



**M3-R5.1**

**Python Programming**

**Short Notes**

**Quick Revision**

# **Detailed Syllabus O Level Python(M3-R5.1)**

## **1. Introduction to Programming**

The basic Model of computation, algorithms, flowcharts, Programming Languages, compilation, testing & debugging and documentation.

## **2. Algorithms and Flowcharts to Solve Problems**

Flow Chart Symbols, Basic algorithms/flowcharts for sequential processing, decision based processing and iterative processing. Some examples like: Exchanging values of two variables, summation of a set of numbers, Decimal Base to Binary Base conversion, reversing digits of an integer, GCD (Greatest Common Divisor) of two numbers, Test whether a number is prime, factorial computation, Fibonacci sequence, evaluate ‘sin x’ as sum of a series, Reverse order of elements of an array, Find largest number in an array, Print elements of upper triangular matrix, etc.

## **3. Introduction to Python**

Python Introduction, Technical Strength of Python, Introduction to Python Interpreter and program execution, Using Comments, Literals, Constants, Python’s Built-in Data types, Numbers (Integers, Floats, Complex Numbers, Real, Sets), Strings (Slicing, Indexing, Concatenation, other operations on Strings), Accepting input from Console, printing statements, Simple ‘Python’ programs.

## **4. Operators, Expressions and Python Statements**

Assignment statement, expressions, Arithmetic, Relational, Logical, Bitwise operators and their precedence, Conditional statements: if, if-else, if-elif-else; simple programs, Notion of iterative computation and control flow –range function, While Statement, For loop, break statement, Continue Statement, Pass statement, else, assert.

## **5. Sequence Data Types**

Lists, tuples and dictionary, (Slicing, Indexing, Concatenation, other operations on Sequence datatype), concept of mutability, Examples to include finding the maximum, minimum, mean; linear search on list/tuple of numbers, and counting the frequency of elements in a list using a dictionary.

## **6. Functions**

Top-down approach of problem solving, Modular programming and functions, Function parameters, Local variables, the Return statement, Doc Strings, global statement, Default argument values, keyword arguments, VarArgs parameters.

Library function-input(), eval(), print(), String Functions: count(), find(), rfind(), capitalize(), title(), lower(), upper(), swapcase(), islower(), isupper(), istitle(), replace(), strip(), lstrip(), rstrip(), split(), partition(), join(), isspace(), isalpha(), isdigit(), isalnum(), startswith(), endswith(), encode(), decode(), String: Slicing, Membership, PatternMatching, Numeric Functions: eval(), max(), min(), pow(), round(), int(), random(), ceil(), floor(), sqrt(), Date & Time Functions, Recursion.

## **7. File Processing**

Concept of Files, File opening in various modes and closing of a file, Reading from a file, Writing onto a file, File functions-open (), close (), read (), readline(), readlines(), write(), writelines(), tell(), seek(), Command Line arguments.

## **8. Scope and Modules**

Scope of objects and Names, LEGB Rule Module Basics, Module Files as Namespaces, Import Model, Reloading Modules.

## **9. NumPy Basics**

Introduction to NumPy ndarray, datatypes, array attributes, array creation routines, Array From Existing Data, Array From Numerical Ranges, Indexing & Slicing.

प्रिय छात्र/साथियों,

■ O Level की सफलता अब आपके हाथोंमें है!

आपके पास जो यह फ्री ई-बुक है, वह केवल एक किताब नहीं बल्कि कलाखों छात्रों के अनुभव, माग निचोड़ है।

इस पसुतक में हमने NIELIT O Level Python Programming (M3-R5.1) के पूरे सिलेबस को बेहद आसान भाषा में **Short Notes, One Liners**, और **Chapterwise MCQs** केरपूर्में प्रस्तुत किया है, ताकि आपकी तैयारी तेज़, सटीक और परिणामदायक हो।

हम, **Satyam Sahu** द्वारा संचालित **NewIdeasYT** यूट्यूब चैनल के माध्यम से पिछले 6 वर्षों से O Level की गारंटी ड तैयारी करवा रहे हैं।

अब तक लाखों छात्र इस कोस 'को सफलत मूव' संगलक्ति रूप है।

लेकिन ध्यान दें —

यह तो बस शुरुआत है!

अगर आप चाहते हैं पूरी गहराई से समझना, लाइव क्लासेस, प्रैलॉकक्स असाइनम, डैट्साट सेशन्स और मॉक टेस्ट — तो जुड़िए हमारे **Premium Paid Batch** से जहाँ मिलेगा:

**Special:** NIELIT द्वारा बनाये गये लगभग 400+ खतरनाक MCQs का कलेक्शन जो न तो इन्टरनेट पर है, न किसी बुक या अन्य प्लेटफोर्म पर

- |   |   |
|---|---|
| <ul style="list-style-type: none"><li>❖ Chapterwise PDF Notes &amp; Classes</li><li>❖ 2000+ MCQs PDF &amp; Classes</li><li>❖ Old Solved Paper PDFs &amp; Videos</li><li>❖ Original Practical Old Paper PDF &amp; Videos</li></ul> | <ul style="list-style-type: none"><li>❖ Viva &amp; Practical Concept Classes</li><li>❖ Premium Full Length Test (Same Exam Interface)</li><li>❖ Bilingual Content</li></ul> |
|---|---|

**Batch Registration:** <https://examjila.com/courses/m3r5>

अब बारी आपकी है— सफलता की ओर पहला कदम उठाइए!

