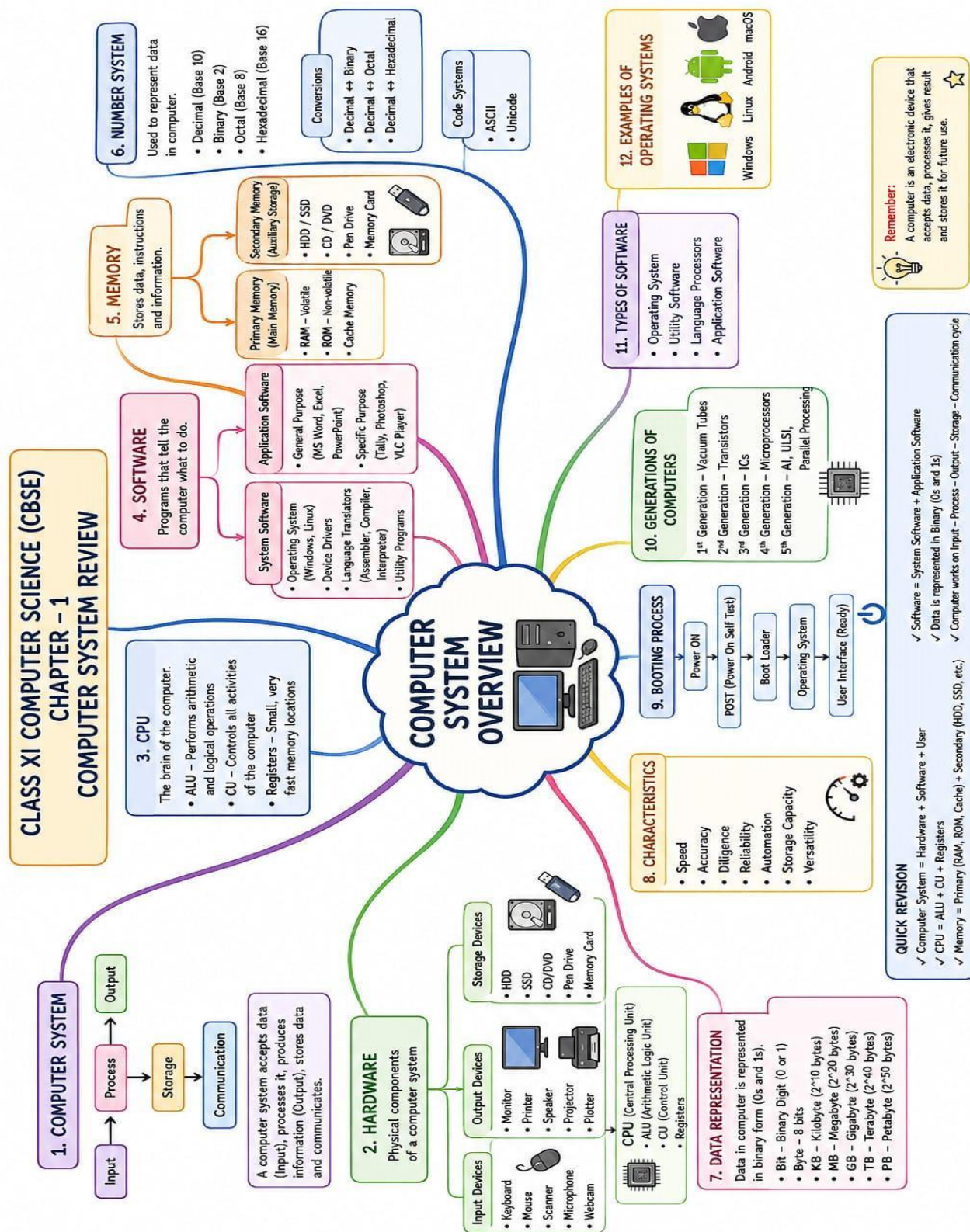
S. No.	Prepared by	Topics/Chapters Name	Compiled, Reviewed & Moderated By
1.	Mr. Deepak Tripathi, PGT(CS), PM SHRI KV Jhansi Cantt No.2	UNIT-1 : Computer System Organization (Computer System Overview, Data Representation, Boolean Logic)	1. Mr. Ajay Kumar, PGT(CS), PM SHRI KV SL Meerut 2. Mr. Sandeep Kumar, PGT(CS), PM SHRI KV DL Meerut
2.	Mr. Sumit Kumar Sahu, PGT (CS) PM SHRI KV Babina Cantt, Jhansi	UNIT-2 PART-1: Computational Thinking and Programming - Introduction to problem solving, Getting started with python, Flow of control - conditional statements introduction to python modules	
3.	Mr. Ramesh Kumar PGT (CS), PM SHRI KV Baoli Baghpat	UNIT-2 PART-2: Computational Thinking and Programming - Loops, String, List, Tuple & Dictionary in Python	
4.	Mr. Ravi Tewari, PGT(CS), PM SHRI KV Bulandshahar (II Shift)	UNIT -3 : Society, Law and Ethics	

VIDEO LECTURES (HD) FOR CLASS XI CS

<u>S.NO</u>	Prepared By	TOPIC	VIDEO LINK
1	Mrs Reetu Randev PM SHRI KV Moradabad	Computer Systems and Organisation	https://drive.google.com/drive/u/0/folders/1sV4KoHYH0ttTM1EQ8Q9frgoGGG98GK1K
2	Mr Abhoy Kumar Gupta PM SHRI KV Hathras	Computational Thinking and Programming -I	https://drive.google.com/drive/folders/10RBzUb7NCibWw_URMiAKenVy7Sixtc1l?usp=sharing
3	Sh. Sunit Kumar PM SHRI KV Muzaffarnagar	Computational Thinking and Programming-	https://drive.google.com/file/d/16ryyu5dLOQLwavyxnrz-fXebyVR-H61H/view?usp=drive link
4	Mr Sandeep Kumar PM SHRI KV DL Meerut Cantt	Society, Law and Ethics (Part-1): Digital footprints, Digital Society, Netiquettes	https://youtu.be/dH78ouOooxs
5		Society, Law and Ethics (Part-2): Data Protection, IPR, Licensing	https://youtu.be/bK-mZVhFYhk
6		Society, Law and Ethics (Part-3): Cyber Crimes - Phishing, Eavesdropping, hacking	https://youtu.be/UAyXngAI94M

CLASS XI Computer Science (083) : UNIT-1 Computer System Organization

Chapter-01 Computer System Overview



S. NO.	QUESTION
	Select Response type Questions (MCQ -20 %) - (06 Question)
1	Which of the following is considered the "brain" of the computer? a) RAM b) ROM c) CPU d) Hard Disk
2	What is the smallest unit of memory in a computer? a) Byte b) Bit c) Kilobyte d) Megabyte
3	Which type of memory is volatile and loses its data when the power is turned off ? a) ROM b) Hard Disk c) RAM d) SSD
4	Which of these is an example of an input device? a) Monitor b) Printer c) Keyboard d) Speaker
5	A _____ is a program that translates a high-level language program into machine language line by line. a) Assembler b) Compiler c) Interpreter d) Translator
6	A collection of programs that controls and coordinates the overall operations of a computer system is called: a) Application Software b) System Software c) Utility Software d) Programming Language
	Constructed response questions (SHORT ANSWER AND LONG ANSWER) -30 %-(09)
7	What is the role of drivers in computer hardware?
8	Differentiate between system software and application software with one example each.
9	Why is utility software important? Give an example.
10	Give any two differences between input and output devices with examples.
11	Discuss the role of system software. How does it differ from application software?
12	Explain the different types of number systems with examples.

13	Differentiate between compiler and interpreter with at least four points.
14	A programmer made an error in line 5 of a program. Which translator (compiler or interpreter) will detect it first and why?
15	State one advantage of using a compiler over an interpreter.
	Competency based Questions in the form of MCQs/Case Based Questions, Source-Based Integrated or any other Questions=50% (17 Q)
16	A graphic designing studio uses high-end PCs with 32 GB RAM, SSD storage, and dedicated graphic cards. They work with Adobe Photoshop, Corel DRAW, and Windows 10 Pro. On the basis of graphic designing studio information give the answers of following questions 1. Identify one input and one output hardware device they might use. 2. Mention one system software and one application software. 3. Why is SSD preferred over HDD in this case?
17	A school established a new computer lab with 20 desktop computers. Each computer has Intel i5 processor, 8 GB RAM, 1 TB HDD, and runs on Windows 11. For academic activities, they installed MS Office, Python IDE, and antivirus software. On the basis of computer lab information give the answers of following questions 1. Identify two hardware components and two software components from the case study. 2. Classify the antivirus software as system software or application software. 3. Which component stores the operating system?
	Q. 18 & 19 are ASSERTION AND REASONING based questions. Mark the correct choice as 1. Both A and R are true and R is the correct explanation for A 2. Both A and R are true and R is not the correct explanation for A c) A is True but R is False d) A is false but R is True
18	A: Cache memory speeds up processing by storing frequently accessed data. R: Cache is located between the CPU and the main memory.
19	A: Operating systems are considered application software.

	R: Application software directly interacts with hardware to perform basic tasks.
20	A company wants to upgrade its office computers to improve speed and performance. Which hardware components should they consider upgrading and why?
21	A student is facing slow system performance even with sufficient RAM. Which other hardware component may be responsible?
22	An organization is using antivirus software. Which category does it fall into and why?
23	Why is cache memory important in computer systems?
24	A company uses payroll software for salary processing. Identify the type of software and justify.
25	Give two examples each of system software and application software.
26	If the data bus width increases from 32-bit to 64-bit, what impact does it have?
27	Classify the following as system or application software: MS Excel, Windows OS, Disk Cleanup, Photoshop.
28	Why is the address bus unidirectional, whereas the data bus is bidirectional?
29	What is cache memory?
30	A student is working on a science project and he stores all his files in RAM while working. Suddenly, the power goes out, and all the work is lost. Questions: 1. Which type of memory was used here? 2. Why was the data lost? 3. Suggest a type of memory that would have retained the data even after power loss.
31	The school computer lab uses different software. Students write Python programs using an IDE, while the operating system manages memory, files, and hardware resources. Teachers prepare question papers using word processing software. i. Which software manages computer hardware and resources?

	a) MS Word b) Python IDE c) Operating System d) Paint ii. Which of the following is system software? a) Windows b) MS Excel c) Chrome Browser d) VLC Player iii. Python is an example of: a) Utility Software b) Programming Language c) Operating System d) Antivirus Software iv. Which software is used for creating documents? a) MS Word b) Python c) BIOS d) Linux Kernel v. Which software translates high-level language into machine language? a) Compiler/Interpreter b) Operating System c) Browser d) Utility Software
32	The computer teacher is preparing a list of devices required for the computer laboratory. Students need to enter data, listen to audio lessons, print certificates, and scan documents. i. Which of the following is an input device? a) Monitor b) Printer c) Scanner d) Speaker ii. Which device converts hard copy into digital form? a) Plotter b) Scanner c) Printer d) Projector iii. Which device is mainly used to produce printed documents? a) Speaker b) Printer c) Scanner d) Webcam iv. Which device is both an input and output device? a) Touch Screen b) Monitor c) Keyboard d) Mouse v. Which output device is commonly used to display soft copy? a) Keyboard b) Mouse c) Monitor d) Scanner

ANSWERS : Chapter-01 Computer System Overview

ANSWER Q(1-6)

Q1	Q2	Q3	Q4	Q5	Q6
C	B	C	C	C	B

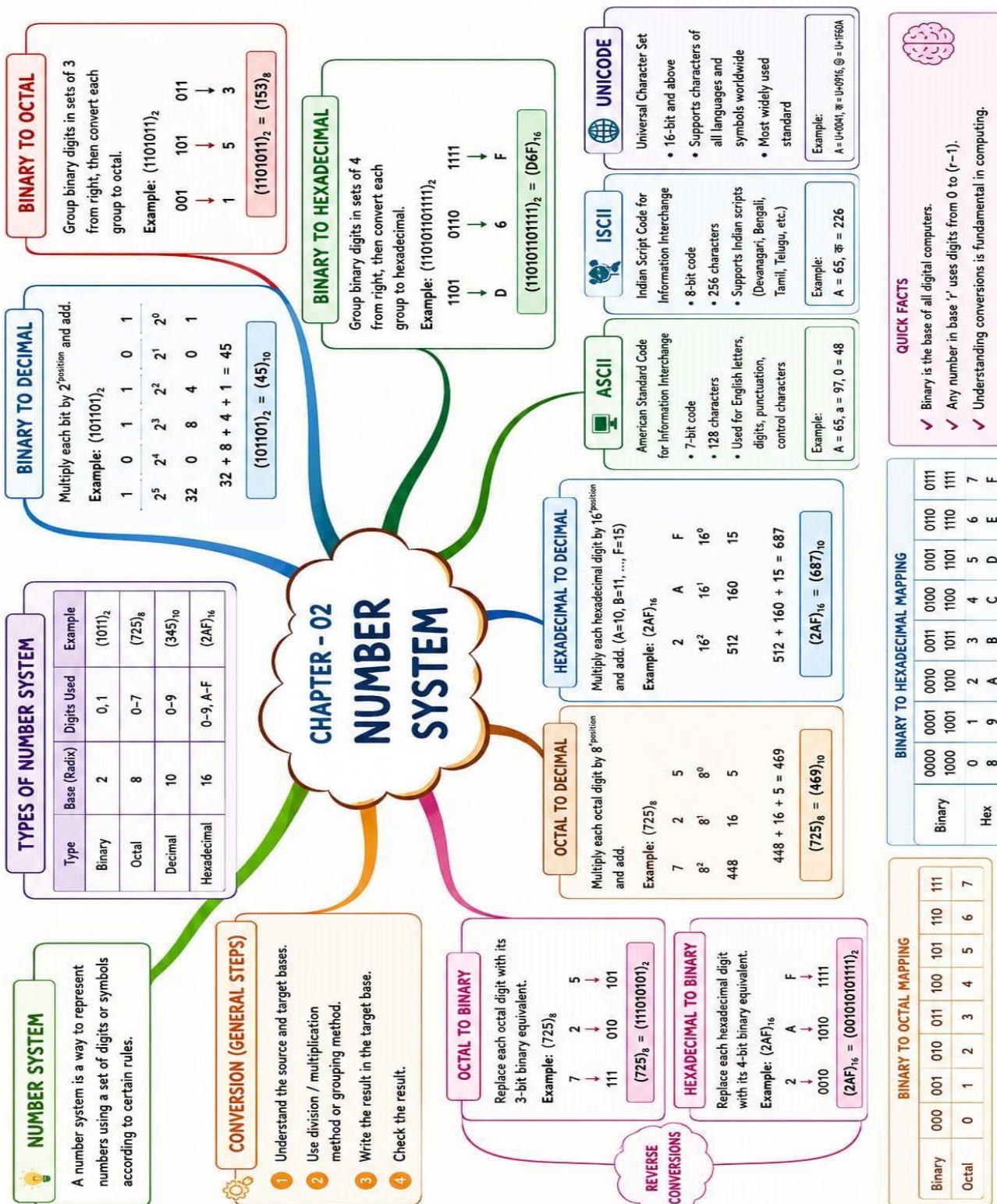
ANSWER Q(07-32)

7	Drivers are specialized programs that enable the operating system to communicate with hardware devices. Example: Printer driver helps OS communicate with a printer.
8	System Software: It manages and controls computer hardware and provides a platform to run application software. Example: Operating System (Windows, Linux). Application Software: It helps the user perform specific tasks. Example: MS Word, Photoshop.
9	Utility software helps in system maintenance and smooth operation by performing specific tasks. Example: Antivirus software
10	Input Devices: Used to enter data into a computer system. Example: Keyboard, Scanner. Output Devices: Used to display processed data. Example: Monitor, Printer.
11	Role of System Software: 1.Manages hardware components 2.Controls input/output devices 3. Allocates memory and processor time Difference with Application Software: 1.System software works in the background and runs the computer. 2. Application software is used by users to perform tasks like creating documents or editing photos.
12	1. Translation Method: Compiler: Translates whole code at once. Interpreter: Translates line by line.

	2. Execution Speed: Compiler: Faster after compilation. Interpreter: Slower because it translates during execution. 3. Error Detection: Compiler: Displays all errors after compilation. Interpreter: Shows errors immediately line by line. 4. Example: Compiler: C, C++ Interpreter: Python, JavaScript
14	Answer: Interpreter, because it executes and checks each line sequentially.
15	Answer: Compiler-generated code runs faster since the program is translated once and then executed multiple times.
16	1.Input device: Graphic tablet; Output device: High-resolution monitor. 2.System software: Windows 10 Pro; Application software: Adobe Photoshop. 3.SSD is preferred because it offers faster data access speed and better performance for graphic-heavy tasks.
17	1.Hardware: Processor, RAM, Hard Disk; Software: Windows 11, MS Office. 2. Antivirus is an application software. 3. The hard disk stores the operating system.
18	A and R are true, and R is the correct explanation of A
19	Both A and R are false.
20	They should consider upgrading RAM (for faster multitasking), SSD (for faster data access and boot time), and CPU (for higher processing speed).
21	Answer: The issue could be due to a slow hard disk drive (HDD) or overheating CPU, affecting data processing speed.

22	Answer: It is a utility software (a type of system software) as it helps maintain system health and security.
23	Answer: Cache memory stores frequently accessed data, reducing CPU waiting time and improving processing speed.
24	Answer: It is an application software because it is designed for a specific task—salary processing.
25	Answer: System Software: Operating System, Device Drivers Application Software: MS Word, Tally ERP
26	Answer: It increases the amount of data transferred per cycle, improving processing speed.
27	Answer: MS Excel: Application Software Windows OS: System Software Disk Cleanup: Utility Software (System) Photoshop: Application Software
28	Answer: Address bus only sends addresses (from CPU to memory), while data bus carries data both ways (read/write).
29	Answer: Cache memory is a high-speed memory located between the CPU and RAM to speed up data access.
30	Which type of memory was used here? Answer 1: RAM (Random Access Memory) Answer 2: RAM is volatile memory, meaning it loses data when power is turned off Answer: 3 Hard disk drive (HDD), Solid-State Drive (SSD), or ROM.
31	I.C II.A III.B IV.A V.A
32	I.C II. B III.B IV.A V.C

Chapter -02 Data Representation



SN	QUESTION
	Select Response type Questions (MCQ -20 %)-(06 Questions)
1	What is the binary equivalent of the decimal number 13? A) 1100 B) 1101 C) 1110 D) 1011
2	The octal number system uses digits from: A) 0-7 B) 0-8 C) 1-8 D) 1-7
3	Convert the octal number 17 to decimal. A) 14 B) 15 C) 16 D) 17
4	Which of the following is a valid hexadecimal digit? A) G B) 9 C) H D) 8G
5	ASCII is primarily used to represent: A) Numbers only B) Characters (letters, digits, symbols) C) Images D) Videos
6	How many bits are used in standard ASCII encoding? A) 7 bits B) 8 bits C) 16 bits D) 32 bits
	Constructed response questions (SHORT ANSWER AND LONG ANSWER) -30 %-(09 Q)
7	Convert the binary number $(110101)_2$ to octal.
8	What is the hexadecimal number system?
9	Convert binary $(11011011)_2$ to hexadecimal.
10	Convert the hexadecimal number 2F to decimal and binary.
11	What is the hexadecimal equivalent of the binary number 111100?
12	Explain the difference between ASCII and Unicode with examples.
13	Convert the binary number 1011 to decimal.
14	Write the steps to convert a decimal number into octal with an example.

15	Explain with an example how ASCII codes represent characters in binary.
	Competency based Questions in the form of MCQs/Case Based Questions, Source-Based Integrated or any other Questions=50% (17 Q)
16	An IP address of the system is 11000000.10101000.00000001.00000001. On the basis of IP address of the system, write the answer of the following questions. 1.Convert the first octet (11000000) of IP address into decimal. 2. Identify the number system used for generating IP addresses.
17	A 24-hour clock stores the time in BCD (Binary-Coded Decimal) format. If the clock shows 23:45. On the basis of BCD of the 24-hour clock . Write the answer of the following questions 1. Write BCD representation of 23. 2. Why is BCD used instead of pure binary for clocks?
	Q. 18 & 19 are ASSERTION AND REASONING based questions. Mark the correct choice as 3. Both A and R are true and R is the correct explanation for A 4. Both A and R are true and R is not the correct explanation for A c) A is True but R is False d) A is false but R is True
18	A: Converting a binary number to octal is faster when grouped in sets of four bits. R: Octal grouping is done in sets of three bits.
19	A: Binary number system uses only two digits, 0 and 1. R: Binary numbers are used because computers understand electrical signals represented by ON and OFF.
20	A: Hexadecimal numbers are easier to read and write compared to binary numbers. R: Each hexadecimal digit represents 4 binary digits (bits).
21	A: Binary 1010 is equivalent to decimal 12. R: Binary numbers are converted to decimal using powers of 2.
22	Why is octal number system sometimes used in computing?