आपकी सफलता हमारी जिम्मेदार है।

आपका माग

Satyam Sahu  
(Founder & Educator – NewIdeasYT)

# Index

Chapter	Chapter Name	Page No.
1.	Introduction to Programming	7
2.	Algorithms and Flowcharts to Solve Problems	17
3.	Introduction to Python	28
4.	Operators, Expressions and Python Statements	35
5.	Sequence Data Types	44
6.	Functions	51
7.	File Processing	62
8.	Scope and Modules	70
9.	NumPy Basics	78



## Chapter 1

# Introduction to Programming (प्रोग्रामिंग का परिचय)

### What is Programming?

Programming is the process of designing and building executable computer instructions to automate tasks or solve problems. It is done using programming languages that the computer can understand and execute.

### प्रोग्रामिंग क्या है?

प्रोग्रामिंग वह प्रक्रिया है जिसमें हम कंप्यूटर को कोई कार्य करने के लिए निर्देश (instructions) देते हैं। यह निर्देश एक विशेष प्रोग्रामिंग भाषा में लिखे जाते हैं जिसे कंप्यूटर समझ सके और निष्पादित कर सके।

### 1. Basic Model of Computation

The basic model of computation involves how a computer processes data through:

- **Input:** Data is received from the user or environment.
- **Processing:** The CPU performs calculations or logic using the input.
- **Output:** Results are provided to the user through a display or another output device.

This is sometimes referred to as the **IPO model (Input → Process → Output)**.

### गणना का मूल मॉडल

इस मॉडल में यह बताया जाता है कि कंप्यूटर किस तरह से डेटा को प्रोसेस करता है:

- **इनपुट (Input):** उपयोगकर्ता या वातावरण से डेटा प्राप्त करना।



- **प्रोसेसिंग (Processing):** CPU द्वारा गणनाएँ या निर्णय लेना।
- **आउटपुट (Output):** परिणामों को उपयोगकर्ता को दिखाना या बाहर भेजना।

इसे आमतौर पर **IPO मॉडल (Input → Process → Output)** कहा जाता है।

## 2. Algorithms

An **algorithm** is a finite set of well-defined instructions to perform a task or solve a problem.

**Characteristics of a good algorithm:**

- Clear and unambiguous steps
- Finite number of steps
- Produces correct results
- Efficient in terms of time and memory

### एल्गोरिदम

एल्गोरिदम एक क्रमबद्ध, सीमित, और स्पष्ट निर्देशों का समूह होता है जो किसी समस्या का समाधान करता है।

**एक अच्छे एल्गोरिदम की विशेषताएँ:**

- स्पष्ट और असंदिग्ध निर्देश
- सीमित चरणों में समाप्त हो
- सही परिणाम दे
- समय और मेमोरी में कुशल हो

**Example:**

**To find the sum of two numbers:**



1. Start
  2. Input number1 and number2
  3. Calculate sum = number1 + number2
  4. Print sum
  5. End
- 

### 3. Flowcharts

A **flowchart** is a diagram that represents an algorithm or process using standard symbols:

- **Oval** – Start or End
- **Rectangle** – Process
- **Diamond** – Decision (yes/no)
- **Arrow** – Flow of control

Flowcharts help in visualizing the logic before writing code.

### फ्लोचार्ट

फ्लोचार्ट एक चित्रात्मक तरीका है एल्गोरिदम को दिखाने का। इसमें मानक चिह्नों का उपयोग होता है:

- **अंडाकार (Oval):** प्रारंभ या अंत
- **आयत (Rectangle):** प्रक्रिया (Process)
- **हीरा (Diamond):** निर्णय (Yes/No)
- **तीर (Arrow):** नियंत्रण का प्रवाह

फ्लोचार्ट से कोड लिखने से पहले लॉजिक को समझना आसान होता है।

---



## 4. Programming Languages

Programming languages are tools used to write programs that control the behavior of a machine.

They are classified as:

- **Low-Level Languages** – Machine language (binary) and Assembly language (closer to hardware)
- **High-Level Languages** – Python, Java, C++, etc. (easier for humans to read and write)

Python is popular due to its simplicity and readability.

### प्रोग्रामिंग भाषाएँ

प्रोग्रामिंग भाषाएँ ऐसे औजार हैं जिनका उपयोग कंप्यूटर प्रोग्राम लिखने में किया जाता है।

ये दो प्रकार की होती हैं:

- **लो-लेवल (Low-level):** मशीन भाषा, असेम्बली भाषा (हार्डवेयर के करीब)
- **हाई-लेवल (High-level):** पायथन, जावा, सी++ (मनुष्यों के लिए पढ़ने योग्य)

पायथन आज की लोकप्रिय भाषाओं में से एक है क्योंकि यह आसान और पठनीय है।

---

## 5. Compilation

Compilation is the process of converting source code written in a high-level language into machine code that the computer can execute.

- **Compiler:** Converts the entire code into machine code before execution.
- **Interpreter:** Converts and executes code line-by-line (e.g., Python).



## कंपाइलेशन

कंपाइलेशन वह प्रक्रिया है जिसमें हार्ड-लेवल कोड को मशीन कोड में बदला जाता है।

- **कंपाइलर:** पूरे कोड को पहले मशीन कोड में बदलता है और फिर निष्पादन करता है।
- **इंटरप्रेटर:** कोड को एक-एक लाइन करके पढ़ता और चलाता है। (जैसे: पायथन)

## 6. Testing and Debugging

- **Testing** is the process of checking whether the program produces the correct output for different inputs.
- **Debugging** is identifying and fixing errors (called bugs) in the code.

Types of bugs:

- **Syntax errors** (wrong language rules)
- **Logical errors** (wrong results)
- **Runtime errors** (crash during execution)

## परीक्षण और डिबगिंग

- **परीक्षण (Testing):** यह जाँचना कि प्रोग्राम अलग-अलग इनपुट पर सही परिणाम दे रहा है या नहीं।
- **डिबगिंग (Debugging):** कोड में मौजूद त्रुटियों को ढूँढ़ना और सुधारना।

त्रुटियों के प्रकार:

- **सिंटैक्स त्रुटि (Syntax Error):** भाषा के नियमों का उल्लंघन
- **लॉजिकल त्रुटि (Logical Error):** गलत परिणाम
- **रनटाइम त्रुटि (Runtime Error):** प्रोग्राम चलते समय रुक जाना



## 7. Documentation

Documentation involves writing comments and descriptions within or alongside code to help others (and future you) understand what the code does.

It includes:

- Code comments (# in Python)
- User manuals
- Technical documentation

### प्रलेखन

प्रलेखन में कोड के कार्य को समझाने के लिए टिप्पणियाँ और विवरण लिखे जाते हैं। यह अन्य डेवलपर्स और उपयोगकर्ताओं को कोड समझने में सहायता करता है।

इसमें शामिल होते हैं:

- कोड टिप्पणियाँ (# पायथन में)
- उपयोगकर्ता पुस्तिका
- तकनीकी दस्तावेज़

### One Liner

1. Programming is the process of writing instructions for computers.  
प्रोग्रामिंग कंप्यूटर के लिए निर्देश लिखने की प्रक्रिया है।
2. An algorithm is a step-by-step method to solve a problem.  
एल्गोरिदम समस्या को हल करने की चरणबद्ध विधि है।
3. A flowchart is a diagrammatic representation of an algorithm.  
फ्लोचार्ट एल्गोरिदम का चित्रात्मक रूप होता है।



4. Programming languages include Python, C, Java, etc.

प्रोग्रामिंग भाषाएं जैसे Python, C, Java आदि होती हैं।

5. Compilation converts code to machine code.

कंपाइलेशन कोड को मशीन कोड में बदलता है।

6. Testing and debugging help in identifying and fixing errors.

टेस्टिंग और डिबगिंग त्रुटियों की पहचान और सुधार में मदद करते हैं।

7. Documentation helps in maintaining code clarity and reusability.

डॉक्यूमेंटेशन कोड की स्पष्टता और पुनः उपयोग के लिए सहायक होता है।

## MCQ

**1. What does IPO in programming stand for?**

प्रोग्रामिंग में IPO का क्या अर्थ होता है?

A) Input – Processing – Output

B) Input – Program – Output

C) Instruction – Processing – Output

D) Input – Processing – Object

**Explanation:** IPO model describes the basic functioning of a computer: taking input, processing it, and giving output.

IPO मॉडल कंप्यूटर के कार्य को दर्शाता है – इनपुट लेना, प्रोसेस करना, और आउटपुट देना।

एल्गोरिदम क्या है?

A) A programming language

B) A type of error

C) A step-by-step procedure to solve a problem

D) A code editor

**Correct Answer: C) A step-by-step procedure to solve a problem**

**Explanation :** An algorithm is a set of clear instructions to solve a specific problem.

एल्गोरिदम समस्या को हल करने के लिए क्रमबद्ध निर्देशों का समूह होता है।

**2. What is an algorithm?**



**3. Which of the following is used to visually represent an algorithm?**

निम्नलिखित में से किसका उपयोग एल्गोरिदम को चित्र रूप में दिखाने के लिए किया जाता है?