23	Why an online form rejects your password containing emojis when stored in ASCII format ?
24	Which number system is used by the programmer to store color codes in #FF5733 format?
25	Your Wi-Fi router shows a MAC address: 3A:4F:6C:9B:22:D1 . Which number system is this? Why is it preferred?
26	A vehicle number plate scanning system converts characters into Unicode for processing. Why is Unicode better than ASCII for this?
27	What happens if an invalid digit (like 8 in octal) is used in a number?
28	How does grouping binary digits help in converting to octal or hexadecimal?
29	How does Unicode enable cross-platform communication?
30	How would you explain the difference between ASCII and Unicode to a non-technical person?
31	A computer science teacher asks students to understand different number systems. Rahul's roll number is 45 in the decimal number system. He is asked to convert it into binary and hexadecimal before storing it in a computer. i. Which number system is used internally by a computer? a) Decimal b) Binary c) Octal d) Hexadecimal ii. What is the base (radix) of the binary number system? a) 2 b) 8 c) 10 d) 16 iii. Which of the following is the binary equivalent of decimal 45? a) 101101 b) 110101 c) 101011 d) 111001 iv. What is the hexadecimal equivalent of decimal 45? a) 2B b) 2D c) 2F d) 3A
32	The school records the annual function and stores it as a digital video. During editing, the teacher also records narration using a microphone. Both audio and video are stored digitally. I. Audio is stored by converting sound into: a) Binary data b) Decimal numbers c) Roman numerals d) Text II. Which device is used to record digital audio?

a) Printer b) Microphone c) Plotter d) Scanner

III. Which file format is commonly used for audio?

a) MP3 b) JPG c) PNG d) PDF

IV. Which file format is commonly used for video?

a) MP4 b) TXT c) DOCX d) XLSX

Answers: Chapter -02 Data Representation

ANSWER Q(1-6)

Q1	Q2	Q3	Q4	Q5	Q6
B	A	B	B	B	A

7	Group binary digits in sets of 3 from right: 110 101 110 = 6, 101 = 5 Octal: (65) ₈
8	The hexadecimal number system is a base-16 system that uses digits 0–9 and letters A–F (A=10, B=11, ..., F=15). Each hex digit represents four binary bits.
9	Step 1 Separate the numbers into Group of 4 bits: 1101 1011 Step 2 Write the decimal equivalent of each Group 1101 = 13 , 1011 = 11
10	Decimal: $(2F)_{16} = (2 \times 16^1) + (F \times 1^0)$ $= 2 \times 16 + 15 \times 1$ $= 32 + 15 = 47$ Binary: $2 \rightarrow 0010 \quad F \rightarrow 1111$ So, $2F_{16} = 0010 1111_2$
11	Group binary digits in 4-bit groups: 1111 00 $\rightarrow 3C$ So, $(111100)_2 = (3C)_{16}$

12	Answer: <table border="1"> <tr> <td> ASCII (American Standard Code for Information Interchange) Uses 7 bits (128 characters). Supports English alphabets, digits, and basic symbols. Example: 'A' = 65 in ASCII </td><td> Unicode supports characters for multiple languages and symbols. Uses different formats like UTF-8, UTF-16, UTF-32. Example: 'A' = U+0041, 'अ' (Hindi) = U+0905. </td></tr> </table>	ASCII (American Standard Code for Information Interchange) Uses 7 bits (128 characters). Supports English alphabets, digits, and basic symbols. Example: 'A' = 65 in ASCII	Unicode supports characters for multiple languages and symbols. Uses different formats like UTF-8, UTF-16, UTF-32. Example: 'A' = U+0041, 'अ' (Hindi) = U+0905.
ASCII (American Standard Code for Information Interchange) Uses 7 bits (128 characters). Supports English alphabets, digits, and basic symbols. Example: 'A' = 65 in ASCII	Unicode supports characters for multiple languages and symbols. Uses different formats like UTF-8, UTF-16, UTF-32. Example: 'A' = U+0041, 'अ' (Hindi) = U+0905.		
13	Answer: $1011_2 = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = 8 + 0 + 2 + 1 = 11_{10}$		
14	Steps: • Divide the decimal number by 8. • Write down the remainder. • Repeat the process with the quotient until it becomes 0. • Read the remainders in reverse order. Example: Convert 58 to octal $58 \div 8 = 7$ remainder 2 $7 \div 8 = 0$ remainder 7 Octal = 72₈		
15	Answer: Each ASCII character is assigned a unique number, which is then represented in binary. Example: • Character: 'C' • ASCII Decimal Code: 67 • Binary Code: $67_{10} = 01000011_2$ Thus, 'C' is represented as 01000011 in ASCII binary form.		
	Competency based Questions in the form of MCQs/Case Based Questions, Source-Based Integrated or any other Questions=50% (15 Q)		
16	1. $1 \times 2^7 + 1 \times 2^6 + 0 + 0 + 0 + 0 + 0 + 0 = 128 + 64 = 192$ 2. Binary number system		

17	1. $23 \rightarrow 2 = 0010, 3 = 0011 \rightarrow \text{BCD} = 0010\ 0011$. 2. BCD is used because it represents decimal digits directly in binary, making it easier for digital displays.
18	A is false, but R is true.
19	A) Both A and R are true, and R is the correct explanation of A.
20	Answer: A) Both A and R are true, and R is the correct explanation of A.
21	Answer: D) A is false, but R is true. ($1010_2 = 10_{10}$)
22	Answer: Because it provides a shorter representation of binary (3 bits = 1 octal digit), making it easier to read and write.
23	Answer: ASCII cannot represent emojis; they require Unicode.
24	Answer: Hexadecimal
25	Answer: Hexadecimal, It represents long binary codes in a short and readable format.
26	Answer: Unicode supports regional scripts (like Hindi, Tamil), while ASCII cannot.
27	Answer: It becomes an invalid representation because the octal system only allows digits 0–7. The system will raise an error during conversion or computation.
28	Answer: Binary digits are grouped in 3 bits for octal and 4 bits for hexadecimal because these bases are powers of 2 ($8 = 2^3$, $16 = 2^4$). This makes conversion fast and error-free.
29	Answer: Unicode assigns a unique code point to every character , ensuring it displays correctly across devices and systems.
30	Answer: ASCII is like a small alphabet book (128 symbols) for English, while Unicode is a global dictionary that supports thousands of languages, symbols, and emojis.
31	I. B II. A III. A IV. B
32	I. A II. B III. A IV. A

Chapter-03 Boolean Logic

LOGIC GATES TYPES

1 AND Gate

$Y = A \cdot B$

A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

2 OR Gate

$Y = A + B$

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

3 NOT Gate

$Y = \overline{A}$

A	Y
0	1
1	0

4 NAND Gate

$Y = \overline{(A \cdot B)}$

A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

5 NOR Gate

$Y = \overline{(A + B)}$

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

6 XOR Gate

$Y = A \oplus B$

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

7 XNOR Gate

$Y = \overline{(A \oplus B)}$

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	1

LOGIC GATES



- Logic gates are the basic building blocks of digital circuits.
- They perform logical operations on one or more binary inputs and produce a single binary output.
- They operate on two values: 0 (Low/False) and 1 (High/True).

CHAPTER - 03

BOOLEAN ALGEBRA

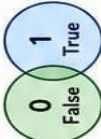
The foundation of Digital Logic

BOOLEAN ALGEBRA

Boolean Algebra is a branch of algebra that deals with binary variables and logical operations.

It uses two values: 0 (False) and 1 (True).

Variables are usually represented by A, B, C ... etc.



ITS NEED



- It is the basis of all digital electronics and computer systems.
- It helps in designing and simplifying digital circuits.
- It provides a systematic way to analyze and minimize logic expressions.
- Essential for the design of logic gates, computers, calculators, and all digital devices.

TRUTH TABLES



A truth table shows all possible input combinations and the corresponding output.

A	B	A+B	$\bar{A}$	$A \oplus B$
0	0	0	1	0
0	1	1	1	1
1	0	1	0	1
1	1	1	0	0

Truth tables help in analyzing and verifying logical expressions and circuits.

IMPORTANT POINTS

- Binary values: 0 = False (Low), 1 = True (High)
- Basic operations: AND ($\cdot$), OR ($+$), NOT ($\bar{}$)
- Used in digital circuit design and simplification
- Forms the basis of all digital systems
- Follows laws and theorems for simplification

TAUTOLOGY



A Boolean expression that is TRUE (1) for all possible input combinations is called a Tautology. It is always TRUE.

Example:

$$A + \bar{A} = 1 \text{ (Tautology)}$$

Truth Table:

A	$\bar{A}$	$A + \bar{A}$
0	1	1
1	0	1

Tautology is used to verify logic identities and simplifications.

FALSECASSY (CONTRADICTION)



A Boolean expression that is FALSE (0) for all possible input combinations is called a Falsecassy (Contradiction). It is always FALSE.

Example:

$$A \cdot \bar{A} = 0 \text{ (Falsecassy)}$$

Truth Table:

A	$\bar{A}$	$A \cdot \bar{A}$
0	1	0
1	0	0

Falsecassy is used to detect impossible or invalid conditions in logic circuits.

SN	QUESTION
	Select Response type Questions (MCQ -20 %)-(06)
1	What will be the output of an OR gate if both inputs are 1? A) 1 B) 0 C) True D) 2
2	Which gate gives output 0 only when all inputs are 0? A) AND B) OR C) NOT D) XOR
3	Write the Boolean expression for an AND gate with three inputs A, B, and C. A) $A+B+C$ B) $A+B.C$ C) $A.B.C$ D) $A.B+C$
4	Which of the following is the symbol for OR operation? A) $\times$ B) $+$ C) $-$ D) $/$
5	Which gate can have two or more inputs but only one output? A) AND B) OR C) Both AND & OR D) NOT only
6	What is the main difference between AND and OR gate output? A) AND gives 1 for all inputs 0 B) OR gives 0 when any input is 1 C) AND requires all inputs to be 1 for output 1, OR requires only one D) Both are same
	Constructed response questions (SHORT ANSWER AND LONG ANSWER)
7	What is the simplified form of the Boolean expression $Y = AB + AB'$?
8	What does the Boolean expression $Y=A+B$ represent in terms of logic gates?
9	If a truth table output is always 1 for all input combinations, what does it represent?
10	Explain the importance of truth tables in designing digital circuits.
11	Construct the truth table for the Boolean expression $Y=(A+B)(A'+B')$ and simplify it.
12	Explain AND, OR, and NOT gates with their symbols and truth tables.

13	Write the Boolean expression for an OR gate with three inputs A, B, and C.
14	Construct the truth table for the Boolean expression $Y=(A'+B')$.
15	Explain XOR and XNOR gates with their applications.
	Competency based Questions in the form of MCQs/Case Based Questions, Source-Based Integrated or any other Questions=50% (17 Q)
16	Draw the logic gate diagram for the Boolean expression $Y = A \cdot B + C$.
17	How many rows will a truth table have for 4 variables (A, B, C, D)?
	Q. 18 & 19 are ASSERTION AND REASONING based questions. Mark the correct choice as a)Both A and R are true and R is the correct explanation for A b)Both A and R are true and R is not the correct explanation for A c) A is True but R is False d) A is false but R is True
18	Assertion (A):The Boolean expression for a two-input NAND gate is $Y = \neg(A \cdot B)$. Reason (R): NAND is a combination of AND followed by NOT.
19	Assertion (A): NOR gate is considered a universal gate. Reason (R): Because using only NOR gates, any Boolean function can be implemented.
20	How do you represent the NOT operation in Boolean algebra and its truth table?
21	Smart home lighting system works under the following conditions: A: Motion detected B: Manual switch ON The light should glow if motion is detected OR the manual switch is ON. a) Write the Boolean expression lighting system. b) Draw the logic gate diagram. c) Determine the output when $A=0$ and $B=1$.
22	Security light turns ON when either motion is detected ($M=1$) or it is dark ($D=1$), but it should remain OFF when both are false. Write the Boolean expression. What is its truth table size if

	there are only two variables?
23	For which values of A and B does $A \oplus B = 1$?
24	Prove that $A(A+B) = A$.
25	Simplify: $(A+B)(A'+B)$ using truth table
26	Write the Boolean expression for the following :- A system grants access when the password is correct ($P=1$) and the user is authorized ($A=1$).
27	Write the Boolean expression for the following :- A smart fan works only when the temperature is high ($T=1$) and the room is occupied ($R=1$).
28	In a voting system, the result light turns ON if either of the two officers approves. Which logic gate is required?
29	Create a truth table for the expression $Y = A \cdot B + C'$.
30	Represent the Boolean equation $Y = A' + B$ using logic gates.
31	A student can access the online examination portal if: a. Username is correct ($U = 1$) b. Password is correct ($P = 1$) If either one is incorrect, login fails. Questions: - Which Boolean operator is used? - Write the Boolean expression. - What will be the output when: (a) $U = 1, P = 0$ (b) $U = 1, P = 1$
32	A student can enter the computer laboratory only if: A. Identity Card (I) is available. B. Lab Permission (L) is granted. A student without an identity card cannot enter even if permission is granted. Questions: - Which logic gate represents this situation? - Write the Boolean expression. - State the output when: (a) $I = 0, L = 1$ (b) $I = 1, L = 1$

ANSWERS: Chapter-03 Boolean Logic

ANSWER Q(1-6)

Q1	Q2	Q3	Q4	Q5	Q6
A	B	C	B	C	C

7	Answer: Factor A: $Y = A(B + B') = A(1) = A$																									
8	Answer: It represents an OR gate , where output is 1 if either A or B is 1.																									
9	Answer: It represents a tautology .																									
10	Answer: Truth tables help designers visualize how input combinations affect output, verify Boolean expressions, detect errors, and simplify logic circuits for efficient design.																									
11	<table><tr><th>A</th><th>B</th><th>A+B</th><th>A'+B'</th><th>Y</th></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td></tr></table>	A	B	A+B	A'+B'	Y	0	0	0	1	0	0	1	1	1	1	1	0	1	1	1	1	1	1	0	0
A	B	A+B	A'+B'	Y																						
0	0	0	1	0																						
0	1	1	1	1																						
1	0	1	1	1																						
1	1	1	0	0																						
12	Answer: AND Gate: Outputs 1 when all inputs are 1. Symbol: D-shaped with two or more inputs. AND GATE TRUTH TABLE <table><tr><th>A</th><th>B</th><th>A.B</th></tr><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td></tr></table> NOT GATE TRUTH TABLE <table><tr><th>A (Input)</th><th>Y = $\bar{A}$ (Output)</th></tr><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table>	A	B	A.B	1	1	1	1	0	0	0	1	0	0	0	0	A (Input)	Y = $\bar{A}$ (Output)	0	1	1	0				
A	B	A.B																								
1	1	1																								
1	0	0																								
0	1	0																								
0	0	0																								
A (Input)	Y = $\bar{A}$ (Output)																									
0	1																									
1	0																									

12 OR GATE TRUTH TABLE

A	B	A+B
1	1	1
1	0	1
0	1	1
0	0	0

13 **Answer:** $Y=A+B+C$

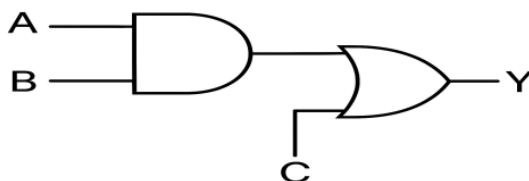
A	B	A'	B'	A'+B'
1	1	0	0	0
1	0	0	1	1
0	1	1	0	1
0	0	1	1	1

15 **Answer:****XOR Gate:** Outputs 1 when inputs are different.Expression: $Y=A \oplus B = A'B + AB'$

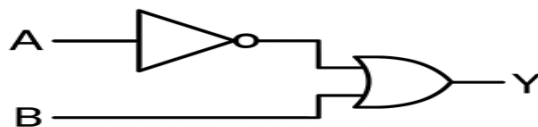
Application: Error detection in digital communication.

XNOR Gate: Outputs 1 when inputs are same.Expression: $Y=A \odot B = AB + A'B'$ 16 **Steps:**1. Take inputs A and B $\rightarrow$ AND gate $\rightarrow$ Output1 = $A \cdot B$ 2. Take Output1 and C $\rightarrow$ OR gate $\rightarrow$ Final Output = $A \cdot B + C$

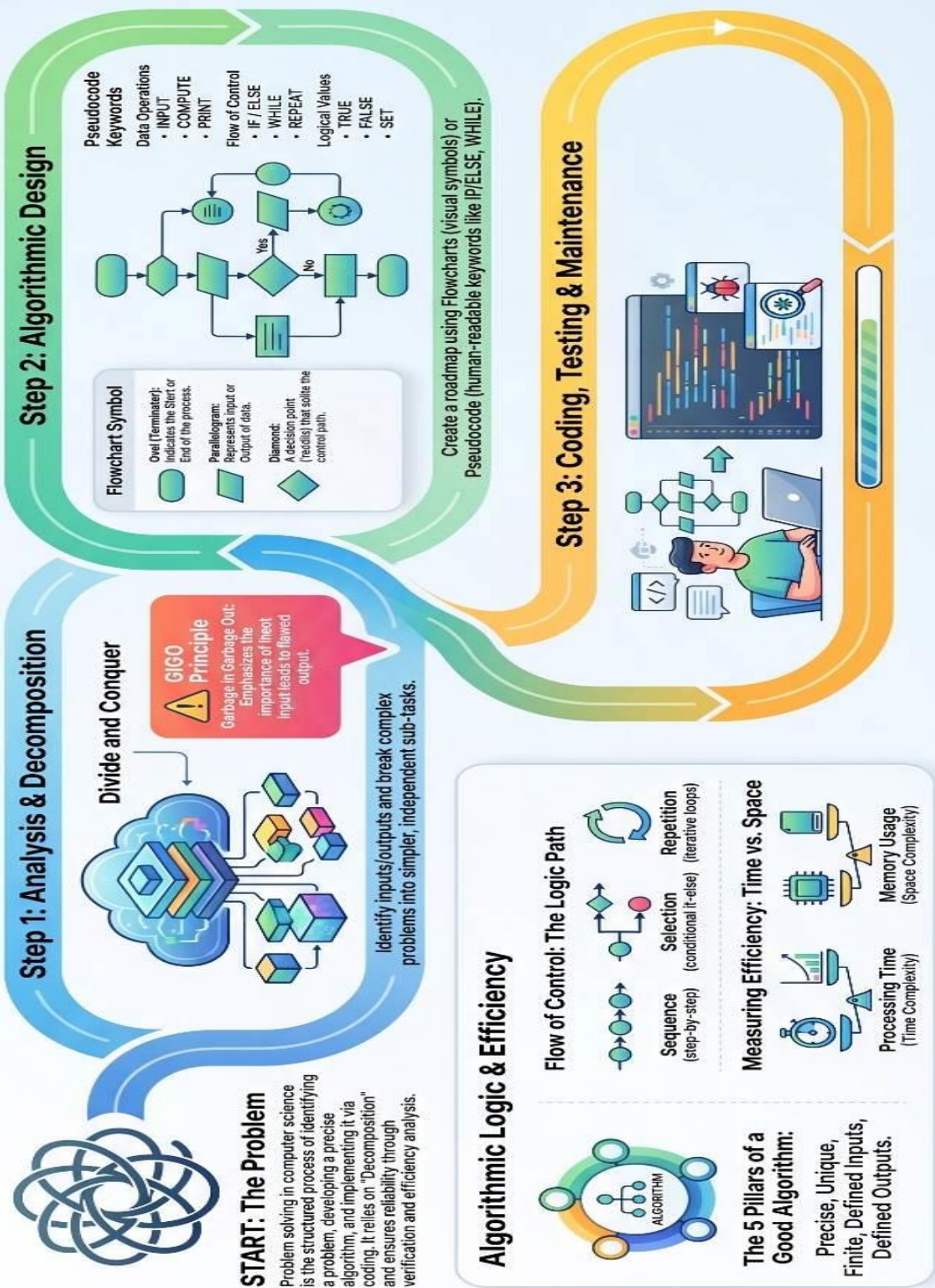
$$Y = A \cdot B + C$$



17	Answer: $2^4 = 16$ rows , because each variable has two possible values (0 and 1).																									
18	Answer: Both A and R are true, and R is the correct explanation of A.																									
19	Answer: Both A and R are true, and R is correct explanation of A .																									
20	Answer: NOT is represented as A' . Truth table: <table><tr><td>A</td><td>A'</td></tr><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table>	A	A'	0	1	1	0																			
A	A'																									
0	1																									
1	0																									
21	a)Boolean expression: $Y = A + B$ b)Logic gates used: OR gate c) Output for $A=0, B=1 \rightarrow Y = 0 + 1 = 1$ (Light ON)																									
22	a)Answer: $Y = M + D$ b)Answer: $2^2 = 4$ rows.																									
23	Answer: XOR outputs 1 when inputs are different . $A=0, B=1 \rightarrow 1$ $A=1, B=0 \rightarrow 1$																									
24	Answer: Expand: $A(A+B)=AA+AB=A+AB=A$ (by Absorption Law)																									
25	Simplify: $(A+B)(A'+B)$ <table><tr><td>A</td><td>B</td><td>$A+B$</td><td>$A'+B$</td><td>$(A+B).(A'+B)$</td></tr><tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td><td>1</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td></tr></table>	A	B	$A+B$	$A'+B$	$(A+B).(A'+B)$	1	1	1	1	1	1	0	1	1	1	0	1	1	1	1	0	0	0	1	0
A	B	$A+B$	$A'+B$	$(A+B).(A'+B)$																						
1	1	1	1	1																						
1	0	1	1	1																						
0	1	1	1	1																						
0	0	0	1	0																						
26	P AND A $\rightarrow$ AND Gate																									
27	Expression: T AND R																									

28	Answer: OR gate																																																						
29	<table><tr><th>A</th><th>B</th><th>C</th><th>$\overline{C}$</th><th>A·B</th><th>Y = A·B + $\overline{C}$</th></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>1</td></tr></table>	A	B	C	$\overline{C}$	A·B	Y = A·B + $\overline{C}$	0	0	0	1	0	1	0	0	1	0	0	0	0	1	0	1	0	1	0	1	1	0	0	0	1	0	0	1	0	1	1	0	1	0	0	0	1	1	0	1	1	1	1	1	1	0	1	1
A	B	C	$\overline{C}$	A·B	Y = A·B + $\overline{C}$																																																		
0	0	0	1	0	1																																																		
0	0	1	0	0	0																																																		
0	1	0	1	0	1																																																		
0	1	1	0	0	0																																																		
1	0	0	1	0	1																																																		
1	0	1	0	0	0																																																		
1	1	0	1	1	1																																																		
1	1	1	0	1	1																																																		
30	A B  $Y = \overline{A} + B$																																																						
31	1. AND Gate 2. Login = U · P 3. (a) 0 (Login Failed) (b) 1 (Login Successful)																																																						
32	1. AND Gate 2. Access = I · L 3. (a) 0 (Access Denied) (b) 1 (Access Granted)																																																						

Mastering Problem Solving: From Logic to Code



12	What role does "Testing" play in the program development life cycle?
13	Draw or list the standard geometric shapes used in flowcharts along with their specific functions.
14	Why is documentation considered a crucial final step in problem solving?
15	An algorithm must be "unambiguous". What does this mean?
Assertion & Reason Questions (5 Questions)	
<i>Directions: In the following questions, a statement of Assertion (A) is followed by a statement of Reason (R). Choose the correct option:</i> A) Both A and R are true and R is the correct explanation of A. B) Both A and R are true but R is NOT the correct explanation of A. C) A is true but R is false. D) A is false but R is true.	
16	Assertion (A): Flowcharts make it easier to understand and communicate the logic of a program compared to dense code files. Reason (R): Flowcharts use standardized visual symbols that map out the flow of control step-by-step.
17	Assertion (A): If a program runs completely without throwing any error messages, you can guarantee that the results are 100% accurate. Reason (R): Logical errors cannot be detected by computer compilers or interpreters.
18	Assertion (A): Pseudocode cannot be directly executed on a computer system. Reason (R): Pseudocode is written in plain language structures independent of any specific programming language syntax.
19	Assertion (A): An algorithm designed to find the average of five numbers can also be adapted to find the average of any N numbers. Reason (R): "Generality" is an essential characteristic that ensures an algorithm can handle variable data parameters within its scope.
20	Assertion (A): While solving a complex computational problem, coding should be started immediately without spending time on design algorithms. Reason (R): Skipping the algorithm phase often results in structured, bug-free software code later on.

	Case-Based Integrated Questions (5 Questions)
	Case Study: <i>An e-commerce platform wants to design an automated module to calculate delivery discounts. The criteria are as follows: If a customer's cart value is greater than ₹2000, they receive a 10% discount. If it is greater than ₹5000, they receive a 20% discount. For anything less than or equal to ₹2000, no discount is applied. A junior developer creates the following algorithm sequence:</i> 1. <i>Input CartValue</i> 2. <i>If CartValue > 2000 then Discount = 10%</i> 3. <i>If CartValue > 5000 then Discount = 20%</i> 4. <i>Else Discount = 0%</i> 5. <i>Print Final Price</i>
21	What fatal logic flaw will happen if a user checks out with a CartValue of ₹6000? A) The program will crash with a runtime divide-by-zero error. B) The program will evaluate step 2, assign 10%, evaluate step 3, assign 20%, but step 4's Else might overwrite it or cause execution confusion depending on how the block drops. C) The user will get a 0% discount because ₹6000 automatically triggers the final Else block attached only to step 3. D) The sequence lacks efficiency but will perfectly assign a 20% discount without issues.
22	How can this algorithm logic be accurately repaired using standard control structures? A) By changing all If statements into isolated, completely independent processes. B) By processing the highest value condition (CartValue > 5000) first, followed by an Else If condition for CartValue > 2000. C) By removing the Else condition completely from the module. D) By shifting input parameters to step 4.
23	Which flowchart symbol must be used to represent the conditional checks for CartValue? A) Rectangle B) Parallelogram C) Diamond D) Oval
24	If we want to test whether this newly corrected module works efficiently for edge cases, which set of values represents the best testing inputs? A) ₹1000, ₹2000, ₹5000, ₹6000 B) Only ₹0 C) Negative numbers only D) Text characters like "Two Thousand"
25	Suppose the developer writes the final program script in Python but accidentally typos

	the condition syntax as if CartValue = 5000:. What type of error will occur? A) Logical Error B) Runtime Error C) Syntax Error D) Execution Error
	Source-Based / Competency MCQs (5 Questions)
	Source Text: <i>"An algorithm can be represented in multiple ways. While a flowchart provides high readability for visual thinkers, it becomes highly chaotic and difficult to draw as the problem size scales up. Pseudocode bridges this gap by offering a scalable, compact layout that mimics code structure without binding the creator to complex strict punctuation rules like semicolons or strict indentation limits."</i>
26	Based on the source text, why does a flowchart become less effective as the complexity of a software problem increases? A) Computers cannot convert geometric symbols to binary. B) Visual flowcharts scale chaotically and become physically difficult to trace or draw with massive datasets or steps. C) Flowcharts do not support the use of variable declarations. D) Flowcharts are entirely limited to basic mathematical problems.
27	According to the source, what is the primary advantage that pseudocode holds over direct coding? A) Pseudocode runs faster on CPU architectures. B) It allows the creator to focus entirely on structural logic without getting stuck on strict punctuation rules like semicolons. C) It automatically designs the user interface layout. D) It can be automatically understood by any web browser.
28	Look at the algorithm structure below: Step 1: Set Count = 1 Step 2: Print Count Step 3: Set Count = Count + 1 Step 4: Go to Step 2 Which crucial property of an algorithm does this sequence violate? A) Unambiguous B) Input parameters C) Finiteness D) Generality
29	A student needs to write a program that calculates the simple interest for 500 different

	bank customers. Instead of writing the calculation code 500 times, which programming concept should be implemented in the algorithm? A) Infinite Sequential Structure B) De-composition Analysis C) Looping / Iteration Structure D) Selection Structure
30	Before executing an algorithm on real data, tracing the logic step-by-step on paper using sample data values to verify accuracy is called: A) White box compiling B) Dry Running C) Syntax Checking D) Beta Deployment

ANSWERS: CHAPTER-4: INTRODUCTION TO PROBLEM SOLVING

1	C	2	D	3	B
4	B	5	C	6	D

7	An algorithm is a step-by-step text-based procedure written in plain language to solve a specific problem. A flowchart is a visual or graphical representation of that same algorithm using standard geometric shapes (like ovals, rectangles, and diamonds) connected by flowlines.
8	Problem decomposition involves breaking down a large, complex, and unmanageable problem into smaller, simpler sub-problems. It makes the problem easier to understand, solve, and track, allowing a programmer to tackle one manageable component at a time.
9	<pre> INPUT number1, number2 IF number1 > number2 THEN PRINT "number1 is the largest" ELSE IF number2 > number1 THEN PRINT "number2 is the largest" ELSE PRINT "Both numbers are equal" ENDIF </pre>

10	Syntax errors occur when code violates the grammatical or structural rules of a programming language. The compiler or interpreter cannot understand the command. For example, writing <code>print("Hello"</code> (forgetting the closing parenthesis) in Python will trigger a syntax error.								
11	A runtime error causes a program to crash abruptly while it is executing (e.g., trying to divide by zero or opening a file that doesn't exist). A logical error does not crash the program; it runs completely, but the output is incorrect because the developer used the wrong formula or logic (e.g., writing <code>Area = length + breadth</code> instead of <code>length * breadth</code>).								
12	Testing involves executing the program with various sets of sample inputs (including normal, boundary, and invalid data) to verify that it functions correctly under all conditions and produces the expected outputs before it is deployed for actual use.								
13	Oval (Terminal): Indicates the Start or Stop points of the process. Parallelogram (Input/Output): Represents data inputs or displaying outputs. Rectangle (Process): Used for calculations, variable assignments, and internal processing. Diamond (Decision): Used for conditions or questions that result in a Yes/No or True/False branch. Arrows (Flowlines): Connect symbols to show the exact sequence of execution.								
14	Documentation provides a detailed written guide about how the program works, its logic, algorithms, and limitations. It is vital for future maintenance, debugging, and upgrades, especially if a different programmer works on the software later.								
15	Unambiguous means that every single step in the algorithm must be completely clear, precise, and have only one possible interpretation. There should be absolutely no confusion or guesswork for the person or machine executing the instruction.								
16	A	17	D	18	A	19	A	20	C
21	C	22	C	23	C	24	A	25	C
26	B	27	B	28	C	29	C	30	B

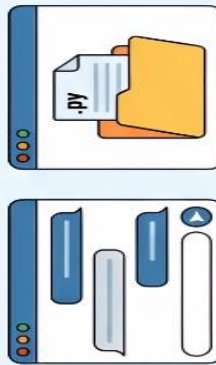
Mastering the Basics: A Guide to Getting Started with Python

Python is a high-level, interpreted language with user-friendly features, multiple execution modes, and essential data types for building functional programs.

PYTHON'S CORE FEATURES & EXECUTION

Interpreted Line-by-Line Execution

An interpreter translates and executes instructions one by one until completion or an error.



Interactive Mode

Use Interactive mode for instant results.

Script Mode

Use Script mode to save programs as .py files.



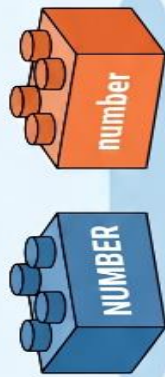
A User-Friendly, High-Level Language

Python is free, open-source, and features a simple syntax that is easy to understand.

THE BUILDING BLOCKS OF PYTHON CODE

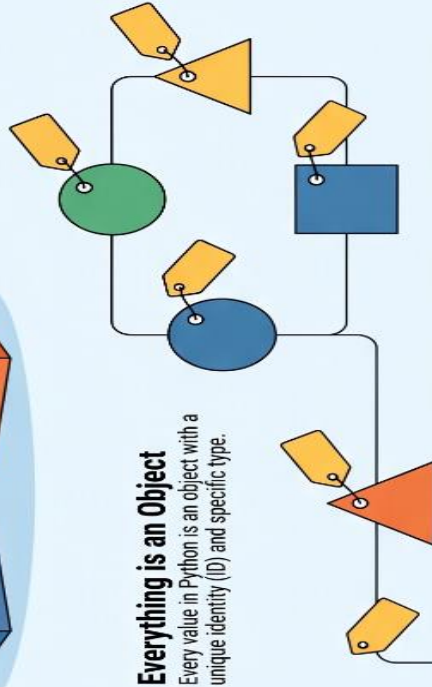
Case-Sensitive Identifiers

Variable names must be unique; for example, "NUMBER" and "number" are treated as different entities.

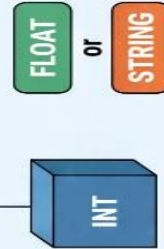


Everything is an Object

Every value in Python is an object with a unique identity (ID) and specific type.



DATA CATEGORY	EXAMPLES	MUTABILITY
Numbers	int, float, complex	Immutable
Sequences	strings, lists, tuples	Lists (Mutable) Tuples (Immutable)
Mappings	Dictionaries	Mutable



Dynamic Type Conversion

Python supports both automatic (implicit) and programmer-forced (explicit) data type changes.

SN	QUESTION
	Select Response type Questions (MCQ -20 %)-(06)
1	Python was developed by Guido van Rossum and was officially released in which year? A) 1989 B) 1991 C) 1995 D) 2000
2	Which of the following is considered a core limitation or disadvantage of using Python? A) It is not an open-source programming language. B) It has a steep learning curve and highly complex syntax. C) It is typically slower in execution speed compared to compiled languages like C/C++. D) It does not support Object-Oriented Programming (OOP) concepts.
3	In Python, which mode of execution is ideal for running single-line expressions and getting instant feedback, but does not save the code for future use? A) Script Mode B) Compile Mode C) Interactive Mode D) Source Mode
4	Which of the following is an invalid identifier in Python? A) _total_sum B) grand_total C) 2nd_row D) itemPrice
5	What is the fundamental difference between a Keyword and an Identifier in Python? A) Identifiers are case-sensitive, but keywords are not. B) Keywords are reserved words with fixed meanings for the compiler/interpreter, while identifiers are user-defined names. C) Keywords can change their behavior at runtime, whereas identifiers cannot. D) Keywords are used for writing text data, while identifiers are strictly for numbers.
6	Which of the following symbols is used to write a single-line comment in Python? A) // B) /* C) # D) \$
	Constructed response questions (SHORT ANSWER AND LONG ANSWER)
7	Python is called an interpreted language. Briefly explain what happens internally to your Python source code when you press the run button.
8	Differentiate between Interactive Mode and Script Mode in Python. Provide one scenario where you would prefer using script mode.
9	A student typed the following statement in Python to assign a value: 100 = score. What type of error will this statement generate, and why? Write the corrected statement.
10	What are Tokens in Python? Name all the 5 major types of tokens available in Python.
11	Identify the errors in the following code snippet and write the corrected version: name = input("Enter your name: ")

	<pre>print("Welcome" name) Age = input("Enter age: ") print(Age + 5)</pre>
12	Explain the purpose of the sep and end parameters in the Python print() function. Give a brief example of how they modify the default behavior.
13	Why is Python called a case-sensitive language? Show an example using variable declaration to support your answer.
14	What are Literals? Mention any three distinct types of literals supported by Python.
15	Write a small script that takes the user's name and their birth year as input, calculates their approximate age, and displays a customized message using a single print() statement.
	Case Study: (Questions 16-20)
	Rahul is setting up a python program to calculate the Simple Interest for a school bank project. He writes down a small script to capture values and perform calculations, but he runs into issues with how Python handles inputs. Below is the draft of his script: <pre># Bank Interest Calculator Draft principal = input("Enter Principal Amount: ") rate = input("Enter Rate of Interest: ") time = input("Enter Time in Years: ") # Calculation Step interest = (principal * rate * time) / 100 print("Your calculated interest is: ", interest)</pre>
16	When Rahul runs this program and inputs 1000, 5, and 2, the program crashes at line 7 with a TypeError. What is the primary cause of this error? A) The formula for simple interest is incorrect. B) The variables contain string data types, which cannot be multiplied together. C) The print statement on line 8 is missing a comma. D) Python does not support comments on line 1.
17	How should Rahul modify line 2 to ensure the principal variable can store decimal inputs correctly? A) principal = int(input("Enter Principal Amount: ")) B) principal = str(input("Enter Principal Amount: "))

	C) principal = float(input("Enter Principal Amount: ")) D) principal = input(float("Enter Principal Amount: "))
18	If Rahul fixes the code by explicitly converting all inputs to numbers, what type of programming concept is he utilizing? A) Type Casting / Explicit Type Conversion B) Dynamic Typing C) Tokenization D) Compilation
19	Suppose Rahul updates his script to dynamically infer values, and changes principal = 5000. Later in the same script, he writes principal = "Five Thousand". Which statement is true about Python regarding this behavior? A) It causes a SyntaxError because a variable's type cannot be changed once declared. B) Python supports dynamic typing, allowing a variable to point to different data types at different times. C) The variable principal will hold both values at the same time. D) It triggers a runtime memory overflow error.
20	If Rahul wants to display the output format exactly as Interest: \$100.0 without any space between the dollar sign and the number using his current print statement, which parameter will help him achieve this? A) end="" B) sep="" C) flush=True D) format()
	Source Snippet: (Questions 21-25)
	Examine the following interactive shell session log recorded by a teacher while demonstrating Python tokens and expressions to Class 11 students: <pre>>>> x = 15 >>> y = 4 >>> z = x // y >>> msg = 'Python\'s World' >>> print(type(msg))</pre>
21	Based on the source session, what will be the exact value stored in variable z? A) 3.75 B) 3 C) 4 D) 3.0
22	In line 4 of the source snippet (msg = 'Python\'s World'), what structural purpose does the backslash (\) serve? A) It marks the end of the string literal.

	B) It acts as an escape sequence to include a literal single quote within single quotes. C) It splits the code into a multi-line string block. D) It acts as an arithmetic operator for integer division.
23	What will be the exact output displayed by the shell after executing the final line: print(type(msg))? A) <class 'string'> B) <type 'str'> C) <class 'str'> D) str
24	Which categorization correctly groups the tokens used in the expression <code>x = 15</code> from the shell session? A) x is an Identifier, = is an Operator, 15 is a Literal B) x is a Keyword, = is an Operator, 15 is an Identifier C) x is an Identifier, = is a Punctuator, 15 is a Keyword D) x is a Keyword, = is an Identifier, 15 is a Literal
25	If the teacher adds a line <code>>>> x = y = z = 10</code> to the session, what concept is being demonstrated? A) Multiple type assignment B) Chained / Multiple assignment to single value C) Dynamic switching D) Invalid identifier mapping
Assertion & Reason Questions (Questions 26-30)	
	<i>Directions: In the following questions, a statement of Assertion (A) is followed by a statement of Reason (R). Choose the correct option:</i> A) Both A and R are true and R is the correct explanation of A. B) Both A and R are true but R is NOT the correct explanation of A. C) A is true but R is false. D) A is false but R is true.
26	Assertion (A): The variable name <code>print</code> can technically be used as an identifier in Python without throwing an immediate compilation crash, though it is highly discouraged. Reason (R): <code>print</code> is a built-in function name in Python, not a reserved keyword.
27	Assertion (A): Python programs require significantly fewer lines of code to express concepts compared to languages like Java or C++. Reason (R): Python is a low-level assembly language that directly interacts with the system hardware.
28	Assertion (A): Running the code block <code>val = input()</code> and entering the number 45 sets the data

	type of val as an integer (int). Reason (R): The built-in input() function always captures user inputs natively as a string (str) literal.
29	Assertion (A): Code comments written after the # symbol are ignored completely by the Python interpreter during execution. Reason (R): Comments are meant entirely for human programmers to improve code readability and documentation.
30	Assertion (A): An expression like a = 0o12 is recognized by Python as a valid numeric literal representation. Reason (R): Python prefixes octal literal integer values using a zero followed by a lowercase or uppercase letter 'O' (0o or 0O).

ANSWERS: GETTING STARTED WITH PYTHON

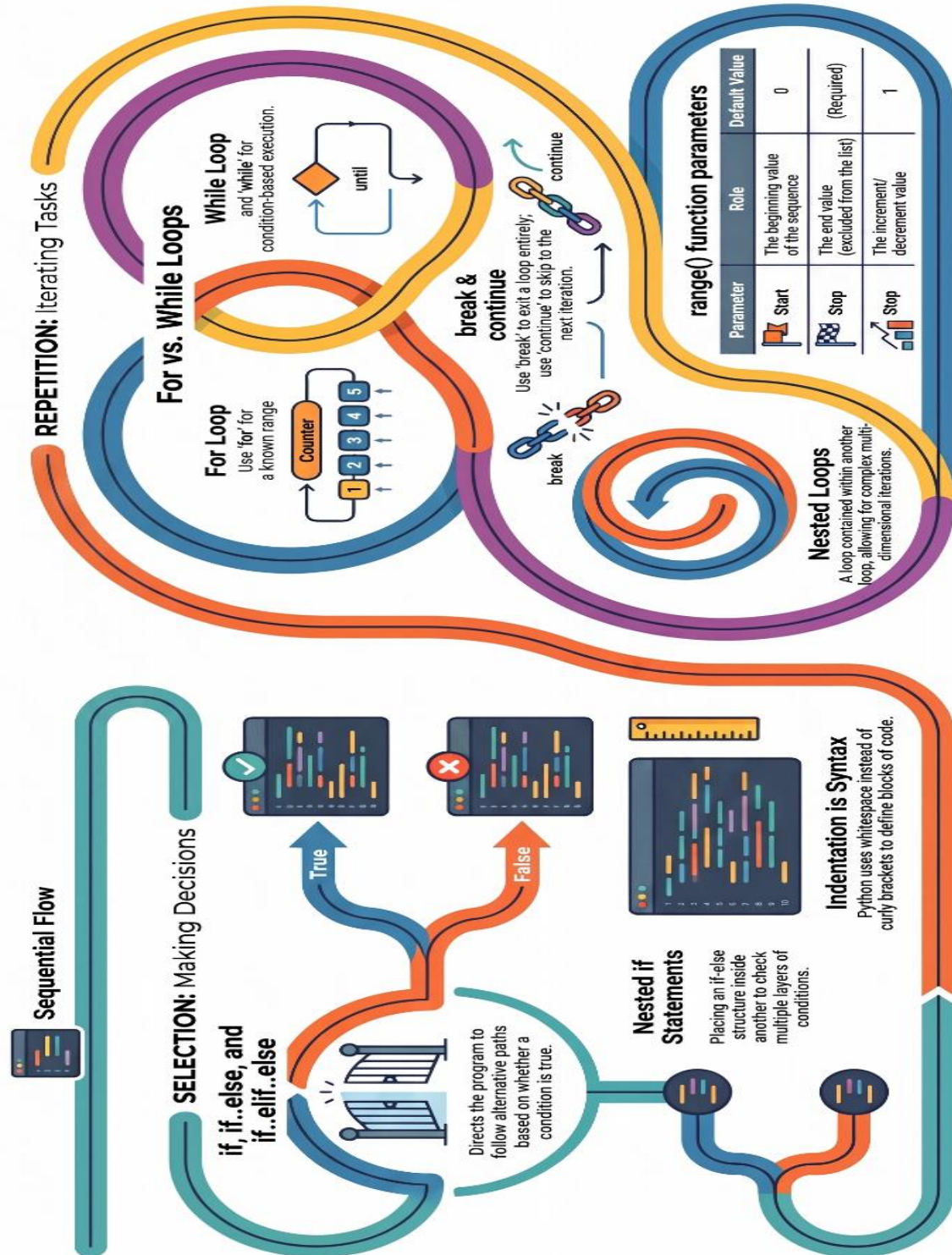
1	B	2	C	3	C
4	C	5	B	6	C

7	When a Python script is run, the source code (.py) is first compiled behind the scenes into an intermediate, lower-level format called Bytecode (.pyc). This bytecode is then fed into the Python Virtual Machine (PVM), which is the interpreter that translates it line-by-line into binary machine language that the CPU understands.
8	Interactive Mode: Useful for running rapid individual experiments. You type a line, hit Enter, and immediately see the outcome. Code is not saved. Script Mode: Useful for writing multi-line programs. Code is saved into a file (with a .py extension) so it can be distributed, modified, and executed repeatedly.
9	Error Type: SyntaxError: cannot assign to literal (or a similar structural assignment error). Reason: The assignment operator (=) evaluates right-to-left. The left-hand side must be a mutable storage path (a variable identifier), not a fixed literal value like 100. Corrected Statement: score = 100
10	A Token is the smallest individual lexical unit in a program's source code that holds meaning for the interpreter. The 5 major categories are: 1. Keywords (e.g., if, for) 2. Identifiers (e.g., user_age, total)