- A) Pseudocode
- B) Flowchart
- C) Compiler
- D) Debugger

**Correct Answer: B) Flowchart**

**Explanation :** Flowcharts use symbols to visually describe logic and flow of a program.

फ्लोचार्ट प्रतीकों का उपयोग करके एल्गोरिदम के लॉजिक और प्रवाह को दिखाता है।

**4. Which symbol in a flowchart represents a decision?**

फ्लोचार्ट में निर्णय (Decision) को दर्शने के लिए कौन सा चिन्ह प्रयोग होता है?

- A) Oval
- B) Rectangle
- C) Diamond
- D) Circle

**Correct Answer: C) Diamond**

**Explanation :** A diamond in a flowchart shows a decision point (Yes/No).

फ्लोचार्ट में डायमंड (हीरा) निर्णय बिंदु (हाँ/नहीं) को दर्शाता है।

**5. What is debugging in programming?**

प्रोग्रामिंग में डिबगिंग क्या होती है?

- A) Writing comments
- B) Typing code
- C) Finding and fixing errors
- D) Converting code to machine language

**Correct Answer: C) Finding and fixing errors**

**Explanation :** Debugging is the process of locating and correcting bugs in the code. डिबगिंग कोड में त्रुटियों को ढूँढ़ने और उन्हें ठीक करने की प्रक्रिया है।

**6. What is the main purpose of documentation in programming?**

प्रोग्रामिंग में प्रलेखन (documentation) का मुख्य उद्देश्य क्या है?



- A) Make code faster
- B) Explain the code to others
- C) Convert code to binary
- D) Increase memory usage

**Correct Answer: B) Explain the code to others**

**Explanation :** Documentation helps others understand the code for use and maintenance.

प्रलेखन दूसरों को कोड को समझने और बनाए रखने में सहायता करता है।

### 7. Which of the following is a high-level programming language?

निम्नलिखित में से कौन सी एक हाई-लेवल प्रोग्रामिंग भाषा है?

- A) Machine Language
- B) Assembly
- C) Python
- D) Binary Code

**Correct Answer: C) Python**

**Explanation :** Python is a high-level language that is easy to read and write.

पायथन एक हाई-लेवल भाषा है जो पढ़ने और लिखने में सरल होती है।

### 8. What type of error occurs when a program runs but gives the wrong result?

किस प्रकार की त्रुटि तब होती है जब प्रोग्राम चलता है लेकिन गलत परिणाम देता है?

- A) Syntax Error
- B) Runtime Error
- C) Logical Error
- D) Compiler Error

**Correct Answer: C) Logical Error**

**Explanation :** Logical errors occur when the logic used in the program is incorrect, even though it runs without crashing.

लॉजिकल त्रुटियाँ तब होती हैं जब प्रोग्राम का लॉजिक गलत होता है, हालांकि प्रोग्राम बिना रुके चलता है।

### 9. Which tool is used to convert high-level language to machine code at once?

कौन सा उपकरण हाई-लेवल भाषा को एक साथ मशीन कोड में बदलता है?



- A) Interpreter
- B) Debugger
- C) Editor
- D) Compiler

**Correct Answer: D) Compiler**

**Explanation :** A compiler translates the entire source code into machine code before execution.

कंपाइलर पूरे स्रोत कोड को एक साथ मशीन कोड में अनुवादित करता है और फिर चलाता है।

**10. Which of the following is NOT a phase of programming?**

निम्नलिखित में से कौन प्रोग्रामिंग का चरण (phase) नहीं है?

- A) Designing
- B) Coding
- C) Washing
- D) Testing

**Correct Answer: C) Washing**

**Explanation :** Washing is unrelated; programming involves designing, coding, testing, and debugging.

वॉशिंग (Washing) का प्रोग्रामिंग से कोई लेना-देना नहीं है; इसमें डिजाइनिंग, कोडिंग, परीक्षण और डिबगिंग शामिल होते हैं।



## Chapter 2

### Algorithms and Flowcharts to Solve Problems

(समस्याओं को हल करने के लिए एल्गोरिदम और फ्लोचार्ट)

In programming, **algorithms** and **flowcharts** are essential tools used to design logical solutions before coding begins.

#### Flowchart Symbols (फ्लोचार्ट के चिन्ह):

Symbol	Meaning	Use in Flowchart
oval (Oval)	Start/End	Begin or end of the flow
rectangle (Rectangle)	Process	Any computation or action
diamond (Diamond)	Decision	Condition check (Yes/No)
arrow (Arrow)	Flow of control	Shows direction of steps

#### Types of Processing in Algorithms and Flowcharts:

##### 1. Sequential Processing (क्रमानुसार प्रोसेसिंग)

Steps are executed in order, one after another.

Example: Calculate sum = a + b

##### 2. Decision-based Processing (निर्णय आधारित प्रोसेसिंग)



Conditions are evaluated to decide which path to take (if-else).

Example: Check if a number is even or odd.

### 3. Iterative Processing (पुनरावृत्ति प्रोसेसिंग)

Loops (while, for) are used to repeat actions.

Example: Print numbers from 1 to 10

#### Examples of Basic Algorithms and Flowcharts:

##### 1. Exchange Values of Two Variables (दो वेरिएबल के मानों की अदला-बदली)

###### Algorithm:

1. Start
2. Input A, B
3. Temp = A
4. A = B
5. B = Temp
6. Print A, B
7. End

##### 2. Sum of a Set of Numbers (संख्याओं के समूह का योग)

###### Algorithm (using loop):

1. Start
2. Input n (count of numbers)
3. Set sum = 0
4. Repeat from i=1 to n:



- Input number
  - $\text{sum} = \text{sum} + \text{number}$
5. Print sum
6. End
- 

### 3. Decimal to Binary Conversion (दशमलव से बाइनरी में रूपांतरण)

#### Algorithm:

- 
- 1. Start
  - 2. Input Decimal number D
  - 3. While  $D > 0$ :
    - $R = D \% 2$
    - Store R
    - $D = D // 2$
  - 4. Print binary digits in reverse
  - 5. End
- 

### 4. Reverse Digits of an Integer (एक पूर्णांक की अंकों को उलटना)

#### Algorithm:

- 1. Start
- 2. Input number N
- 3.  $\text{rev} = 0$
- 4. While  $N > 0$ :
  - $\text{rem} = N \% 10$



- o  $\text{rev} = \text{rev} * 10 + \text{rem}$
  - o  $\text{N} = \text{N} // 10$
5. Print rev
6. End
- 

## 5. GCD of Two Numbers (दो संख्याओं का महत्तम समान भाजक - GCD)

### Algorithm:

- 
- 1. Start
  - 2. Input A, B
  - 3. While  $B \neq 0$ :
    - o  $\text{temp} = B$
    - o  $B = A \% B$
    - o  $A = \text{temp}$
  - 4. GCD = A
  - 5. Print GCD
  - 6. End
- 

## 6. Test Prime Number (संख्या अभाज्य है या नहीं जांचना)

### Algorithm:

- 1. Start
- 2. Input number N
- 3. If  $N \leq 1 \rightarrow$  Not prime
- 4. For  $i = 2$  to  $\text{sqrt}(N)$ :



- o If  $N \% i == 0 \rightarrow$  Not prime
  - 5. Else  $\rightarrow$  Prime
  - 6. End
- 

## 7. Factorial of a Number (संख्या का फैक्टोरियल)

### Algorithm:

- 1. Start
  - 2. Input N
  - 3. fact = 1
  - 4. For  $i = 1$  to N:
    - o fact = fact \* i
  - 5. Print fact
  - 6. End
- 

## 8. Fibonacci Sequence (फिबोनाची अनुक्रम)

### Algorithm:

- 1. Start
- 2. Input n (number of terms)
- 3. a = 0, b = 1
- 4. Print a, b
- 5. For  $i = 3$  to n:
  - o  $c = a + b$
  - o Print c



- o  $a = b, b = c$
  - 6. End
- 

## 9. Evaluate sin(x) as a Series (sin(x) को श्रृंखला के रूप में निकालना)

Use Taylor series expansion.

$$\sin(x) = x - x^3/3! + x^5/5! - x^7/7! + \dots$$

This requires:

- Factorial calculation
  - Power calculation
  - Looping for fixed number of terms
- 

## 10. Reverse Order of Elements in an Array (ऐरे को उल्टे क्रम में करना)

**Algorithm:**

1. Start
  2. Input array elements
  3. For  $i = 0$  to  $n/2$ :
    - o Swap  $\text{arr}[i]$  with  $\text{arr}[n-1-i]$
  4. Print array
  5. End
-



## 11. Find Largest Number in an Array (ऐसे में सबसे बड़ी संख्या निकालना)

### Algorithm:

1. Start
2. Input n elements into array
3. max = arr[0]
4. For i = 1 to n-1:
  - o If arr[i] > max → max = arr[i]
5. Print max
6. End

## 12. Print Upper Triangular Matrix (ऊपरी त्रिकोणीय मैट्रिक्स प्रिंट करना)

### Algorithm:

1. Start
2. Input matrix A[n][n]
3. For i = 0 to n-1:
  - o For j = 0 to n-1:
    - If i > j → print "0"
    - Else → print A[i][j]
4. End

MCQ



**1. Which symbol in a flowchart is used to show a process or instruction?**

फ्लोचार्ट में किसी प्रक्रिया या निर्देश को दर्शने के लिए कौन सा चिन्ह प्रयोग किया जाता है?

- A) Oval
- B) Diamond
- C) Rectangle
- D) Parallelogram

**Correct Answer: C) Rectangle**

**Explanation :** A rectangle in a flowchart represents a process or a calculation.

फ्लोचार्ट में आयत (rectangle) किसी प्रक्रिया या गणना को दर्शाता है।

**2. What does a diamond symbol in a flowchart represent?**

फ्लोचार्ट में डायमंड (हीरे) का चिन्ह क्या दर्शाता है?