	3. Literals / Constants (e.g., 3.14, "Hello") 4. Operators (e.g., +, //, ==) 5. Punctuators (e.g., [], { }, ,, :)
11	Corrected Code: <pre>name = input("Enter your name: ") print("Welcome", name) Age = int(input("Enter age: ")) # Typecast string to integer print(Age + 5)</pre>
12	sep (Separator): Dictates what character strings go between multiple arguments separated by commas inside a single print() function call. Defaults to a single space (" "). end (End character): Dictates what character prints at the absolute end of the print statement execution. Defaults to a newline escape ("\n"). Example: <code>print("A", "B", sep="-", end="!")</code> # Output: A-B!
13	Case-sensitivity means that Python distinguishes between uppercase and lowercase configurations of alphabetical letters. The variables Value, value, and VALUE point to three completely different object slots in your system's memory. Example: <code>age = 25</code> <code>print(Age)</code> # NameError: name 'Age' is not defined
14	A Literal represents a fixed, raw value assigned directly to a variable inside the code. Three distinct types are: 1. String Literals: Text sequences enclosed in quotes (e.g., "Delhi"). 2. Numeric Literals: Whole or fractional numbers (e.g., 45, 98.6). 3. Boolean Literals: Logical values containing either True or False.
15	<pre>name = input("Enter name: ") birth_year = int(input("Enter birth year: ")) approx_age = 2026 - birth_year # Adjust based on the current year print(f"Hello {name}, you are approximately {approx_age} years old.")</pre>

16	B	17	C	18	A	19	B	20	B
21	B	22	B	23	C	24	A	25	B
26	A	27	C	28	D	29	B	30	A

Python Fundamentals: Mastering Flow of Control



SN	QUESTION
	Select Response type Questions (MCQ -20 %)-(06)
1	Which structural element is structurally missing from the end of the conditional clause below that will result in a compilation Syntax Error? <pre>if score >= 90 print("Grade A")</pre> A) ; B) . C) : D) {}
2	How many times will the body of the following while loop execute? <pre>x = 1 while x < 5: x += 1 if x == 3: break</pre> A) 1 B) 2 C) 3 D) 4
3	What is the generated output of this code snippet? <pre>for num in range(2, 8, 3): print(num, end=" ")</pre> A) 2 5 B) 2 5 8 C) 3 6 D) 2 3 4 5 6 7
4	Consider the expression not (A and B). If A = True and B = False, how does the evaluation rule process this conditional? A) It evaluates to False because both parameters must be true. B) It evaluates to True because the inner conjunction is false, which is then negated. C) It raises a TypeError due to invalid boolean operators. D) It evaluates to None.
5	Which keyword allows a programmer to bypass execution for the current iteration of a loop and immediately evaluate the loop's conditional expression for the subsequent cycle? A) break B) pass C) continue D) return
6	What is the structural value of x after the following conditional code block resolves? <pre>x = 10 if x > 5: x = x + 5</pre>

	<pre> elif x > 10: x = x * 2 else: x = x - 5 </pre> A) 10 B) 20 C) 15 D) 5
	Constructed response questions (SHORT ANSWER AND LONG ANSWER)
7	Rewrite the given while loop code structure using a clean, deterministic for loop: <pre> count = 5 while count <= 25: print(count) count += 5 </pre>
8	Analyze the following implementation and determine the exact string output printed to the terminal console: <pre> p = 1 for q in range(1, 4): p *= q print("Final Product:", p) </pre>
9	A developer writes an if-elif ladder where the conditions read: if x > 10: followed by elif x > 20: Explain why the elif block will never execute, regardless of the value of x.
10	Identify the two syntax errors in the snippet below and provide the corrected version: <pre> value = int(input("Enter number: ")) if value % 2 = 0: print("Even number") </pre>
11	Predict the line-by-line output of the following nested loop block: <pre> for outer in range(1, 3): for inner in range(2): print(outer, inner, sep="-") </pre>
12	Write a compact Python program that accepts an integer from the user and uses conditional execution to print whether the number is positive, negative, or zero.
13	Trace the execution steps of the program below and write down the final value stored in the

	<pre> variable total_sum: total_sum = 0 for i in range(1, 6): if i == 4: continue total_sum += i </pre>
14	What unique behavior occurs when an else clause is tied to a while loop? Under what specific scenario will the code inside this else block fail to run?
15	Using short logical short-circuiting rules, determine what the variable res will hold after running: <code>res = (5 > 3) or (10 / 0 == 2)</code> . Explain why this code does not crash.
	Case Study: Case-Based Questions (Questions 16–19)
	A regional automated transport network relies on an embedded Python script to calculate passenger ticketing fares based on age criteria and travel hours (Peak hours vs. Off-Peak hours). The baseline ticket price is fixed at ₹50. The core logic snippet is laid out as follows: <pre> age = int(input("Passenger age: ")) is_peak = input("Peak hours? (yes/no): ") == "yes" fare = 50 if age < 12: if is_peak: fare = fare - 10 else: fare = fare - 25 elif age >= 60: fare = fare - 20 else: if is_peak: fare = fare + 15 print("Final Fare: ₹", fare, sep="") </pre>
16	A senior citizen passenger aged 65 boards a train during high-density traffic Peak hours (is_peak = True). According to the structural code path, what final fare is printed? A) ₹50 B) ₹30 C) ₹45 D) ₹65

17	A child passenger aged 8 travels during Off-Peak hours. Trace the nested conditional block execution to find their ticket cost. A) ₹25 B) ₹40 C) ₹50 D) ₹15
18	What would occur if an administrative clerk runs the script and inputs an age value of 12? Which condition matches, and what is the fare? A) Matches age < 12, fare is ₹25 or ₹40. B) Fall-through straight to the outer else structure; final fare depends on peak hour conditions. C) Matches age >= 60, fare is ₹30. D) Generates a structural runtime logic error crash.
19	If the public transport authority decides to offer free rides to children under 5 only during off-peak hours, how should the inner conditional under if age < 12: be structurally extended? A) By adding an nested elif age < 5 and not is_peak: fare = 0 clause. B) By wrapping the entire program inside an infinite while loop structure. C) By eliminating the is_peak flag entirely from memory. D) By replacing the if block with a single pass instruction statement.
Source-Based Integrated Questions (Questions 20–22)	
	Source Document: Analyze the textbook algorithm template below designed to calculate the mathematical factorial of a user-supplied target number: <pre># Source Reference: Factorial Engine Validation target = int(input("Calculate factorial for: ")) factorial = 1 if target < 0: print("Invalid negative domain execution.") else: while target > 0: factorial = factorial * target target -= 1 print("Result Value:", factorial)</pre>
20	If the user runs the validation script and inputs a target integer of 0, what will be the

	exact execution path and eventual printed result? A) The while loop executes indefinitely resulting in an out-of-memory crash. B) The program execution flow skips the while loop entirely because the conditional $0 > 0$ evaluates to False, printing Result Value: 1. C) The code jumps immediately to the negative domain warning message block. D) The variable factorial defaults to a value of 0.
21	Which statement best describes the termination condition of the loop inside the source template? A) It is count-controlled and runs exactly 10 times. B) It is an event-controlled loop dependent on the variable target systematically decrementing down until it no longer satisfies the condition $\text{target} > 0$. C) It relies on an explicit internal break statement keyword trigger. D) It relies on a structural indentation change to exit the looping workspace.
22	What structural bug would be introduced into the program execution flow if the decrement update step $\text{target} -= 1$ on line 10 was completely deleted? A) The program would throw a runtime NameError block exception. B) The loop will convert itself into a one-pass execution step structure. C) The loop condition will remain permanently True if the initial input was greater than 0, trapping the program inside an infinite loop. D) The value of factorial will rapidly approach zero.
	Assertion-Reasoning Based Questions (Questions 23–26)
	Directions: Select the choice matching these structural rules: A: Both (A) and (R) are true, and (R) is the correct explanation of (A). B: Both (A) and (R) are true, but (R) is <i>not</i> the correct explanation of (A). C: (A) is true, but (R) is false. D: (A) is false, but (R) is true
23	Assertion (A): The loop string statement <code>for i in range(5):</code> will run its code block values down an array of steps exactly 5 times. Reason (R): The <code>range(n)</code> function generates an unalterable continuous sequence of numbers beginning at 0 and terminating at $n - 1$.
24	Assertion (A): Indentation shifts inside loop structures or conditional statements are purely

	optional formatting choices to make code look nice for developers. Reason (R): Python interprets blocks of code and scope layers based entirely on strict whitespace indentation levels instead of curly brackets {}.
25	Assertion (A): The execution of an empty loop framework can be syntactically safeguarded by inserting the placeholder keyword pass. Reason (R): The pass token acts as a null operation statement that avoids generating an architectural compilation error.
26	Assertion (A): A program containing an explicit structural loop condition set as while True: will <i>always</i> freeze the system context and can never be naturally exited. Reason (R): An internal conditional flow control check using a break statement can forcibly terminate an active loop context instantly, sending the pointer downstream.
Long Answer Type Questions (Questions 27–30)	
27	A computer science student needs a clear architectural framework for loop selection. 1. Compare entry-controlled loops with exit-controlled loops. Clarify which type Python natively implements. 2. Provide a fully annotated Python script demonstrating a scenario where a while loop structure is highly preferred over a traditional for loop format.
28	A student writes the script below to print all prime numbers between 1 and 20. The code contains syntax, indentation, and structural logical bugs. Identify and explain five distinct errors, then provide the corrected script. <pre># Prime Extraction Tool print("Extracting Prime Numbers...") for num in range(2, 21) isprime = True for i in range(2, num): if num % i == 0 isprime = False continue if isprime = True: print(num)</pre>
29	Trace the execution flow of the highly nested architecture detailed below. You must

provide a structured trace table tracking each state update of variables outer, inner, and accumulator.

accumulator = 0

for outer in range(1, 4):

 for inner in range(1, 3):

 if (outer + inner) % 2 == 0:

 accumulator += (outer * inner)

 else:

 accumulator -= 1

print("Final Value Matrix:", accumulator)

30

Design a robust, interactive, menu-driven Python script that continuously offers a user two math operations:

1. Calculate the sum of natural numbers up to N .

2. Check if a user-supplied number is an Armstrong number (e.g., $153 = 1^3 + 5^3 + 3^3$).

The script must continuously loop back to display the main choices menu until the user inputs an explicit value (3) to terminate the session execution loop.

ANSWERS : FLOW OF CONTROL - CONDITIONAL STATEMENTS

1	C	2	B	3	A
4	B	5	C	6	C

7

for count in range(5, 26, 5):
 print(count).

8

Iteration 1 ($q = 1$): $p = 1 * 1 = 1$

Iteration 2 ($q = 2$): $p = 1 * 2 = 2$

Iteration 3 ($q = 3$): $p = 2 * 3 = 6$

Terminal Output: Final Product: 6

9

Python conditions evaluate sequentially from top to bottom. Any value of x large enough to trigger the elif branch (e.g., $x = 25$) will always satisfy the top condition ($x > 10$) first. The structure will run the if block and skip the rest of the chain entirely, rendering the elif statement mathematically unreachable code.

10

Error 1: Line 2 uses a single assignment sign (=) inside a condition instead of the relational

	equality operator (==). Error 2: Line 3 is missing its mandatory structural code block indentation. <pre>value = int(input("Enter number: ")) if value % 2 == 0: print("Even number")</pre>
11	<pre>1-0 1-1 2-0 2-1</pre>
12	<pre>num = int(input("Enter an integer: ")) if num > 0: print("Positive") elif num < 0: print("Negative") else: print("Zero")</pre>
13	<pre>i = 1: total_sum = 0 + 1 = 1 i = 2: total_sum = 1 + 2 = 3 i = 3: total_sum = 3 + 3 = 6 i = 4: Triggers continue, immediately skipping the sum step. i = 5: total_sum = 6 + 5 = 11</pre> Final Value: 11
14	An else block tied to a loop executes if and only if the loop terminates naturally (when the loop condition evaluates to False). The loop's else block will fail to run if the execution path exits the loop prematurely via a break statement.
15	The variable res will hold True. Python implements short-circuit logic for logical or statements. Because the first condition (5 > 3) evaluates directly to True, the net expression outcome is guaranteed to be true. The interpreter completely skips parsing the right-side statement (10 / 0 == 2). Because that section is never executed, the program avoids throwing a runtime ZeroDivisionError and runs smoothly.

16	B	17	A	18	B	19	A
20	B	21	B	22	C	23	A
24	D	25	A	26	D	---	---

27	1. Entry-Controlled Loops: Check their conditional expression <i>before</i> executing the loop body. If the condition evaluates to false on the very first check, the body is never executed. Python's for and while loops are strictly entry-controlled. Exit-Controlled Loops: Execute the loop body first, and then evaluate the termination condition at the end of the iteration. This guarantees the body runs at least once (e.g., do-while in C++). Python does not have a native exit-controlled loop structure. 2. Highly Preferred Scenario Sample for a while Loop: A while loop is preferred when the number of iterations is not known in advance (event-driven program flow control). # Scenario: Keep accepting data inputs from a sensor loop until an error code text 'STOP' is found <pre>sensor_reading = "" while sensor_reading != "STOP": sensor_reading = input("Enter current sensor metric (or type 'STOP' to end): ") if sensor_reading != "STOP": print("Processing metric value...", float(sensor_reading) * 1.18) print("System disconnected smoothly.")</pre>
28	Identified Structural Bugs: - Line 3: Missing a colon character (:) at the close of the outer for loop statement block declaration. - Line 6: Missing a colon character (:) at the close of the conditional if num % i == 0 check statement. - Line 8: Using continue instead of break. Once a factor is found, you should stop checking numbers immediately to save runtime resources. - Line 9: Invalid conditional comparison. It uses a single assignment operator = instead of a relational equity operator check ==. - Line 9 & 10: Faulty alignment indentation structure. The print routine is incorrectly aligned with the inner loop execution workspace instead of sitting outside it. # Prime Extraction Tool Corrected

	<pre> print("Extracting Prime Numbers...") for num in range(2, 21): isprime = True for i in range(2, num): if num % i == 0: isprime = False break # Fixed statement if isprime == True: # Fixed conditional comparison & indentation print(num) </pre>
29	Final Value Matrix: 5
30	<pre> while True: print("\n--- MATHEMATICAL MATRIX MENU ---") print("1. Calculate Sum of First N Natural Numbers") print("2. Verify Armstrong Number Status") print("3. Terminate Program Execution Session") choice = int(input("Enter choice operation option input selection (1-3): ")) if choice == 1: n = int(input("Enter value boundary metric N: ")) total_sum = 0 for i in range(1, n + 1): total_sum += i print("Sum of natural numbers up to", n, "is:", total_sum) elif choice == 2: num = int(input("Enter number to check: ")) temp = num sum_digits = 0 num_str = str(num) num_len = len(num_str) while temp > 0: digit = temp % 10 </pre>

```
sum_digits += (digit ** num_len)
temp //= 10
if num == sum_digits:
    print(num, "is an Armstrong number.")
else:
    print(num, "is NOT an Armstrong number.")
elif choice == 3:
    print("Exiting execution environment framework grid safely. Goodbye!")
    break # Exits the infinite menu structure cleanly
else:
    print("Invalid operational choice variable index token. Please select again.")
```


	<pre>s = s*10 + n%10 n //= 10 print(s)</pre> a) 1234 b) 4321 c) 9999 d) Infinite loop
9	Which of the following numbers will be printed? <pre>for num in range(10, 20): if num % 2 == 0: continue print(num, end=" ")</pre> a) All numbers 10 to 19 b) Only even numbers c) Only odd numbers d) None
10	What will this code display? <pre>a, b = 0, 1 for i in range(5): print(a, end=" ") a, b = b, a+b</pre> a) 0 1 2 3 4 b) 0 1 1 2 3 c) 0 1 1 2 3 5 d) Error
11	What will be the last value of n printed? <pre>n = 1 while n < 100: n = n * 2 print(n)</pre> a) 128 b) 100 c) 64 d) Infinite loop
12	Which of the following statements about break and continue is correct? a) break stops the entire program, continue skips a loop iteration b) break exits only the current loop, continue moves to the next iteration c) Both are the same d) None of the above
13	Choose the correct statement about loops in Python: a) Only for loops can be nested b) break can only be used in for loops

	c) while and for loops can both use break and continue d) Python does not support nested loops
14	Assertion (A): A for loop is preferred when the number of iterations is known. Reason (R): The range() function helps generate a fixed sequence of numbers.
15	Assertion (A): A while True loop always results in an infinite loop. Reason (R): It cannot be terminated once started.
16	Which of the following is a valid python if statement? a) if a >=2: b) if (a>=2) c) if (a=> 22) d) if a >=22
17	How many times will the given loop run? i=2 while(i>0): i=i-1 a)1 b) 3 c) 2 d) 0
	Question Type: Find Output / Short Answer
18	What is a loop? Why are loops used in programming?
19	Differentiate between for loop and while loop with examples.
20	What is the role of the range() function in a for loop? Give an example.
21	What is an infinite loop? Give an example.
22	Write a program using loops to calculate factorial of a number.
23	What is the difference between break and continue statements?
24	Why is indentation important in loops in Python? :
25	Write a program to print Pyramid Pattern * *** ***** *****
26	Write a program to check whether a given number is Palindrome or not Input: 1221 → Output: Palindrome
27	Convert the following for loop code into while loop. for x in range (1,100,2): print(x)

28	Find error(s) in the following code (if any) . correct them by rewriting the code and underline the corrections. <pre> x= int("Enter Value for x:") for in range[0,11]: if x=y print x+y else: print x-y </pre>
29	Write a program to find sum of first 10 natural numbers.
Question Type: Find Output / Long Answers	
30	Write a program to print all prime factors of a given number (e.g., 84 → 2 2 3 7).
31	Write a program to compute the sum of the series: $S=1+1/2+1/3+\dots+1/n$ for n = 10.
32	Write a program to check if a number is armstrong.
33	Convert a decimal number to binary using a loop.
34	Write a program to print Fibonacci series up to 10 terms.

Answers: Additional Questions: Loops in Python

1. c) do-while loop	2. c) if $n \% 3 == 0$ and $n \% 5 == 0$:
3. b) for I in range(10, 0, -1): print(i)	4. : b) It skips the current iteration and continues with the next one.
5. b) It will result in an infinite loop	6. c) range()
7. b) 5 3 1	8. a) 1234 (reverse of 4321)
9. c) Only odd numbers	10. b) 0 1 1 2 3
11. a) 128	12. b) break exits only the current loop, continue moves to the next iteration
13. c) while and for loops can both use break and continue	14. Both A and R are true, and R is the correct explanation of A. ✓

15. A is true, but R is false (it can be terminated with break)

16. a) if a >=2:

17. c) 2

18. A loop is a control structure used to execute a block of code repeatedly as long as a condition is true. Loops are used to reduce repetition and make programs shorter and more efficient.

19. for loop → used when the number of iterations is known.

```
for i in range(5):
```

```
    print(i)
```

- while loop → used when the number of iterations is not fixed but depends on a condition.

```
i = 0
```

```
while i < 5:
```

```
    print(i)
```

```
    i += 1
```

20. The range() function generates a sequence of numbers which the loop iterates over.

Example:

```
for i in range(1, 6):
```

```
    print(i) # prints numbers 1 to 5
```

21. An infinite loop is a loop that never ends because its condition is always true.

Example:

```
while True:
```

```
    print("Hello")
```

22. n = int(input("Enter any number"))

```
fact = 1
```

```
for i in range(1, n+1):
```

```
    fact *= i
```

```
print("Factorial:", fact)
```

23. break: exits the loop completely.

- continue: skips the current iteration and moves to the next one.

24. Indentation defines the block of statements inside the loop. Without proper indentation, Python

will raise an IndentationError.

```
25. rows = 4
    for i in range(1, rows+1):
        print(" " * (rows-i) + "*" * (2*i-1))
```

```
26.
n = int(input("Enter any number"))
temp, rev = n, 0
while temp > 0:
    rev = rev*10 + temp % 10
    temp //= 10
if rev == n:
    print("Palindrome")
else:
    print("Not Palindrome")
```

```
27.
x = 1
while x < 100:
    print(x)
    x += 2
```

```
28.
x= int(input("Enter Value for x:"))
for y in range(0,11):
    if x==y:
        print (x+y)
    else:
        print( x-y)
```

```
29.
s=0
for i in range(1,11):
    s+=i
print(s) #55
```

```
30.
n = int(input("Enter any number"))
i = 2
while n > 1:
    if n % i == 0:
        print(i, end=" ")
        n //= i
    else:
        i += 1
```

```
31.  
n = 10  
s = 0  
for i in range(1, n+1):  
    s += 1/i  
print("Sum =", s)
```

```
32.  
n = int(input("Enter anynumber"))  
s = 0  
temp = n  
while temp > 0:  
    digit = temp % 10  
    s += digit ** 3  
    temp //= 10  
if s == n:  
    print("Armstrong")  
else:  
    print("Not Armstrong")
```

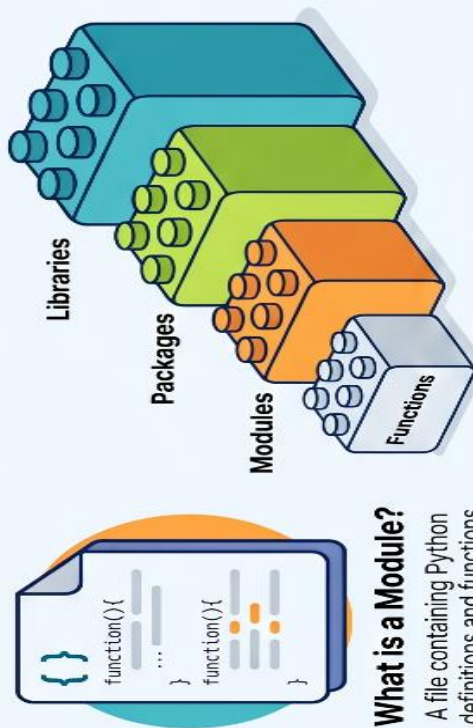
```
33.  
n =int(input("ENter any no"))  
binary = ""  
while n > 0:  
    binary = str(n % 2) + binary  
    n //= 2  
print("Binary =", binary )
```

```
34.  
a,b=0,1  
for i in range(10):  
    print(a,end=' ')  
    a,b=b,a+b
```

Mastering Python Modules: Organize and Empower Your Code

Python modules allow programmers to organize complex code by partitioning it into individual, reusable components. This blueprint outlines the structural hierarchy of Python libraries and provides a quick-start guide to the most common built-in modules used for data calculation and randomness.

The Foundation of Modularity



What is a Module?

A file containing Python definitions and functions that organizes code for easier use.

Why Use Modules?



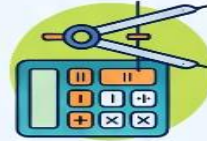
Essential Standard Modules

```
import math
result = math.sqrt(25)
```

Accessing Modules via "Import"

Use dot notation (e.g., `math.sqrt`) to reference functions after importing the module.

The Math & Statistics Modules



Access advanced mathematical functions like square roots and statistical data like mean or median.

Module	Keywords
math	Square roots, powers, and trigonometry
statistics	Calculating Mean, Median, and Mode

The Random Module



Generate random integers, floating-point numbers, or values within a specific range and step.



Generating random numbers and ranges

SN	Questions
Select Response type Questions (MCQ -20 %)-(06)	
1	Which keyword is used in Python to import an entire module into the current workspace? A) include B) require C) import D) extend
2	If you want to use only the sqrt function from the math module without prefixing it with the module name, which statement should you use? A) import sqrt from math B) from math import sqrt C) import math.sqrt D) from math use sqrt
3	Which of the following functions belongs to the standard built-in statistics module in Python? A) pow() B) mean() C) randint() D) floor()
4	What is the default file extension for a standalone, reusable Python module file? A) .mod B) .pyc C) .txt D) .py
5	What will be the output of the expression math.ceil(4.3)? A) 4 B) 5 C) 4.0 D) 5.0
6	Which function from the random module returns a random floating-point number N such that $0.0 \leq N < 1.0$? A) random() B) uniform() C) randint() D) randrange()
Constructed response questions (SHORT ANSWER AND LONG ANSWER)	
7	What is a module in Python, and why is it useful?
8	Differentiate between import math and from math import *.
9	Write a Python snippet that imports the random module and uses it to generate a random integer between 10 and 50 (both inclusive).
10	Explain the purpose and output of the math.floor() function with a brief example.
11	What is a namespace collision, and how does using a module alias prevent it?
12	Look at this code snippet. Will it work? If not, explain the issue: <pre>import math print(pow(2, 3))</pre>
13	How does random.randint(1, 6) differ from random.randrange(1, 6)?
14	Name three mathematical constants provided by the built-in math module in Python.
15	Briefly explain what happens behind the scenes when Python executes an import statement for the first time in a program script.

	Assertion & Reason Questions (Questions 16-20)
	<i>Directions: Choose the correct option:</i> <i>A) Both A and R are true and R is the correct explanation of A.</i> <i>B) Both A and R are true but R is NOT the correct explanation of A.</i> <i>C) A is true but R is false.</i> <i>D) A is false but R is true.</i>
16	Assertion (A): The statement from <code>math import pi, sin</code> is generally preferred over <code>from math import *</code> in professional software development. Reason (R): Explicitly importing only required components keeps the namespace clean and prevents accidental overriding of existing functions.
17	Assertion (A): The function <code>random.randint(5, 5)</code> will throw a <code>ValueError</code> because the start and stop parameters are identical. Reason (R): <code>randint(a, b)</code> requires that the value of parameter <code>a</code> must strictly be less than parameter <code>b</code>.
18	Assertion (A): The <code>statistics.median()</code> function automatically handles both odd and even-sized lists of numerical data elements. Reason (R): For an even number of data points, the median function calculates the average of the two middlemost values.
19	Assertion (A): You can create your own custom Python module by saving a set of Python functions into a text file named <code>test.module</code>. Reason (R): The Python interpreter recognizes any file with a <code>.py</code> extension as a source code file or module.
20	Assertion (A): The math function call <code>math.sqrt(-16)</code> will execute smoothly and output <code>4j</code> in a standard Python script. Reason (R): The built-in <code>math</code> module deals exclusively with real numbers and raises a <code>ValueError</code> if passed a negative value.
	Case-Based Integrated Questions (Questions 21-25)
	Case Study: A physical education instructor wants to build a simple digital program to assist with sports day tracking. The program needs to do three specific tasks: 1. Randomly select a student's token number out of a group of 30 contestants to win a

	raffle. - Find the average (mean) score of all structural agility tests. - Calculate the exact square root of the final team track timings. A student coder writes the initial structural layout: <pre>import math import random import statistics scores = [88, 92, 79, 85, 94, 90]</pre>
21	Which function statement should the student use to select a completely random integer between token number 1 and token number 30 inclusive? A) random.random(1, 30) B) random.randint(1, 30) C) random.randrange(1, 30) D) statistics.choice(1, 30)
22	The user wants to calculate the mean of the scores list using the imported modules. Which code line achieves this? A) mean = math.mean(scores) B) mean = statistics.average(scores) C) mean = statistics.mean(scores) D) mean = scores.mean()
23	If the total team running time is saved in a variable as total_time = 14.56, what is the correct syntax to compute its square root using the imported module? A) res = sqrt(total_time) B) res = math.sqrt(total_time) C) res = math.pow(total_time, 0.5) D) Both B and C are valid syntax lines
24	Suppose the student wants to clean up the code and type less module names. They rewrite the top of the file to read: <pre>from statistics import mean as m</pre> How must they now call the mean function later in the script? A) statistics.m(scores) B) mean(scores) C) m(scores) D) statistics.mean(scores)
25	If the scores list becomes empty during runtime due to an input error, what will happen if statistics.mean(scores) is invoked? A) It returns 0. B) It returns None. C) It raises a StatisticsError. D) It gets trapped in an infinite loop.
	Long Answer Type Questions- (Questions 26-30)
26	A student executes the following script. Out of the given options (A) to (D), find all the possible

	outputs that can be displayed on the screen. <pre>import random # randrange(start, stop, step) num = random.randrange(2, 9, 3) for i in range(num): print(i, end="*")</pre> A. 0*1* B) 0*1*2*3*4* C)0*1*2*3*4*5* D)0*1*2*3*4*5*6*7*
27	Study the following code carefully. What is the minimum and maximum possible value that can be assigned to the variable index? Write one possible output of the program. <pre>import random cities = ["Delhi", "Mumbai", "Chennai", "Kolkata"] # randint is inclusive of both endpoints index = random.randint(0, 2) print(cities[index], end="!!")</pre>
28	What will be the exact output when the following Python code is executed? <pre>import math val1 = math.ceil(5.2) val2 = math.floor(5.8) val3 = math.floor(-3.4) print(val1, val2, val3, sep="#")</pre>
29	A school administrator needs to process exam scores to calculate core metrics. Write a complete Python program that imports the statistics and math modules to perform the operations listed below. You must define a list containing the scores [78, 85, 92, 85, 90, 85, 100, 71, 88]. 1. Calculate the standard mathematical average (mean) score. 2. Find the most common recurring score (mode). 3. Determine the middle score (median) of the dataset. 4. Calculate the variance of the grades. 5. Compute the square root of the variance, and use a math function to round that final floating-point value <i>up</i> to the nearest integer.
30	A student writes the following broken program script aiming to generate three random floating-point values between 10.0 and 20.0, calculate their statistical median, and round down the

result. The code contains multiple syntax, naming, and module usage errors.

```
# Broken Student Script
```

```
from random import random
```

```
import statistics as stat
```

```
values = []
```

```
for i in range(1, 4):
```

```
    val = random.uniform(10.0, 20.0)
```

```
    values.append(val)
```

```
print("Generated list: " + values)
```

```
final_median = statistics.median(values)
```

```
print("Rounded Median: " + math.floor(final_median))
```

Identify, list, and explain the bugs within this script. Then, write the corrected version of the program.

ANSWERS: INTRODUCTION TO PYTHON MODULES

1	C	2	B	3	B
4	D	5	B	6	A

7	A module is a file containing Python definitions, functions, and statements that can be reused across different programs. Modules encourage code reusability, help organize large projects into manageable chunks, and prevent naming conflicts.
8	import math imports the entire module. To use its functions, you must prefix them with the module name (e.g., math.sqrt(16)). from math import * imports all functions directly into the current namespace. You can call them directly (e.g., sqrt(16)), but it risks naming collisions with your own variables.
9	<pre>import random num = random.randint(10, 50) print(num)</pre>
10	The math.floor() function returns the largest integer less than or equal to the given number (it effectively rounds down). For example, math.floor(7.9) returns 7, and math.floor(-3.2) returns -4.
11	A namespace collision occurs when two different modules have functions with the exact same

	name, confusing the program. An alias (using as) renames a module locally. For example, import statistics as st allows you to call st.mean() safely without conflicting with a custom function named mean.								
12	Yes, it will work, but it will execute Python's standard built-in pow() function, not the one from the math module. If the developer intended to use the math module's version, they should write math.pow(2, 3).								
13	random.randint(1, 6) includes both boundaries, meaning it can generate 1, 2, 3, 4, 5, or 6. random.randrange(1, 6) excludes the upper boundary, meaning it will only generate 1, 2, 3, 4, or 5.								
14	math.pi (Ratio of a circle's circumference to its diameter, roughly 3.14159) math.e (Euler's number, roughly 2.71828) math.tau (Equivalent to 2π , roughly 6.28318)								
15	Python searches for the module file (e.g., mymodule.py) in its search paths. Once found, it runs the code inside that module to initialize it, creates a module object, and binds it to a name in the current script so its functions are accessible.								
16	A	17	D	18	A	19	D	20	D
21	B	22	C	23	D	24	C	25	C

LONG ANSWER TYPE QUESTIONS

26	Options (A) and (C) are the only possible outputs.								
27	Minimum value of index: 0 Maximum value of index: 2 Possible Outputs: Any one of the following is correct: • Delhi!! • Mumbai!! • Chennai!!								
28	6#5#-4 Explanation: • math.ceil(5.2) rounds up to the next integer, which is 6. • math.floor(5.8) rounds down to the next integer, which is 5. • math.floor(-3.4) rounds down towards negative infinity. On a number line, the integer								

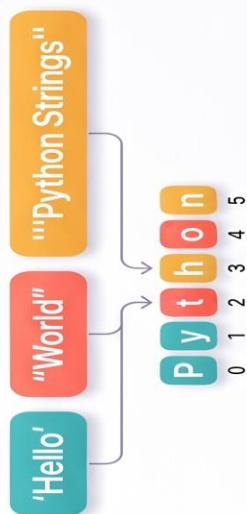
	lower than -3.4 is -4. • <code>sep="#"</code> prints a hash symbol between the values instead of a default space.
29	<pre> import statistics import math # Initializing the dataset containing exam scores scores = [78, 85, 92, 85, 90, 85, 100, 71, 88] # 1. Calculate the average score (Mean) mean_score = statistics.mean(scores) print("1. Mean of the scores:", mean_score) # 2. Find the most common recurring score (Mode) mode_score = statistics.mode(scores) print("2. Mode of the scores:", mode_score) # 3. Determine the middle score (Median) median_score = statistics.median(scores) print("3. Median of the scores:", median_score) # 4. Calculate the variance of the dataset variance_score = statistics.variance(scores) print("4. Variance of the scores:", variance_score) # 5. Compute the square root of the variance std_deviation = math.sqrt(variance_score) print(" Exact Square Root of Variance:", std_deviation) # Round the value UP to the nearest integer using math.ceil() final_rounded_val = math.ceil(std_deviation) print("5. Final rounded up integer (Ceiling):", final_rounded_val) </pre>
30	Corrected Python Code Structure <pre> # Corrected Script import random import statistics as stat import math values = [] # Loop runs exactly 3 times (i = 1, 2, 3) </pre>

```
for i in range(1, 4):  
    # Correctly calling uniform from the imported random module  
    val = random.uniform(10.0, 20.0)  
    values.append(val)  
# Corrected string conversion for list display  
print("Generated list:", values)  
# Using the correct alias 'stat' to find the median  
final_median = stat.median(values)  
print("Exact Median Value is:", final_median)  
# Using math.floor after importing math, with safe comma formatting  
print("Rounded Median (Floored):", math.floor(final_median))
```

Python Strings: The Ultimate Quick-Reference Guide

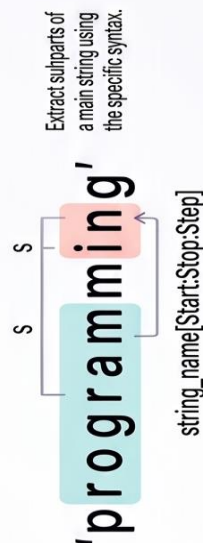
Core Mechanics & Operations

> Creation & Access

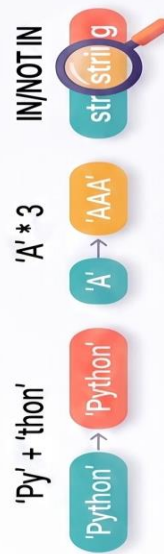


Strings are created using single, double, or triple quotes and accessed via zero-based indexing.

> Advanced Slicing Syntax



> Membership & Replication Operators



Use '+' for concatenation, '*' for repetition, and 'in/not in' to find substrings.

Essential Built-in Methods

> Case & Text Formatting



Effortlessly transform text using methods.

> String Analysis & Validation

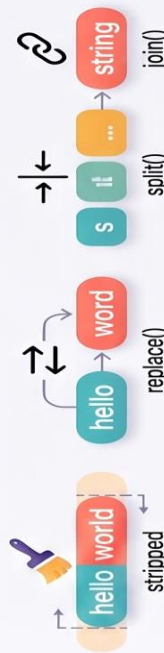


Identify content patterns using methods like find(), count(), isalpha(), isdigit(), and isspace().

Method Examples

Category	Method Examples
Formatting	capitalize(), upper(), title()
Search/Find	find(), index(), count()
Structural	split(), join(), strip()

> Transformation & Cleaning



Modify or restructure strings using strip() for whitespace, replace() for content, and split()/join() for arrays.

Python Strings: A Quick-Reference Guide

CORE PRINCIPLES

- Strings are immutable sequences of characters.
- Create them with single, double, or triple quotes.
- Access text with zero-based indexing and slicing.
- Use concatenation (+), repetition (*), and membership (in / not in).
- Built-in methods make text processing easier.

ACTION STEPS

1. Create strings using `"`, `'`, or triple quotes.
2. Access a character with ``my_string[0]``.
3. Extract substrings with ``my_string[start:stop:step]``.
4. Concatenate strings with ``+``; repeat them with ``*``.
5. Check for substrings with ``in`` or ``not in``.
6. Change case with ``upper()``, ``lower()``, ``capitalize()``, or ``title()``.
7. Search with ``find()``, ``count()``, or ``index()``.
8. Remove whitespace with ``strip()``, ``lstrip()``, or ``rstrip()``.
9. Modify text with ``replace()``.
10. Split text into lists with ``split()``; combine list elements with ``join()``.

PRO TIPS

Use f-strings for efficient, readable formatting.

Strings are immutable: methods return new strings rather than changing the original.

A negative slicing step reverses a string.

``join()`` efficiently combines many strings.

PITFALLS

Indexing a position that does not exist raises an error.

Do not expect string methods to alter the original value.

Incorrect slicing parameters can yield surprising results.

Do not confuse ``split()`` (string → list) **with** ``join()`` (list → string).

KEY TERMS

String: An immutable sequence of characters.

Indexing: Accessing individual characters by numerical position, starting from 0.

Slicing: Extracting a substring using start, stop, and step parameters.

Concatenation: Combining two or more strings with the `+` operator.

Repetition: Repeating a string with the `\*` operator.

Immutability: The property of an object whose state cannot be modified after creation.

REAL-WORLD EXAMPLES

FORMATTING & VALIDATION

Format user input for display with

``capitalize()`` or ``title()``.

Validate data entry with ``isdigit()`` when only numeric text is expected.

PARSING & CLEANING

Parse logs or text data with ``split()`` and delimiters.

Clean user-generated content with ``strip()`` to remove surrounding whitespace.

REFLECTION

- How can string immutability prevent common bugs?
- When would you prefer ``find()`` over ``index()``?
- How can slicing and methods combine into powerful text-processing pipelines?

S.No.	Question Type: MCQ
1	1. Which of the following is a valid way to create a string in Python? a) <code>s = 'Hello'</code> b) <code>s = "Hello"</code> c) <code>s = ""Hello""</code> d) All of the above
2	2. Strings in Python are: a) Mutable sequences of characters b) Immutable sequences of characters c) Lists of numbers d) Changeable character arrays
3	What will be the output of: <code>s = "ComputerScience"</code> <code>print(s[3:8])</code> a) putters b) pute c) puts d) puter

4	Which method is used to find the position of a substring in a string? a) position() b) index() c) findpos() d) locate()
5	Which of these will convert a string s to uppercase? a) s.upper() b) upper(s) c) toUpper(s) d) str.uppercase(s)
6	What will be the output of: print("abc"*3) a) abc3 b) abcabcbabc c) abc*abc*abc d) Error
7	Which method removes whitespace from the beginning and end of a string? a) strip() b) trim() c) remove() d) clear()
8	If s = "Python", which statement is invalid? a) print(s[0]) b) s[1] = "y" c) print(s[-2]) d) print(s[2:])
9	The correct way to check if "th" exists in "Python" is: a) "th" in "Python" b) contains("Python","th") c) "Python".has("th") d) find("th","Python")
10	What will be the result of the following command:- >>>"i love python".capitalize() (a) 'I Love Python' (b) 'I love python' (c) 'I LOVE PYTHON' (d) None of the above
11	Write the correct code for the following output:- Output: 'W\$O\$R\$L\$D' (a) >>>str="\$" >>> join("WORLD",'\$') (c) >>> "\$" = str >>> join("WORLD",'\$') (b) >>>str="\$" >>>str.join("WORLD") (d) >>>str="\$" >>> join("WORLD").str
12	What is the output of: print("apple,banana,grape".split(",")) (a) ['apple banana grape'] (b) Error (c) ('apple', 'banana', 'grape') (d) ['apple', 'banana', 'grape']
13	Which of the following methods can be used to check if a string starts with a particular prefix. (a) end() (b) endswith() (c) finish() (d) startswith()
14	The data type of x in the below code is: txt = "I could eat bananas all day" x = txt.partition("bananas")

	(a) string	(b) List	(c) Tuple	(d) Dictionary
15	What will be the output of the following Code: <pre>>>>"India is my Country. I love my Country India".replace("Country","India")</pre> (a) 'India is my India. I love my India India' (b) Country is my Country. I love my Country Country (c) " India is my Country. I love my Country India". (d) ERROR			
16	What is the output of len("Class XI")? a) 7 b) 8 c) 9 d) 6			
17	Which operator is used to check whether a substring exists in a string? a) == b) in c) is d) %			
	Question Type: Find Output / Short Answer			
18	Explain the difference between find() and index() methods.			
19	Write a Python statement to extract every second character from "InformationTechnology" starting from index 1.			
20	Given text = "PythonProgramming", how would you extract "Program" using slicing only?			
21	Reverse only the first half of the string "ABCDEFGHijkl".			
22	Remove all vowels from the string "Examination".			
23	Extract all digits from the string "CS2025Exam123" into a new string.			
24	Given a sentence, write a Python snippet to find the longest word.			
25	What happens if the slicing range exceeds the string length?			
26	What is the difference between single, double, and triple quotes in string creation?			
27	Suggest appropriate functions for the following tasks – (a) To check whether the string contains digits. (b) To find the occurrence of a string within another string. (c) To convert the first letter of a string to uppercase. (d) To convert all the letters of a string to uppercase. (e) To check whether all the letters of the string are in capital letters.			

	(f) To remove all the white spaces from the beginning of a string.
28	What is string indexing? Give an example.
29	Write a Python statement to count the number of times "a" occurs in "banana". Give the output.
	Question Type: Find Output / Long Answers
30	Explain the concept of string immutability in Python with an example. Why are strings immutable?
31	Write a program to count vowels, consonants, digits, and spaces in a string.
32	Remove duplicate characters from a string.
33	Explain splitting and joining strings with examples.
34	Explain any five commonly used string methods with examples.

Answers: STRINGS

1. d) All of the above	2. b) Immutable sequences of characters
3. a) puterS	4. : b) index()
5. a) s.upper()	6. b) abcabcabc
7. a) strip()	8. b) s[1] = "y" (<i>Strings are immutable</i>)
9. a) "th" in "Python"	10. b) 'I love python'
11. b)	12. d)
13. d)	14. c)
15. a) c	16. b) 8
17. b) in	
18	find() returns -1 if the substring is not found. index() raises a ValueError if the substring is not found.
19	s = "InformationTechnology" print(s[1::2]) # Output: nfrmtinTchnlg
20	text = "PythonProgramming" print(text[6:13]) # Output: Program

21	<pre>s = "ABCDEFGHijkl" half = len(s)//2 print(s[:half][::-1] + s[half:]) # Output: FEDCBA GHijkl</pre>
22	<pre>s = "Examination" vowels = "aeiouAEIOU" result = ''.join(ch for ch in s if ch not in vowels) print(result) # Output: xmntn</pre>
23	<pre>s = "CS2025Exam123" digits = ''.join(ch for ch in s if ch.isdigit()) print(digits) # Output: 2025123</pre>
24	<pre>sentence = "Python is a powerful programming language" words = sentence.split() longest = max(words, key=len) print(longest) # Output: programming</pre>
25	Python does not raise an error; it returns characters up to the string's end.
26	Single and double quotes are used for single-line strings. Triple quotes are used for multi-line strings or docstrings.
27	(a) isalnum() (b) find() (c) capitalize() (d) upper() (e) isupper() (f) lstrip()
28	String indexing is used to access individual characters of a string using position numbers. Indexing starts from 0. Example: <pre>s = "Python" print(s[2])</pre> Output: t
29	<pre>print("banana".count("a"))</pre> Output: 3
30	Definition: Strings in Python are <i>immutable</i>, meaning their characters cannot be changed after creation. Example: <pre>s = "Hello"</pre>

	<pre># Attempt to modify the first character # s[0] = "Y" # This will raise: TypeError s = "Y" + s[1:] # Correct way: create a new string print(s) # Output: Yello</pre> Reason: ○ Improves performance (strings can be safely shared in memory). ○ Helps maintain data integrity in multi-threaded or complex programs.
31	<pre>s = input("Enter text") vowels = "aeiouAEIOU" vowel_count = consonant_count = digit_count = space_count = 0 for ch in s: if ch in vowels: vowel_count += 1 elif ch.isalpha(): consonant_count += 1 elif ch.isdigit(): digit_count += 1 elif ch.isspace(): space_count += 1 print("Vowels:", vowel_count) # 5 print("Consonants:", consonant_count) # 9 print("Digits:", digit_count) # 2 print("Spaces:", space_count) # 4</pre>
32	<pre>s = input("Enter string") result = "" for ch in s: if ch not in result: result += ch print(result)</pre>
33	Split: Divides a string into a list.