- A) Start/End
- B) Input/Output
- C) Decision
- D) Loop

**Correct Answer: C) Decision**

**Explanation :** A diamond symbol is used

to represent decision-making or conditional branches (Yes/No).

डायमंड चिह्न निर्णय लेने या कंडीशन पर आधारित शाखाओं (हाँ/नहीं) को दर्शाता है।

**3. What is the first step in developing an algorithm?**

एल्गोरिदम बनाने की पहली प्रक्रिया क्या है?

- A) Writing code
- B) Testing
- C) Understanding the problem
- D) Drawing flowchart

**Correct Answer: C) Understanding the problem**

**Explanation :** The first step is to clearly understand the problem so a correct solution can be planned.

सबसे पहला कदम है समस्या को अच्छी तरह समझना ताकि उसका सही समाधान तैयार किया जा सके।

**4. What is the output of the following algorithm?**

- Input: A = 5, B = 7



- Temp = A
- A = B
- B = Temp
- Print A, B

**इस एल्गोरिदम का आउटपुट क्या होगा?**

A = 5, B = 7 के लिए

- A) 5, 7
- B) 7, 5
- C) 0, 0
- D) 12, 0

**Correct Answer: B) 7, 5**

**Explanation :** The algorithm swaps the values of A and B using a temporary variable.

यह एल्गोरिदम एक अस्थायी वेरिएबल का उपयोग करके A और B के मानों की अदला-बदली करता है।

### 5. Which of the following is an example of iterative processing?

निम्नलिखित में से कौन पुनरावृत्ति प्रोसेसिंग (Iterative Processing) का उदाहरण है?

- A) Checking if a number is even or odd
- B) Adding two numbers

- C) Printing numbers from 1 to 10
- D) Finding the largest of two numbers

**Correct Answer: C) Printing numbers from 1 to 10**

**Explanation :** Iterative processing involves repeating a set of instructions, like a loop.

पुनरावृत्ति प्रोसेसिंग में निर्देशों को बार-बार दोहराया जाता है, जैसे कि लूप में।

### 6. Which algorithm is used to find GCD of two numbers efficiently?

दो संख्याओं का GCD (महत्तम समान भाजक) कुशलता से निकालने के लिए कौन सा एल्गोरिदम प्रयोग किया जाता है?

- A) Prime Check
- B) Euclidean Algorithm
- C) Fibonacci Algorithm
- D) Summation Algorithm

**Correct Answer: B) Euclidean Algorithm**

**Explanation :** Euclidean algorithm uses remainder and division repeatedly to find GCD.

यूक्लिडियन एल्गोरिदम भाजक और शेषफल का प्रयोग करके GCD निकालता है।



### 7. What is the purpose of writing an algorithm before coding?

कोडिंग से पहले एल्गोरिदम लिखने का उद्देश्य क्या है?

- A) To run the code faster
- B) To reduce memory usage
- C) To plan and solve the problem logically
- D) To debug the program

**Correct Answer: C) To plan and solve the problem logically**

**Explanation :** Algorithms help plan the logic of the program step by step.

एल्गोरिदम प्रोग्राम को चरणबद्ध तरीके से तर्कसंगत रूप से हल करने में सहायता करता है।

### 8. What does a parallelogram represent in a flowchart?

फ्लोचार्ट में समांतर चतुर्भुज (Parallelogram) क्या दर्शाता है?

- A) Start/End
- B) Process
- C) Decision
- D) Input/Output

**Correct Answer: D) Input/Output**

**Explanation :** A parallelogram in a flowchart represents input or output operations like reading or displaying data. फ्लोचार्ट में समांतर चतुर्भुज इनपुट या आउटपुट (जैसे डेटा पढ़ना या दिखाना) को दर्शाता है।

### 9. In which of the following problems is a loop NOT required?

निम्नलिखित में से किस समस्या को हल करने के लिए लूप की आवश्यकता नहीं होती?

- A) Printing Fibonacci series
- B) Finding factorial of a number
- C) Swapping two variables
- D) Summation of N numbers

**Correct Answer: C) Swapping two variables**

**Explanation :** Swapping values is a one-time process and does not need iteration. दो मानों की अदला-बदली एक बार की प्रक्रिया है, इसके लिए लूप की आवश्यकता नहीं होती।

### 10. Which of the following best defines an algorithm?



760 741 8817



New Ideas YT



www.examjila.com

निम्नलिखित में से कौन एल्गोरिदम को सबसे अच्छी तरह परिभाषित करता है?

- A) A programming language
- B) A flowchart symbol
- C) A step-by-step solution to a problem
- D) A compiled code

**Correct Answer:** C) A step-by-step solution to a problem

**Explanation :** An algorithm is a finite set of instructions to solve a problem logically and systematically.

एल्गोरिदम समस्या को तर्कसंगत और क्रमबद्ध रूप से हल करने के लिए चरणों का एक सीमित समूह होता है।





## Chapter 3

### Introduction to Python (पाइथन का परिचय)

**Python** is a high-level, interpreted, general-purpose programming language. It is known for its readability and simplicity. Python supports multiple programming paradigms including object-oriented, procedural, and functional programming.