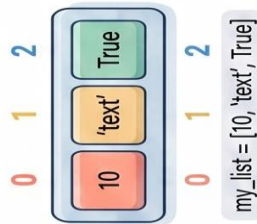
	<pre>sentence = "Python is fun" words = sentence.split() print(words) # ['Python', 'is', 'fun']</pre> Join: Combines a list of strings with a delimiter. <pre>joined = "-".join(words) print(joined) # Python-is-fun</pre> <i>Use Cases: Splitting CSV data, joining file paths.</i>
34	<pre>1. upper() — converts all letters to uppercase. print("Python".upper()) # PYTHON 2. lower() — converts all letters to lowercase. print("PYTHON".lower()) # python 3. strip() — removes spaces from both ends of a string. print(" Hello ".strip()) # Hello 4. replace(old, new) — replaces a substring with another substring. print("I like Java".replace("Java", "Python")) # I like Python 5. count(substring) — counts the occurrences of a substring. print("banana".count("a")) # 3</pre>

Master the Python List: A Visual Guide

Core Concepts & Operations

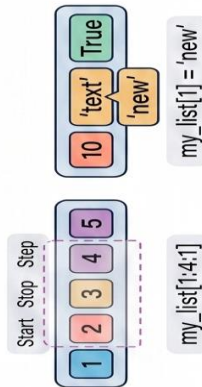
Creation & Indexing

Lists use square brackets `[]` with comma-separated elements and support index-based access.



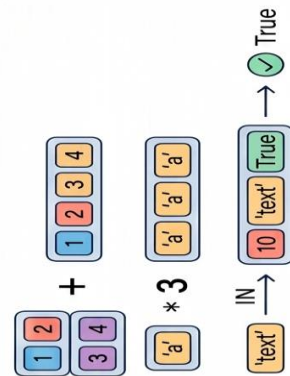
Slicing & Mutability

Access subparts using `[Start:Stop:Step]` and modify elements directly without recreating the list.



Logical & Mathematical Operators

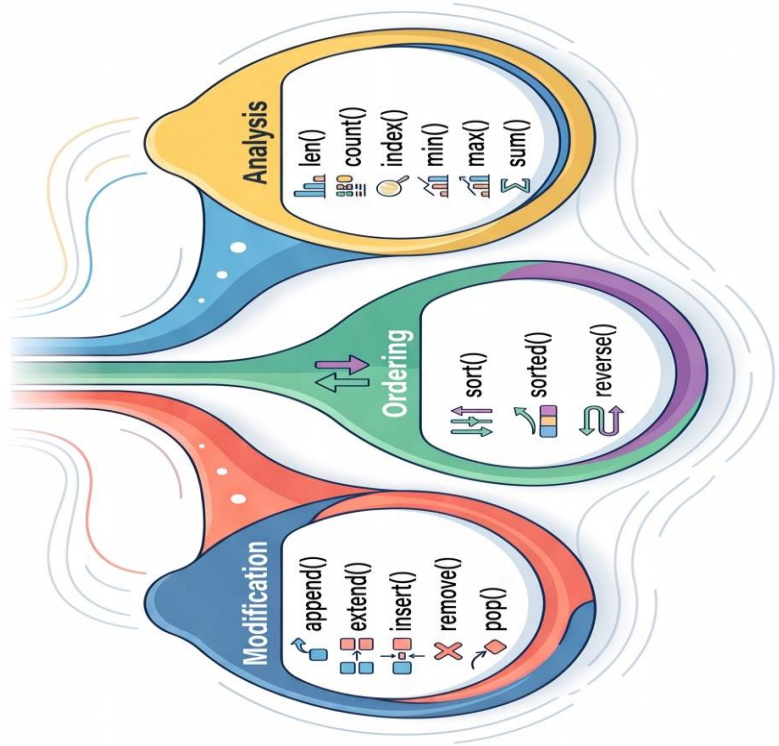
Use `+` to concatenate, `*` to replicate, and `in` to check membership.



Built-in Methods & Functions

Comprehensive Data Management

Python provides numerous methods to add, remove, sort, and analyze list elements.



Mastering Python Lists: A Visual Guide

CORE PRINCIPLES

- Lists are created using square brackets `[]`.
- Elements are separated by commas.
- Lists support zero-based indexing for element access.
- Lists are mutable, so their elements can be changed after creation.
- Slicing accesses sub-parts of a list with `[start:stop:step]`.
- The `+` operator concatenates lists.
- The `*` operator replicates list elements.
- The `in` operator checks membership.

ACTION STEPS

1. Create a list using `my_list = [element1, element2, ...]`.
2. Access an element using `my_list[index]`.
3. Modify an element with `my_list[index] = new_value`.
4. Slice a list with `my_list[start:stop:step]`.
5. Concatenate lists with `list1 + list2`.
6. Replicate a list with `my_list * n`.
7. Check for membership with `element in my_list`.
8. Add an element to the end with `my_list.append(element)`.
9. Insert at a specific position with `my_list.insert(index, element)`.
10. Remove the first occurrence with `my_list.remove(element)`.
11. Remove and return the last element with `my_list.pop()`.
12. Sort the list in place with `my_list.sort()`.
13. Return a new sorted list with `sorted(my_list)`.
14. Reverse the list in place with `my_list.reverse()`.
15. Get the number of elements with `len(my_list)`.
16. Count occurrences with `my_list.count(element)`.
17. Find the first occurrence with `my_list.index(element)`.
18. Find the minimum with `min(my_list)`.
19. Find the maximum with `max(my_list)`.
20. Calculate the sum with `sum(my_list)`.

KEY TERMS

List: An ordered, mutable (changeable) sequence of items in Python.

Index: The position of an element in a list, starting from 0.

Slicing: Extracting a portion of a list using a range of indices.

Mutability: The ability of an object to be changed after it is created.

Concatenation: Joining two or more lists together.

Replication: Repeating the elements of a list multiple times.

Membership: Checking whether an element exists within a list.

REAL-WORLD EXAMPLES

GAME SCORES

Store user scores in a list.

Use `append()` to add scores, `sort()` to rank them, and `len()` to get the total.

PLAYLISTS

Manage a playlist of songs.

Use `insert()` for positions, `remove()` for deletions, and `reverse()` for reverse order.

PROCESSING DATA FROM A FILE

Read lines into a list, use slicing to process specific parts, and use `sum()` or `min()` / `max()` for analysis.

REFLECTION

- How does list mutability compare with the immutability of strings or tuples?
- When would you prefer `sort()` over `sorted()` and vice versa?
- For a large list, which operations are likely to be most computationally expensive?

	Question Type: MCQ
1	Find output produced by following code: lst = [1,2,3] print(lst*2) a) [1,2,3,1,2,3] b) [2,4,6] c) Error d) [1,2,3,6]
2	Which method removes and returns the last item from a list? a) remove() b) del() c) pop() d) discard()
3	What is the result of len([[[]],[1,2],3])?
4	Which of these creates a shallow copy of a list a? a) b = a b) b = a.copy() c) b = list(a) d) Both b and c
5	Which statement deletes the second element of a=[1,2,3]?
6	Write statement to sort a list a in descending order.
7	What is the output? a = [1,2,3] b = a b.append(4) print(a) a) [1,2,3] b) [1,2,3,4] c) [4,1,2,3] d) Error
8	Which statement creates a new list b that is an independent copy of a? a) b = a b) b = a.copy() c) b = list(a) d) Both b and c
9	Evaluate: lst = [10,20,30,40,50] print(lst[-3:1:-1]) a) [30,20] b) [30,40,50] c) [30,40] d) [30]
10	What will be the Output of: nums = [1,2,3,4,5] nums[1:4] = [9,8] print(nums) a) [1,8,9,5] b) [1,9,8,4,5] c) [1,9,8,5] d) [1,9,8]

11	Which method would you use to insert multiple elements at a specific position? a) append() b) extend() c) Slice assignment d) pop()
12	What will be the output? lst = [1,2,3,4,5] print(lst[::-2]) a) [5,3,1] b) [4,2] c) [1,3,5] d) [5,3]
13	What does nums = [None]*5 produce? a) [] b) [0,0,0,0,0] c) [None,None,None,None,None] d) Error
14	Which method removes the first occurrence of a specified value from a list? a) pop() b) del c) remove() d) discard()
15	What will be the output? lst = [1,2,3] print(lst*0) a) [] b) [0,0,0] c) [1,2,3,0] d) Error
16	Which of the following creates a list in Python? a) L = (1, 2, 3) b) L = {1, 2, 3} c) L = [1, 2, 3] d) L = <1, 2, 3>
17	Which list method adds an element at the end of a list? a) insert() b) append() c) remove() d) pop()
18	What is the output of the following code? L = [1, 2, 3] L.insert(1, 10) print(L) a) [10, 1, 2, 3] b) [1, 2, 3, 10] c) [1, 10, 2, 3] d) [1, 2, 10, 3]
	Question Type: Find Output / Short Answer
19	How do shallow copy and deep copy differ when copying nested lists? Give an example.
20	Find the output: lst = [1,2,3,4,5] lst[::-2] = [9,8,7] print(lst)
21	Explain the difference between append() and + for lists.

22	Predict and explain: nums = [1,2,3] print(nums*3 is nums*3)
23	Predict and explain: nums = [1,2,3] print(nums.append([4,5])) print(nums)
24	What will this print and why? a = [1,2,3] print(a * -2)
25	How can you insert multiple elements [7,8] between 3 and 4 in [1,2,3,4,5] without using insert()?
26	What happens here? a = [1,2] a.extend("34") print(a)
27	Explain the difference between nums.sort() and sorted(nums).
28	How can you pair elements from two lists [1,2,3] and [4,5,6] into [(1,4),(2,5),(3,6)]?
29	Write Python statements to insert 50 at index 2 in a list L, and then remove 50.
30	What is the use of the pop() method in a list?
Question Type: Find Output / Long Answers	
31	Describe three different methods to reverse a list in Python. Discuss their advantages and disadvantages.
32	Write a program to read a list of integers and produce a second list containing only prime numbers. Explain the logic used for primality testing and how the final list is built.
33	Start with the list [8,9,10]. Do the following using list functions: (a) Set the second entry (index 1) to 17 (b) Add 4,5 and 6 to the end of the list. (c) Remove the first entry from the list. (d) Sort the list (e) Print the list element twice.

	(f) Insert 25 at index 3
34	Write a Python program that accepts a list L of numbers and finds the sum of all even numbers and the sum of all odd numbers. If L=[1,2,3,4] Output : Sum of even nos:6 Sum of odd nos:4
35	Based on the given sample list write python commands to obtain as asked sample_list = [10, 20, 30, 40, 50, 60, 70, 80, 90, 100] i) Get the first 3 elements in reverse order. ii) Exclude the first and last elements, then return every second element from the result. iii) Reverse the list and return every third element. iv) Get the 4th, 3rd, and 2nd last elements in the same order. v) Get [90, 70, 50, 30] using slicing only.
36	Write a program to input n numbers in a list and display the largest and smallest number.

Answers: LISTS

1. a) [1,2,3,1,2,3]	2. c) pop()
3. 3	4. : d) Both b and c
5. del a[1]	6. sort(reverse=True)
7. b) [1,2,3,4] (<i>b and a refer to the same object</i>)	8. d) Both b and c
9. d) [30]	10. c) [1,9,8,5] (<i>replaces slice of length 3 with two elements</i>)
11. c) Slice assignment	12: a) [5,3,1] (<i>steps backward by 2</i>)
13. c) [None,None,None,None,None]	14. c) remove()
15. a) []	16. c) L = [1, 2, 3]
17. b) append()	18. c) [1, 10, 2, 3]
19	Shallow copy: Copies outer list but shares inner objects → modifying nested data affects both copies. Deep copy: Creates independent copies of all levels. <pre>import copy a = [[1,2],[3,4]] b = copy.copy(a) b[0][0]=99</pre>

	<pre># a changes: [[99,2],[3,4]] a = [[1,2],[3,4]] b = copy.deepcopy(a) b[0][0]=77 # a remains: [[1,2],[3,4]]</pre>
20	Output: [9,2,8,4,7]. Slice lst[:2] selects positions 0,2,4. The replacement updates those indices.
21	<pre>append(x) adds x as a single element to the existing list (e.g., [1,2].append([3,4]) → [1,2,[3,4]]).</pre> <pre>+ concatenates two lists and creates a new list (e.g., [1,2]+[3,4] → [1,2,3,4]).</pre>
22	Output: False. Each multiplication creates a new list object , even if the content is identical.
23	<pre>Output: None [1,2,3,[4,5]] append() modifies the list in place and returns None.</pre>
24	Prints []. Multiplying by a negative number returns an empty list .
25	<pre>lst = [1,2,3,4,5] lst[3:3] = [7,8] # [1,2,3,7,8,4,5]</pre>
26	[1,2,'3','4']. Strings are iterables, so characters are added individually.
27	<pre>nums.sort() sorts in place and returns None. sorted(nums) returns a new sorted list without modifying the original.</pre>
28	pairs = list(zip([1,2,3],[4,5,6]))
29	<pre>L.insert(2, 50) L.remove(50)</pre>
30	<pre>The pop() method removes and returns an element from a list. Without an index, it removes the last element. L = [10, 20, 30] print(L.pop())</pre>
31	Slicing: reversed_list = lst[::-1] (creates a new list). Built-in reversed(): list(reversed(lst)) (returns an iterator, memory efficient). In-place reverse: lst.reverse() (modifies the original list). Advantages: slicing is concise, reversed() avoids extra copies until needed, reverse() is fastest

	for large lists when in-place modification is acceptable.
32	Answer Key Points: • Use a function is_prime(n) to check primality. • Loop through original list and filter primes. <pre>def is_prime(n): if n < 2: return False for i in range(2,int(n**0.5)+1): if n % i == 0: return False return True nums=[] ch='y' while ch=='y' or ch=='Y': n=int(input("Enter element to list")) nums.append(n) ch=input("Do you want to continue....y/n") primes = [x for x in nums if is_prime(x)] print(primes)</pre>
33	<pre>my_list = [8, 9, 10] (a) my_list[1] = 17 # my_list is now [8, 17, 10] (b) _list.extend([4, 5, 6]) # my_list is now [8, 17, 10, 4, 5, 6] (c) my_list.pop(0) # my_list is now [17, 10, 4, 5, 6] (d) my_list.sort() # my_list is now [4, 5, 6, 10, 17] (e) my_list = my_list * 2 # my_list is now [4, 5, 6, 10, 17, 4, 5, 6, 10, 17] (f) my_list.insert(3, 25) # my_list is now [4, 5, 6, 25, 10, 17, 4, 5, 6, 10, 17] # Final list #print(my_list) #[4, 5, 6, 25, 10, 17, 4, 5, 6, 10, 17] 3</pre>
34	<pre>nums=[] ch='y' se=0 so=0 while ch=='y' or ch=='Y':</pre>

	<pre> n=int(input("Enter element to list")) nums.append(n) ch=input("Do you want to continue....y/n") print("list is :",nums) for i in range(len(nums)): if nums[i]%2==0: se=se+nums[i] else: so=so+nums[i] print("sum of even elements are ",se) print("sum of even elements are ",so) </pre>
35	<pre> i) sample_list[2::-1] ii) sample_list[1:-1:2] iii) sample_list[::-1][:3] iv) sample_list[-4:-1] v) sample_list[8:1:-2] </pre>
36	<pre> n = int(input("Enter number of elements: ")) L = [] for i in range(n): num = int(input("Enter a number: ")) L.append(num) print("Largest =", max(L)) print("Smallest =", min(L)) </pre>

Python Tuples: The Essentials

Fundamentals & Creation

Creation & Syntax



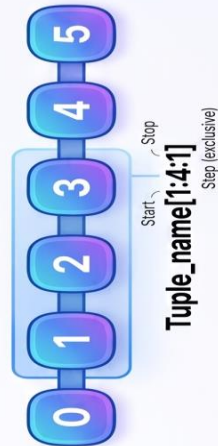
Enclose elements in parentheses; single-value tuples must include a trailing comma to be recognized.

Element Indexing



Every element within a tuple is assigned a specific index for direct access.

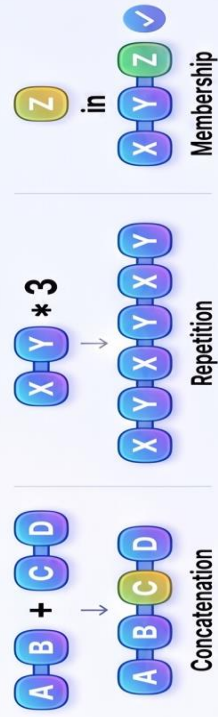
Slicing Logic



Access subparts of a tuple using the specific syntax: `tuple_name[Start:Stop:Step]`.

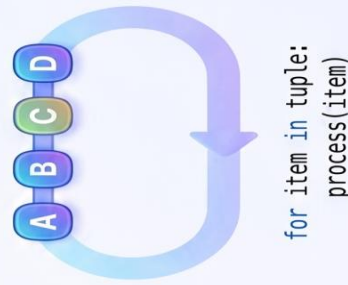
Operations & Built-in Functions

Common Operators



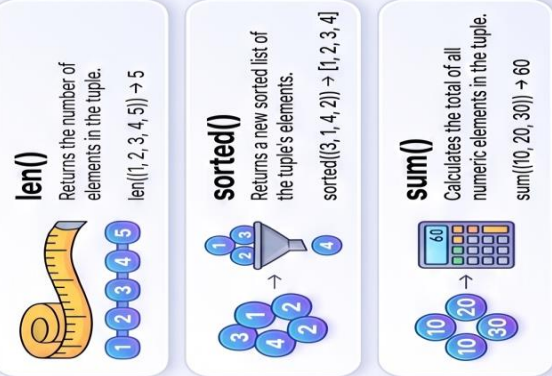
Use `(+)` for concatenation, `(*)` for repetition, and `'in'` to check for membership.

Traversing Tuples



Use loops to access and process every element within the tuple sequentially.

Data Table



Python Tuples: The Essentials

CORE PRINCIPLES

- Tuples are defined using parentheses `()`.
- Single-element tuples require a trailing comma: `(1,)`.
- Tuples are immutable; their elements cannot be changed after creation.
- Elements are accessed using zero-based indexing.
- Slicing extracts sub-sequences with `[start:stop:step]`.
- Tuples can be traversed with loops.
- Common operations include concatenation `(+)`, repetition `(\*)`, and membership testing `(in)`.
- Built-in functions such as `len()`, `sorted()`, and `sum()` are useful.

ACTION STEPS

1. Create tuples with parentheses and comma-separated elements.
2. Use a trailing comma for a single-element tuple.
3. Access individual elements by index.
4. Extract sub-sequences with slicing using start, stop, and step values.
5. Combine tuples with the `+` operator.
6. Repeat tuple elements with the `\*` operator.
7. Check whether an element exists with the `in` operator.
8. Use `len()` for the number of elements.
9. Use `sorted()` to create a sorted list from a tuple.
10. Use `sum()` to total numeric elements.

FORMULAS

CREATE & ACCESS

TUPLE CREATION

```
`mytuple = (element1, element2, ...)`
```

SINGLE-ELEMENT TUPLE

```
`singletuple = (element,)`
```

INDEXING

```
`element = mytuple[index]`
```

OPERATIONS

CONCATENATION

```
`newtuple = tuple1 + tuple2`
```

REPETITION

```
`repeatedtuple = mytuple * n`
```

MEMBERSHIP

```
`ispresent = element in mytuple`
```

SLICING

```
`subtuple = mytuple[start:stop:step]`
```

LENGTH

```
`length = len(mytuple)`
```

SORTING

```
`sortedlist = sorted(mytuple)`
```

SUM

```
`total = sum(mytuple)`
```

KEY TERMS

Tuple: An ordered, immutable sequence of elements in Python.

Immutable: Cannot be changed after creation.

Indexing: Accessing elements by their position, starting from 0.

Slicing: Extracting a portion (sub-sequence) of a tuple.

Concatenation: Joining two or more tuples together.

Repetition: Creating a new tuple by repeating the elements of an existing tuple.

Membership: Checking whether an element exists within a tuple.

REAL-WORLD EXAMPLES

FIXED DATA

Represent coordinates `(x, y)`, RGB color values `(r, g, b)`, or database records.

Tuples make fixed collections clear and dependable.

FUNCTIONS & DICTIONARIES

Functions can return a tuple to provide several results at once.

Because tuples are immutable, they can serve as dictionary keys unlike mutable lists.

REFLECTION

- Why is immutability a key tuple feature, and when is it advantageous?
- How do tuple operations differ from list operations in Python?
- When would you choose a tuple over a list, and vice versa?

	Question Type: MCQ
1	Which method is not valid for tuples? a) count() b) index() c) append() d) len()
2	Tuples are faster than lists because: a) They support more operations b) They are mutable c) They can be changed anytime d) They are immutable and require less overhead
3	What is the output? t = tuple("abc") print(t) a) ['a','b','c'] b) ('a','b','c') c) "abc" d) Error
4	Which of the following tuples is valid? a) (1,2,3,) b) (1,) c) () d) All of the above
5	What is the result of: a = (1,2) b = (3,4) print(a + b * 2) a) (1,2,3,4,3,4) b) (1,2,3,4,4,4) c) (1,2,3,4,3,4,3,4) d) (1,2,3,4,3,4)
	In the following questions, a statement of Assertion(A) is followed by a statement of Reason (R). Make the correct choice as : a. Both A and R are true and R is the correct explanation of A. b. Both A and R are true but R is not the correct explanation of A. c. A is true but R is false. d. A is false but R is true.
6	Assertion (A): Tuple in Python is an ordered and immutable data type. Reasoning (R): Tuples can contain heterogenous data and permit duplicate values as well.
7	Consider the given two statements: Statement 1: t1 = tuple('python') Statement 2: t1[4] = 'z' Assertion (A): The above code will generate an error. Reasoning (R): Tuple is immutable by nature.

8	Assertion(A): Lists and Tuples are similar sequence types of Python, yet they are two different data types. Reason(R): List sequences are immutable and Tuple sequences are mutable.
9	Assertion (A): Tuples can only be created by using parentheses. Reason (R): t = (1, 2, 3) This python statement will create a tuple t.
10	Assertion (A): A tuple can contain other tuples as elements. Reason (R): Tuples support nesting of data structures.
11	What will be the output? <pre>t = (1, [2,3], 4) t[1].append(5) print(t)</pre> a) (1, [2,3], 4) b) (1, [2,3,5], 4) c) (1, [2,3,4,5], 4) d) Error
12	What is the result of: <pre>t = (1,2,3) print(sum(t,10))</pre> a) 6 b) 16 c) 10 d) Error
13	What is the type of t = (5) in Python? a) Tuple b) Integer c) List d) Error
14	What will this print? <pre>t = (1,2,3) x,y,z = t print(z,y,x)</pre> a) 1 2 3 b) 3 2 1 c) (3,2,1) d) Error
15	Choose the correct statement about tuples: a) Tuples consume more memory than lists. b) Tuples can be keys in a dictionary if they contain only immutable elements. c) Tuples are always slower than lists. d) Tuples support item assignment.
	Question Type: Find Output / Short Answer
16	Why can tuples be used as dictionary keys but lists cannot?

17	Why are tuples considered more memory-efficient than lists?
18	What will happen? t = (1,2,3) del t[1]
19	Consider the given tuple and write the output of given statements: T1= (1, 23, 4, 5, "A",["C","D"], (23, 45), 45) 1. print(len(T1)) 2. print(T1.index(45)) 3. print(T1.count(45)) 4. print(T1 [5][0]*2) 5. print(T1 [:5]) 6. print(T1 [5:]) 7. print(T1.index("A")) 8. print(T1 [-1 : -7 : -2]) 9. print(T1 [4] + T1[5]) 10. print(max(T1))
20	Predict the output: t1 = (1,2) t2 = (1,2) print(t1==t2, t1 is t2)
21	What will be the output of the following Python code? tp = (5) tp1 = tp * 2 print(len(tp1))
22	Given tp = (5,3,1,9,0). Which of the following two statements will give the same output? (i) print(tp[:-1]) (ii) print(tp[0:5]) (iii) print(tp[0:4]) (iv) print(tp[-4:])
23	Does the slice operator always produce a new tuple ?
24	When would you prefer tuples over lists ?
25	How can you pair elements from two lists [1,2,3] and [4,5,6] into [(1,4),(2,5),(3,6)]?

	Question Type: Find Output / Long Answers
26	Write a Python program using a tuple to store the marks of five subjects. Then, calculate the total marks, average, and display the subject with the highest marks. Explain the working of your code.
27	Explain the concept of tuple packing and unpacking in Python with examples. Why is unpacking useful in programming?
28	Discuss the immutability of tuples with an example. How can you modify a tuple indirectly if required?
29	Create the following tuples using a for loop containing the squares of the integers 1 through 50.
30	Write a program in Python to find the index of the given number from a tuple using linear search.

Answers: TUPLES

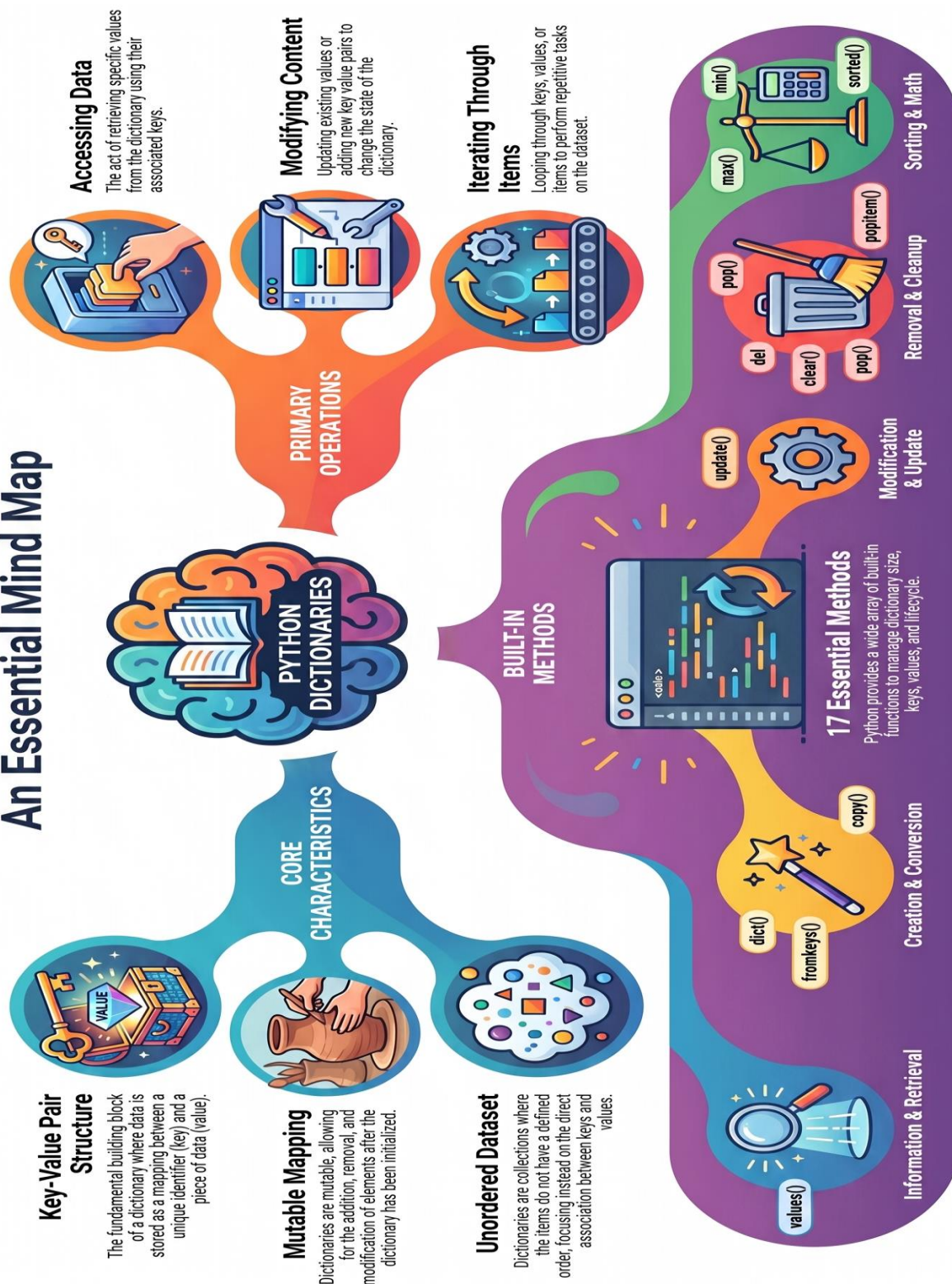
1 c) append() Explanation: append() works on lists, not tuples.	2. d) They are immutable and require less overhead.
3. b) ('a','b','c').	4. d) All of the above.
5. a) (1,2,3,4,3,4) Explanation: b*2 is (3,4,3,4), then concatenated with a.	6. b
7. a	8. c
9. d	10. a
11. : b) (1, [2,3,5], 4) Explanation: Tuples are immutable, but mutable elements inside them can be changed.	12: Answer: b) 16 Explanation: sum(t,10) adds 10 as the start value: 10+1+2+3=16.
13. Answer: b) Integer Explanation: Without a trailing comma, (5) is treated as an integer in parentheses.	14. b) 3 2 1
15. b) Tuples can be keys in a dictionary if they contain only immutable elements.	
16	A: Tuples are immutable and hashable; lists are mutable and not hashable.
17	Because their immutability allows Python to store them in a more compact, optimized way and

	reuse identical tuples internally.
18	A: Raises <code>TypeError</code> — you cannot delete individual tuple elements.
19	1. 8 2. 7 3. 1 4. CC 5. (1, 23, 4, 5, 'A') 6. (['C','D'], (23, 45), 45) 7. 4 8. (45,['C','D'], 5) 9. Error 10. Error
20	True False – values are equal, but they're different objects in memory.
21	0 tp is not a tuple and holds an integer value hence object of type 'int' has no <code>len()</code>
22	(i), (iii) Both will yield (5, 3, 1, 9) . We can use indexes of tuple elements to create tuple slices as per following format : <code>seq = T[start:stop]</code>
23	No, the slice operator does not always produce a new tuple. If the slice operator is applied on a tuple and the result is the same tuple, then it will not produce a new tuple, it will return the same tuple as shown in the example below: <pre>a = (1, 2, 3) print(a[:])</pre> Slicing tuple a using <code>a[:]</code> results in the same tuple. Hence, in this case, slice operator will not create a new tuple. Instead, it will return the original tuple
24	Tuples are preferred over lists in the following cases: 1. When we want to ensure that data is not changed accidentally. Tuples being immutable do not allow any changes in its data. 2. When we want faster access to data that will not change as tuples are faster than lists. 3. When we want to use the data as a key in a dictionary. Tuples can be used as keys in a dictionary, but lists cannot. 4. When we want to use the data as an element of a set. Tuples can be used as elements of a set, but lists cannot.

25	<code>list(zip([1,2,3], [4,5,6]))</code>
26	<pre># Storing marks in a tuple marks = [] print("Enter marks for 5 subjects") for i in range(5): e=int(input("Enter marks: ")) marks.append(e) print(marks) # Total marks total = sum(marks) # Average marks average = total / len(marks) # Highest marks highest = max(marks) print("Total Marks:", total) print("Average Marks:", average) print("Highest Marks:", highest)</pre>
27	Tuple Packing: Assigning multiple values into a single tuple variable. <pre>student = ("Ravi", 16, "Class XI") # Packing</pre> Tuple Unpacking: Extracting values from a tuple into individual variables. <pre>name, age, grade = student # Unpacking print(name, age, grade) # Output: Ravi 16 Class XI</pre> Usefulness: Makes code cleaner and easier to read. Useful when functions return multiple value
28	Answer: Immutability: Once a tuple is created, its elements cannot be changed. <pre>tup = (1, 2, 3) # tup[1] = 99 # ✗ This raises: TypeError</pre> Indirect Modification Methods: Convert the tuple into a list, modify it, then convert it back.

	Concatenate with another tuple to form a new one. Example: <pre>tup = (1, 2, 3) lst = list(tup) # Convert to list lst[1] = 99 # Modify tup = tuple(lst) # Convert back to tuple print(tup) # Output: (1, 99, 3)</pre>
29	<pre>l=[] for i in range(1,51): l.append(i*2) t=tuple(l) print(t)</pre>
30	<pre>t=(10,2,30,4,8,5,45) print("Tuple is :",t) c='y' while c=='y'or c=='Y': n=int(input("Enter the number to be searched in the tuple: ")) found=False for i in t: if i==n: found=True print("Element found at index: ",t.index(i)) if found==False: print("Number is not found") c=input("do you want to continue....y/n")</pre>

Mastering Python Dictionaries: An Essential Mind Map



Mastering Python Dictionaries: An Essential Mind Map

CORE PRINCIPLES

Key-Value Pair Structure: The basic unit maps a unique identifier (key) to a piece of data (value).

Mutable Mapping: Dictionaries can be modified after initialization: add, remove, or change elements.

Dataset Organization: Items are accessed through their key-value association; modern dictionaries preserve insertion order.

ACTION STEPS

1. Understand the key-value pair structure as the dictionary foundation.
2. Recognize dictionaries as mutable, enabling dynamic changes.
3. Access data using its associated key.
4. Modify existing values or add new key-value pairs.
5. Iterate through keys, values, or items for repeated tasks.
6. Use built-in methods for creation, conversion, modification, removal, and sorting.

PRO TIPS

- Keys must be immutable, such as strings, numbers, or tuples.
- Use descriptive keys to improve code readability.
- Standard dictionaries maintain insertion order in Python 3.7 and later.
- Avoid ``KeyError`` with ``.get()`` when a key may not exist.

PITFALLS

- Using mutable objects such as lists as keys.
- Overwriting existing keys unintentionally.
- Assuming dictionaries are ordered in older Python versions.
- Failing to handle potential ``KeyError`` exceptions.

KEY TERMS

Dictionary: A mutable collection of key-value pairs; insertion-ordered in modern Python.

Key: A unique, immutable identifier used to access a value.

Value: The data associated with a key.

Mutable: An object whose state can be modified after creation.

Immutable: An object whose state cannot be modified after creation.

REAL-WORLD EXAMPLES

PROFILES & COUNTS

Store user profiles: keys such as 'username', 'email', and 'age' map to their values. Count word frequencies: words become keys and their counts become values.

CONFIGURATION

Represent configuration settings with setting names such as 'database_url' as keys. Map each setting to its corresponding value for simple lookup.

REFLECTION

- How can dictionary operations improve the efficiency of Python code?
- When is dictionary mutability a major advantage or disadvantage?
- How does key-value mapping apply to other data structures or real-world systems?

	Question Type: MCQ
1	What will be the output of the following Python code snippet? <pre>d = {"Apple":40, "Orange":45} print("Apple " in d)</pre> a) True b) False c) None d) Error
2	What will be the output of the following Python code snippet? <pre>d1 = {" Apple ":40, " Orange ":45} d2 = {" Apple ":466, " Orange ":45} print(d1 == d2)</pre> a) True b) False c) None d) Error
3	What will be the output of the following Python code snippet? <pre>d1 = {" Apple ":40, " Orange ":45} d2 = {" Apple ":466, " Orange ":45} print(d1 > d2)</pre> a) True b) False c) Error d) None

4	Suppose d = {"Delhi":45, "Pune":32}, to delete the entry for "Delhi" what command do we use? a) d.delete("Delhi":40) b) d.delete("Delhi") c) del d["Delhi"] d) del d("Delhi":40)
5	Suppose d = {"Delhi":45, "Pune":32}, To obtain the number of entries in dictionary which command do we use? a) len(d) b) d.size() c) size(d) d) d.len()
6	Suppose d = {"Mayank":40, "Abhimanyu":45}, what happens when we try to retrieve a value using the expression d["Abhi"]? a) Since "Abhi" is not a value in the set, Python raises a KeyError exception b) It is executed fine and no exception is raised, and it returns None c) Since "Abhi" is not a key in the set, Python raises a KeyError exception d) Since "Abhi" is not a key in the set, Python raises a syntax error
7	Which of these about a dictionary is false? a) The values of a dictionary can be accessed using keys b) The keys of a dictionary can be accessed using values c) Dictionaries aren't ordered d) Dictionaries are mutable
8	Which of the following is not a declaration of the dictionary? a) {1: 'A', 2: 'B'} b) dict([[1,"A"],[2,"B"]]) c) {1,"A",2"B"} d) { }
9	What will be the output of the following Python code snippet? <pre>a={1:"A",2:"B",3:"C"} for i,j in a.items(): print(i,j,end=" ")</pre> a) 1 A 2 B 3 C b) 1 2 3 c) A B C d) 1:"A" 2:"B" 3:"C"
10	What will be the output of the following Python code snippet? <pre>a={1:"A",2:"B",3:"C"} print(a.get(1,4))</pre> a) 1 b) A c) 4 d) Invalid syntax for get method

11	What will be the output of the following Python code snippet? <pre>a={1:"A",2:"B",3:"C"} print(a.setdefault(3))</pre> a) {1: 'A', 2: 'B', 3: 'C'} b) C c) {1: 3, 2: 3, 3: 3} d) No method called setdefault() exists for dictionary
12	What will be the output of the following Python code? <pre>a={1:"A",2:"B",3:"C"} b={4:"D",5:"E"} a.update(b) print(a)</pre> a) {1: 'A', 2: 'B', 3: 'C'} b) Method update() doesn't exist for dictionaries c) {1: 'A', 2: 'B', 3: 'C', 4: 'D', 5: 'E'} d) {4: 'D', 5: 'E'}
In the following questions, a statement of Assertion(A) is followed by a statement of Reason (R). Make the correct choice as : a. Both A and R are true and R is the correct explanation of A. b. Both A and R are true but R is not the correct explanation of A. c. A is true but R is false. d. A is false but R is true.	
13	Assertion (A): You can use a list as a key in a Python dictionary. Reason (R): Dictionary keys must be immutable data types, and lists are mutable.
14	Assertion (A): The clear() method removes all elements from a dictionary. Reason (R): clear() is used to delete an entire dictionary, not just remove its contents.
15	Assertion (A): You can add a new key-value pair to a dictionary using assignment. Reason (R): If the key already exists, the new assignment updates the existing key's value; otherwise, a new entry is created.
	Question Type: Find Output / Short Answer
16	Why dictionaries are called mutable types?
17	How are individual elements of dictionaries accessed?
18	How is indexing of a dictionary different from that of a list or a string?

19	What do you understand by ordered collection and unordered collection ? Give examples.
20	What will be the output of following Python code? <pre> d1 = {"a":10,"b":2,"c":3} str1="" for i in d1: str1 = str1 + str(d1[i]) + " " str2 = str1[:-1] print(str2[::-1]) </pre>
21	Can you change the order of dictionary's contents, i.e., can you sort the contents of a dictionary?
22	Dictionary is a mutable type, which means you can modify its contents? What all is modifiable in a dictionary? Can you modify the keys of a dictionary?
23	How is del D and del D[<key>] different from one another if D is a dictionary?
24	What does fromkeys() method do?
25	How is pop() different from popitem()?
	Question Type: Find Output / Long Answers
26	Write a program to convert a number entered by the user into its corresponding number in words. For example, if the input is 876 then the output should be 'Eight Seven Six'. (Hint. use dictionary for keys 0-9 and their values as equivalent words.)
27	Create a dictionary whose keys are month names and whose values are the number of days in the corresponding months. (a) Ask the user to enter a month name and use the dictionary to tell how many days are in the month. (b) Print out all of the keys in alphabetical order. (c) Print out all of the months with 31 days. (d) Print out the (key-value) pairs sorted by the number of days in each month.
28	Write a program that checks if two same values in a dictionary have different keys. That is, for dictionary D1 = { 'a' : 10, 'b': 20, 'c' : 10}, the program should print 2 keys have same values and for dictionary D2 = { 'a' : 10, 'b' : 20, 'c' : 30} , the program should print No keys

	have same values.
29	How do you add key:value pairs to an existing dictionary?
30	Write a Python program to remove a specific key from a dictionary, retrieve all key-value pairs, and check whether a given key exists.