**Python** एक उच्च स्तरीय (High-level), इंटरप्रेटेड (Interpreted), सामान्य प्रयोजन वाली प्रोग्रामिंग भाषा है। यह अपनी सरलता और पठनीयता के लिए जानी जाती है। पाइथन कई प्रोग्रामिंग शैलियों को समर्थन करता है जैसे ऑब्जेक्ट-ऑरिएंटेड, प्रोसीजरल और फंक्शनल प्रोग्रामिंग।

#### Technical Strengths of Python (पाइथन की तकनीकी विशेषताएं):

Easy to learn and use	सीखना और उपयोग करना आसान
Interpreted language	इंटरप्रेटेड भाषा
Cross-platform compatibility	कई प्लेटफॉर्म पर चलता है
Large standard library	विशाल स्टैंडर्ड लाइब्रेरी
Supports GUI, Web, AI, ML, etc.	GUI, वेब, AI, ML आदि को सपोर्ट करता है

#### Python Interpreter and Program Execution (पाइथन इंटरप्रेटर और प्रोग्राम निष्पादन)

- **Interpreter** runs code line-by-line.
- You can use **Python Shell (REPL)** or create a .py file and run using python filename.py.



### Example:

```
print("Hello, World!")
```

### Output:

Hello, World!

---

### Comments in Python (टिप्पणियाँ):

Used to make code readable. Python ignores comments during execution.

```
# This is a comment  
print("Python is awesome") # This is also a comment
```

---

### Literals and Constants (लिटरल और स्थिरांक):

- **Literals** are fixed values in code like: 10, "Hello", 3.14, etc.
- **Constants** are values that do not change, like:

PI = 3.14159

---

### Built-in Data Types (इनबिल्ट डेटा टाइप्स):

Type	Example
Integer	x = 5
Float	y = 3.14
Complex	z = 2 + 3j
Boolean	flag = True



760 741 8817



New Ideas YT



www.examjila.com

String	s = "Python"
Set	a = {1, 2, 3}

### String Operations (स्ट्रिंग पर ऑपरेशन):

```
s = "Python"  
print(s[0])      # P  
print(s[1:4])    # yth  
print(s + "3")   # Python3  
print(len(s))   # 6
```

### Accepting Input and Printing Output (इनपुट लेना और आउटपुट देना):

```
name = input("Enter your name: ") # User input  
print("Hello", name)
```

### Simple Python Program (सरल पाइथन प्रोग्राम):

```
# Program to add two numbers  
a = int(input("Enter first number: "))  
b = int(input("Enter second number: "))  
print("Sum is:", a + b)
```

### One Liner



1. Python is simple, readable, and powerful.  
पाइथन सरल, पठनीय और शक्तिशाली है।
2. Python supports multiple programming paradigms.  
पाइथन कई प्रोग्रामिंग शैलियों का समर्थन करता है।
3. Interpreter executes code line-by-line.  
इंटरप्रेटर कोड को पंक्ति दर पंक्ति निष्पादित करता है।
4. Comments use # and help explain code.  
टिप्पणियाँ # से शुरू होती हैं और कोड को समझाने में मदद करती हैं।
5. Python literals include numbers, strings, and booleans.  
पाइथन लिटरल्स में संख्याएं, स्ट्रिंग्स और बूलियन शामिल होते हैं।

## MCQ

**1. What is the output of:**

`print("Python"[1:4])?`

निम्न कोड का आउटपुट क्या होगा:

`print("Python"[1:4])?`

- A) Pyt
- B) ytho
- C) yth
- D) tho

**Correct Answer: C) yth**

निम्नलिखित में से कौन पाइथन में इनबिल्ट डेटा टाइप नहीं है?

- A) Integer
- B) String
- C) Float
- D) Character

**Correct Answer: D) Character**

Python treats characters as strings of length 1.

**2. Which of the following is not a built-in data type in Python?**

**3. What will be the output of this code?**

```
x = 2 + 3j
print(type(x))
```



- A) <class 'int'>
- B) <class 'float'>
- C) <class 'complex'>
- D) <class 'string'>

**Correct Answer: C) <class 'complex'>**

---

#### 4. Which function is used to take user input in Python?

पाइथन में उपयोगकर्ता इनपुट लेने के लिए कौन सा फंक्शन प्रयोग किया जाता है?

- A) scanf()
- B) cin>>
- C) input()
- D) read()

**Correct Answer: C) input()**

---

#### 5. Which of these is the correct way to write a comment in Python?

पाइथन में टिप्पणी (comment) लिखने का सही तरीका क्या है?