Answers: DICTIONARY

1. a) True	2. b) False
3. c) Python does not support dictionary comparison operators	4. c)
5. a)	6. c)
7. b)	8. c)
9. a)	10. b)
11. b)	12. c)
13. (c) A is false and R is true	
14(c) A is true, but R is false. (The clear() method removes items but doesn't delete the dictionary itself, it returns an empty dictionary.)explanation of A.	
15. (a) Both A and R are true, and R is the correct explanation of A.	
16	Dictionaries can be changed by adding new key-value pairs and by deleting or changing the existing ones. Hence they are called as mutable types. For example: <pre>d = {"a" : 1 , "b" : 2} d["c"] = 3 d["b"] = 4 del d["a"] print(d)</pre> <i>Output</i> <pre>{'b': 4, 'c': 3}</pre>
17	Individual elements of a dictionary can be accessed by using their corresponding keys as per the syntax shown below: <dictionary-name> [<key>]

	For example: <pre>d = {'a': 1, 'b': 2, 'c': 3} print(d['a'])</pre>		
18	In lists or strings, the elements are accessed through their index where as in dictionaries, the elements are accessed through the keys defined in the key:value pairs. Moreover, lists and strings are ordered set of elements but dictionaries are unordered set of elements so its elements cannot be accessed as per specific order.		
19	Ordered Collection is the one in which the position of each element is fixed. Example: List, strings, Tuples Unordered Collection is the one in which position of each element is not fixed i.e., the order of all the elements are not maintained. Example: Sets, Dictionaries		
20	3 2 01		
21	No, the contents of a dictionary cannot be sorted in place like that of a list. However, we can indirectly sort the keys and values of a dictionary by using sorted() function: • sorted(dictionary.keys()) • sorted(dictionary.values()) • sorted(dictionary) • sorted(dictionary.items())		
22	Yes, we can modify the contents of a dictionary. Values of key-value pairs are modifiable in dictionary. New key-value pairs can also be added to an existing dictionary and existing key-value pairs can be removed. However, the keys of the dictionary cannot be changed. Instead we can add a new key : value pair with the desired key and delete the previous one.		
23	del D is used to delete the whole dictionary where as del D[<key>] is used to delete only the key:value pair with the given key.		
24	The fromkeys() method is used to create a new dictionary from a sequence containing all the keys and a common value, which will be assigned to all the keys as per syntax shown below: <pre>dict.fromkeys(<keys sequence>, [<value>])</pre>		
25	The differences between pop() and popitem() are mentioned below: <table> <tr> <td>pop()</td><td>popitem()</td></tr> </table>	pop()	popitem()
pop()	popitem()		

	pop() removes the item with the specified key name. With pop(), we can specify a return value or a message if the given key is not found in the dictionary.	popitem() removes the last inserted item from the dictionary. With popitem(), we cannot specify any such message/return value while deleting from an empty dictionary. It will raise an error in this case.
26	<pre> num = int(input("Enter a number: ")) d = {0 : "Zero" , 1 : "One" , 2 : "Two" , 3 : "Three" , 4 : "Four" , 5 : "Five" , 6 : "Six" , 7 : "Seven" , 8 : "Eight" , 9 : "Nine"} digit = 0 str = "" while num > 0: digit = num % 10 num = num // 10 str = d[digit] + " " + str print(str) </pre>	
27	<pre> days_in_months = { "january":31, "february":28, "march":31, "april":30, "may":31, "june":30, "july":31, "august":31, "september":30, "october":31, "november":30, "december":31 } </pre>	

```

m = input("Enter name of month: ")
if m not in days_in_months:
    print("Please enter the correct month")
else:
    print("There are", days_in_months[m], "days in", m)
print("Months in alphabetical order are:", sorted(days_in_months))
print("Months with 31 days:", end=" ")
for i in days_in_months:
    if days_in_months[i] == 31:
        print(i, end=" ")
day_month_lst = []
for i in days_in_months:
    day_month_lst.append([days_in_months[i], i])
day_month_lst.sort()
month_day_lst = []
for i in day_month_lst:
    month_day_lst.append([i[1], i[0]])
sorted_days_in_months = dict(month_day_lst)
print()
print("Months sorted by days:", sorted_days_in_months)

```

```

28 D1 = eval(input("Enter a dictionary D1: "))
    print("D1 =", D1)

    val = tuple(D1.values())
    seen = []
    flag = True

    for i in val:
        if i not in seen:
            seen.append(i)
            count = val.count(i)

```

	<pre> if count > 1: print(count, "keys have same value of", i) flag = False if flag: print("No keys have same values") </pre>
29	There are three ways by which new key:value pairs can be added to an existing dictionary: 1. By using assignment as per the following syntax: <pre><dictionary>[<key>] = <value></pre> 2. By using update() method: update() method merges key:value pairs from new dictionary into the original dictionary adding or replacing as needed. The syntax to use this method is: <pre><dictionary>.update(<other-dictionary>)</pre> 3. Using setdefault() method: It inserts a new key:value pair only if the key doesn't already exist. If the key already exists, it returns the current value of the key. The syntax to use this method is: <pre><dictionary>.setdefault(<key>,<value>)</pre>
30	<pre> car = {"brand": "Toyota", "model": "Camry", "year": 2022, "color": "blue"} # Remove a key car.pop("color") print(car) # Get all key-value pairs print(car.items()) # Check key existence print("'brand' exists:", "brand" in car) print("'color' exists:", "color" in car) </pre>

CLASS XI Computer Science (083)
UNIT 3 : SOCIETY, LAW AND ETHICS

CLASS XI – COMPUTER SCIENCE (083)
UNIT 3: SOCIETY, LAW AND ETHICS – SUMMARY

Key Concepts

- * Cybercrime and its types
- * Malware and Security
- * Digital Ethics and Etiquette
- * Intellectual Property Rights (IPR)
- * Open-source Software Licenses
- * E-waste Management
- * IT Act, 2000 and Digital Laws

Quick Revision (MCQs)

1. Which is NOT a cybercrime? → (C) Online learning
2. Open-source license → (A) Creative Commons
3. Malware that demands money → (B) Ransomware
4. Digital footprints → (A) Online activities left behind
5. NOT a social media etiquette → (B) Use offensive language
6. GPL is an example of → (B) Open-source license
7. Law in India for cybercrime & e-commerce → (A) IT Act
8. E-waste should be → (C) Properly recycled
9. Example of e-waste → (A) Old mobile phones
10. Violation of IPR → (B) Downloading pirated movies

Assertion – Reason (Key Points)

11. Digital footprints impact future opportunities.
→ Both A and R are true and R explains A.
12. Plagiarism is NOT an ethical practice.
→ A is false, but R (shows respect) is true.
13. Trojan horse appears useful but harms secretly.
→ A is true, R is false (Trojans are not self-replicating).
14. IT Act, 2000 gives legal recognition to electronic records & digital signatures.
→ A is true, R is false.
15. Safe browsing protects users from cybercrimes.
→ Both A and R are true and R explains A.

Important Definitions

- Cybercrime: Illegal activities using computers or the internet.
- Malware: Malicious software that harms systems (e.g., virus, ransomware, trojan).
- Ransomware: Malware that locks data and demands payment.
- Digital Footprint: Information about a person left behind through online activities.
- IPR (Intellectual Property Rights): Rights given to creators for their work/ideas.
- E-waste: Discarded electronic devices and components.
- IT Act, 2000: Indian law that deals with cybercrimes, e-commerce and digital signatures.

Case Based & Short Answer – Key Takeaways

- | | |
|--|---|
| 16. Pirated movie download & ransomware attack | i) Violations: Piracy, Infringement of IPR
ii) Malware: Ransomware
iii) Safety: Use legal websites, Keep antivirus updated, Avoid suspicious downloads. |
| 17. Copying article without citing source | i) Violation: Plagiarism (IPR violation)
ii) License: (B) Creative Commons
iii) Correct practice: Give proper citation / reference. |
| 18. Fake social media account of friend | i) Cybercrime: Impersonation / Cyber bullying
ii) Rights violated: Right to privacy, Right to dignity.
iii) Harmful effects: Mental stress, Damage to reputation. |
| 19. Fraud email lottery | a) Type: Phishing
b) Identify: Unreal offers, unknown sender, suspicious links
c) Precaution: Don't share personal / bank details, verify before replying. |
| 20. Dumping e-waste in river | a) Issue: Environmental pollution
b) Disposal: Recycle through authorized e-waste collectors
c) Effect: Water pollution, Health hazards, Harm to aquatic life. |

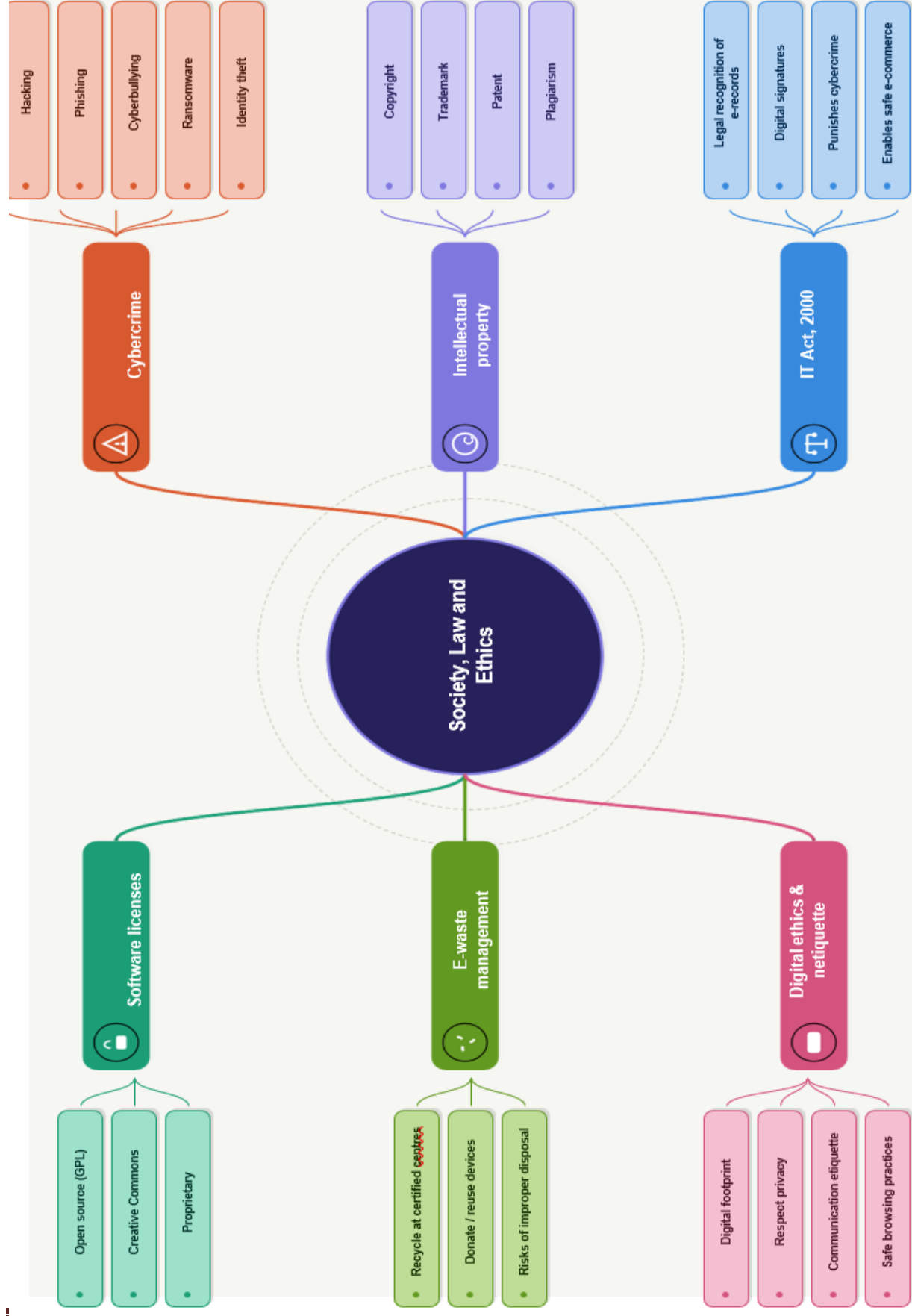
More Short Answers (21-29)

21. Hacking: Unauthorized access. Phishing: Fake messages to steal info.
22. Important because it reduces pollution, saves resources & protects health.
23. Use strong passwords, don't share personal info, enable 2FA.
24. Provides legal framework, punishes cybercrimes, protects users.
25. Legal action, fines, imprisonment, loss of reputation.
26. Examples: Making fun online, Spreading rumours/jokes.
27. Awareness programs, strong passwords, reporting cybercrimes, following safe internet rules.
28. Helps in respectful communication, avoids conflicts, builds a positive digital environment.
29. Virus: Attaches to files, needs user action to spread.
Trojan: Disguised as useful program, does not self-replicate.

Remember!

- * Think before you click.
- * Respect others online.
- * Protect your data.
- * Use original software.
- * Keep the environment clean – manage e-waste properly.
- * Follow the law –
Stay safe, stay ethical!





	Multiple Choice Questions
1	Which of the following is NOT a cybercrime? (A) Hacking (B) Cyber bullying (C) Online learning (D) Phishing
2	Which of these is an open-source software license? (A) Creative Commons (B) Trademark (C) Copyright (D) Proprietary.
3	Which malware is designed to demand money from users? (A) Trojan (B) Ransomware (C) Virus (D) Adware
4	Digital footprints are: (A) Online activities left behind (B) Computer hardware (C) Cyber bullying (D) Net etiquette
5	Which of the following is NOT a social media etiquette? (A) Respect privacy (B) Use offensive language (C) Be polite (D) Avoid spamming
6	GPL is an example of: (A) Proprietary license (B) Open-source license (C) Copyright violation (D) Trademark
7	Which law in India deals with cybercrime and e-commerce? (A) IT Act (B) Copyright Act (C) Trademark Act (D) Companies Act
8	E-waste should be: (A) Burnt openly (B) Thrown in garbage (C) Properly recycled (D) Stored forever
9	Which of the following is an example of e-waste? (A) Old mobile phones (B) Fresh vegetables (C) Wooden furniture (D) Paper books
10	Which among the following is considered a violation of Intellectual Property Rights? (A) Writing your own essay (B) Downloading pirated movies (C) Using GPL software legally (D) Buying an original book
11	Assertion (A): Digital footprints can impact an individual's future opportunities. Reason (R): Employers often check social media activities of candidates before hiring. A) Both A and R are true, and R is the correct explanation of A B) Both A and R are true, but R is not the correct explanation of A C) A is true, but R is false D) A is false, but R is true

12	Assertion (A): Plagiarism is considered an ethical practice in academics. Reason (R): It shows respect towards the original creator's work. A) Both A and R are true, and R is the correct explanation of A B) Both A and R are true, but R is not the correct explanation of A C) A is true, but R is false D) A is false, but R is true
13	Assertion (A): A Trojan horse appears like a useful program but harms the system secretly. Reason (R): Trojans are self-replicating viruses that spread automatically. A) Both A and R are true, and R is the correct explanation of A B) Both A and R are true, but R is not the correct explanation of A C) A is true, but R is false D) A is false, but R is true
14	Assertion (A): The IT Act, 2000 provides legal recognition to electronic records and digital signatures in India. Reason (R): The Act was introduced to promote traditional paper-based communication. A) Both A and R are true, and R is the correct explanation of A B) Both A and R are true, but R is not the correct explanation of A C) A is true, but R is false D) A is false, but R is true
15	Assertion (A): Safe browsing practices help protect users from cybercrimes. Reason (R): Avoiding suspicious links and using strong passwords reduces online risks. A) Both A and R are true, and R is the correct explanation of A B) Both A and R are true, but R is not the correct explanation of A C) A is true, but R is false D) A is false, but R is true
	Short Answer Type Questions
16	Rohan downloads a movie from a pirated website and shares it on social media. Later, his computer gets attacked by ransomware. i) Identify two violations of digital ethics in this case. ii) Which type of malware is mentioned here? (A) Trojan (B) Ransomware (C) Virus (D) Adware

	iii). Suggest two safety measures Rohan could have taken.
17	A student copies large sections of an online article into a project report without citing the source. i). What violation of intellectual property rights has been committed? ii). Which license encourages free use with attribution? (A) GPL (B) Creative Commons (C) Trademark (D) Patent iii). Suggest a correct practice in this situation.
18	A group of students created a fake social media account in the name of their Friend. They posted insulting comments and shared manipulated photos using that account. i) Identify the cyber crime in this case. ii) Which rights of the teacher were violated? iii) Mention two harmful effects of such actions on victims.
19	Riya received an email claiming that she had won a lottery of ₹10 lakh and was asked to share her bank details to claim the prize. (a) Which type of cyber crime is this? (b) Suggest one way Riya can identify such fraud emails. (c) Write one precaution she should take to stay safe from such scams
20	An IT company disposed of old computers by dumping them in a nearby river, leading to water pollution and health hazards in the local area. (a) Identify the issue in this case. (b) Suggest one proper method of disposal. (c) Mention one harmful effect of such actions.
21	Differentiate between hacking and phishing with examples.
22	Why is e-waste management important for the environment?
23	Suggest two ways to protect identity while browsing online.
24	How does the IT Act help in reducing cyber crimes?
25	What are the consequences of violating copyright laws?
26	Give two examples of cyber bullying.
27	Suggest two ways schools can promote cyber safety among students.
28	What is the role of communication etiquettes in digital society?
29	Differentiate between virus and Trojan.

30	What is phishing? Give one real-life example
	Long Answer Type Questions
31	Differentiate between Copyright and Trademark with examples.
32	Explain the importance of Net Etiquettes in online communication
33	Discuss the role of the IT Act in ensuring cyber safety
34	What is ransomware? How can it be prevented?
35	Discuss the impact of digital footprints on society.
36	Write short notes on: (a) Open Source Software, (b) Creative Commons License
37	Explain the social issues of technology usage with respect to gender and disability
38	Rohit received an email from an unknown sender claiming he had won a lottery of ₹10,00,000. The email asked him to share his bank account details and pay a small fee to claim the prize. (a) Identify the type of cybercrime in this case. (b) Which security practice could have helped Rohit identify this fraud? (c) Mention one possible consequence if Rohit shares his bank details. (d) Suggest one step he should immediately take after receiving such an email. (e) Name the law in India that deals with such fraudulent activities.
39	A company replaced all its old smartphones and computers with new ones. Instead of proper disposal, the devices were thrown in a nearby dumping ground. Soon, people living in that area complained of foul smell and water contamination. (a) What is the main issue highlighted in this case? (b) Name two harmful effects of improper disposal of e-waste. (c) Suggest one environmentally friendly method of e-waste management. (d) Mention one role individuals can play in reducing e-waste. (e) What broader environmental concern does this case represent?
40	An online influencer uploaded a song on YouTube without taking permission from the original singer. The video got millions of views, and the influencer earned a lot of money through advertisements. (a) Identify the violation of law in this case. (b) Which Intellectual Property Right is being infringed? (c) Suggest one ethical alternative the influencer could have used.

	(d) Which value is being ignored by the influencer here?
	(e) Name one license type that allows free sharing and usage of work with conditions.

ANSWER: SOCIETY LAW AND ETHICS

1	(C) Online learning – because it is a legitimate online activity, not a crime.
2	(A) Creative Commons – it allows free use, sharing, and modification with conditions
3	(B) Ransomware – it locks/encrypts data and demands payment for release
4	(A) Online activities left behind – every click, search, and post creates footprints.
5	(B) Use offensive language – it goes against proper net etiquette.
6	(B) Open-source license – General Public License (GPL) allows modification and redistribution.
7	(A) IT Act – the Information Technology Act of 2000.
8	(C) Properly recycled – to avoid pollution and health hazards.
9	(A) Old mobile phones – because electronic gadgets, when discarded, become e-waste.
10	(B) Downloading pirated movies – it violates copyright laws.
11	(A) Both A and R are true, and R is the correct explanation of A.
12	(D) A is false, but R is true
13	(C) A is true but R is false.
14	(C) A is true but R is false.
15	(A) Both A and R are true, and R is the correct explanation of A
16	i: Copyright infringement and illegal file sharing. ii Answer: (B) Ransomware – because it locks the system and demands ransom. iii: Answer: Avoid pirated websites and install reliable antivirus software.
17	i: Answer: Plagiarism – using someone else’s work without proper credit. ii) (B) Creative Commons – allows reuse with acknowledgment. iii): Cite the original source and use quotations or paraphrasing properly.
18	i)Cyber bullying, identity theft, and defamation. ii) Right to privacy and reputation. iii) Stress, loss of reputation
19	(i)Phishing/fraud email scam. (ii)) Suspicious email address, spelling errors, unrealistic offers.

	(iii) Never share bank details online; verify sender before responding.
20	(i) Improper e-waste management. (ii) Recycling through authorized e-waste centers. (iii) Toxic chemicals enter soil and water, causing pollution and diseases.
21	Hacking = unauthorized access to systems (e.g., breaking into a server). Phishing = tricking users with fake emails/websites to steal data (e.g., fake bank emails).
22	Improper disposal releases toxic chemicals, pollutes soil and water, and harms human health
23	Use strong, unique passwords; avoid sharing personal details on unknown platforms.
24	It provides legal recognition of online transactions, defines punishments for cyber crimes, and ensures cyber safety.
25	Legal penalties, fines, lawsuits, loss of reputation, and removal of infringing content.
26	Sending abusive messages, spreading false rumors online.
27	Conduct awareness programs; implement strict rules for safe internet usage.
28	Ensures polite, clear, and respectful interaction; reduces misunderstandings online.
29	Virus attaches to files and spreads; Trojan appears harmless but secretly harms systems.
30	Fraudulent attempts to steal information via fake emails/websites; e.g., “bank verification email” scams.
31	Copyright: It protects original literary, artistic, musical, or software works. It prevents unauthorized copying or reproduction. Example: A novel written by an author or software code is protected under copyright. Trademark: It protects unique symbols, names, words, or logos that represent a brand or company. Example: The Apple logo or Nike’s “tick mark” symbol. Difference: Copyright focuses on content creation, while Trademark safeguards brand identity.
32	Net etiquettes are rules of polite behavior while using the internet. They ensure respectful and clear communication, reducing conflicts in digital platforms. Practicing good etiquettes builds a positive digital reputation and safe environment. Example: Avoiding ALL CAPS in chats (as it indicates shouting), respecting others’ privacy, not spamming groups.

33	The Information Technology Act, 2000 gives legal recognition to electronic records, digital signatures, and online transactions. It defines and punishes cyber crimes like hacking, identity theft, fraud, and cyber terrorism. It helps ensure safe e-commerce, online banking, and data security in India. Example: If a person is caught phishing through fake bank websites, IT Act provides penalties and imprisonment.
34	Ransomware is malicious software that locks a user's files by encrypting them and demands payment (ransom) for decryption. It spreads via suspicious email attachments, pirated software, or infected links. Prevention: Maintain backups of important data. Use updated antivirus and firewalls. Avoid clicking on suspicious links/emails. Example: The WannaCry ransomware attack (2017) affected thousands of computers worldwide.
35	Positive Impact: Helps in building professional identity, makes communication transparent, and assists in law enforcement. Negative Impact: Can be misused for cyber stalking, targeted ads, and rejection in jobs due to past irresponsible posts. Example: A student's old offensive tweets may harm chances of getting selected in a job interview.
36	Open Source Software (OSS): Software whose source code is freely available for anyone to use, modify, and distribute. Example: Linux operating system, Mozilla Firefox browser. Creative Commons (CC) License: A license that allows creators to share their work with specific conditions such as attribution, non-commercial use, or share-alike. Example: Photographs on Flickr with CC license that others can reuse with credit.
37	Fewer career opportunities in tech fields. Disability Issues: People with physical or cognitive disabilities may struggle due to lack of accessible software, hardware, or web designs. Solutions:

	Promote digital literacy equally among genders. Design inclusive technologies (screen readers, speech-to-text, accessibility-friendly websites). Example: Developing apps with voice assistance helps visually impaired users.
38	(a) Phishing and fraud email scam. (b) Being cautious of unsolicited emails and verifying sender details. (c) Hackers could steal money or commit identity theft. (d) Delete the email and report it as spam/phishing. (e) Information Technology (IT) Act, 2000.
39	(a) Improper e-waste management. (b) Release of toxic chemicals; Soil and water pollution. (c) Recycling through certified e-waste centers. (d) Donate or reuse old gadgets. (e) Sustainability and environmental protection.
40	(a) Copyright infringement. (b) Copyright of the singer/music creator. (c) Take permission or purchase a license to use the song legally. (d) Honesty and respect for others' creative work. (e) Creative Commons (CC) License