- A) // This is a comment
- B) <!-- This is a comment -->

- C) # This is a comment
- D) /\* This is a comment \*/

**Correct Answer: C) # This is a comment**

#### 6. What is the result of this code?

```
x = "Python"
```

```
print(x[-1])
```

- A) P
- B) n
- C) o
- D) h

**Correct Answer: B) n**

**Explanation :** -1 refers to the last character in a string.

-1 का उपयोग स्ट्रिंग के अंतिम अक्षर को प्राप्त करने के लिए होता है।

---

#### 7. Which data type is used to store a group of unique, unordered elements?

किस डेटा प्रकार का उपयोग अद्वितीय (unique) और अव्यवस्थित (unordered) तत्वों को संग्रहित करने के लिए किया जाता है?

- A) List
- B) Tuple



- C) Dictionary  
D) Set

**Correct Answer: D) Set**

**Explanation :** Sets store unique elements without any order.

सेट में सभी तत्व अद्वितीय होते हैं और उनका कोई निश्चित क्रम नहीं होता।

#### 8. What will be the output of the following code?

```
print(type(5.0))
```

- A) <class 'int'>  
B) <class 'float'>  
C) <class 'double'>  
D) <class 'decimal'>

**Correct Answer: B) <class 'float'>**

**Explanation :** Any number with a decimal point is considered a float in Python.

दशमलव वाले सभी नंबर पाइथन में float माने जाते हैं।

#### 9. Which of the following is used to slice a string in Python?

पाइथन में स्ट्रिंग को स्लाइस (टुकड़ों में काटने) करने के लिए क्या उपयोग किया जाता है?

- A) {}  
B) ()  
C) []  
D) <>

**Correct Answer: C) []**

**Explanation :** Square brackets with a colon : are used for slicing strings.

स्ट्रिंग को स्लाइस करने के लिए वर्ग कोष्ठक ([]) और कॉलन (:) का उपयोग होता है।

#### 10. Which one is a valid string literal in Python?

निम्नलिखित में से कौन सा पाइथन में एक वैध स्ट्रिंग लिटरल है?

- A) 'Hello'  
B) "World"  
C) ""Python""  
D) All of the above

**Correct Answer: D) All of the above**



760 741 8817



New Ideas YT



www.examjila.com

**Explanation :** Python allows single, double, and triple quotes for string literals.

पाइथन में स्ट्रिंग को दर्शनी के लिए सिंगल, डबल और ट्रिपल कोट्स सभी वैध होते हैं।





## Chapter 4

### Operators, Expressions and Python Statements

(ऑपरेटर, एक्सप्रेशन्स और पाइथन स्टेटमेंट्स)

In Python, **operators** are special symbols used to perform operations on variables and values. **Expressions** are combinations of values and operators that Python interprets to produce another value. **Statements** are instructions that Python can execute, like assignments or loops.

Python में ऑपरेटर वे विशेष चिन्ह होते हैं जिनका उपयोग मानों (values) और वेरिएबल्स पर क्रियाएं (operations) करने के लिए किया जाता है। एक्सप्रेशन्स मानों और ऑपरेटर का समूह होते हैं जो किसी मूल्य (value) को जन्म देते हैं। स्टेटमेंट्स वे निर्देश (instructions) होते हैं जिन्हें पाइथन निष्पादित कर सकता है, जैसे असाइनमेंट या लूप।

#### Assignment Statement (असाइनमेंट स्टेटमेंट)

Used to assign values to variables:

x = 10

name = "Python"

#### Expressions (एक्सप्रेशन्स)

Combination of variables, literals, and operators:

a = 5



```
b = 3
```

```
c = a + b # Expression: a + b
```

---

### Types of Operators (ऑपरेटर के प्रकार):

Operator Type	Description	उदाहरण (Hindi)
<b>Arithmetic</b>	<code>+, -, *, /, %, **, //</code>	<code>a + b, a ** b</code>
<b>Relational</b>	<code>==, !=, &lt;, &gt;, &lt;=, &gt;=</code>	<code>a &gt; b</code>
<b>Logical</b>	<code>and, or, not</code>	<code>a &gt; 5 and b &lt; 10</code>
<b>Bitwise</b>	<code>&amp;, `</code>	<code>, ^, ~, &lt;&lt;, &gt;&gt;`</code>
<b>Assignment</b>	<code>=, +=, -=, *=, /=, etc.</code>	<code>x += 2</code>
<b>Identity</b>	<code>is, is not</code>	<code>a is b</code>
<b>Membership</b>	<code>in, not in</code>	<code>'a' in 'apple'</code>

### Precedence of Operators (ऑपरेटर की प्राथमिकता):

Higher precedence operators are evaluated first.

```
result = 10 + 5 * 2 # result = 20, not 30
```

---

### Conditional Statements (नियतात्मक कथन)

#### if Statement:

```
if
if x > 0:
    print("Positive number")
```

**if-else Statement:**

```
if x % 2 == 0:  
    print("Even")  
else:  
    print("Odd")
```

**if-elif-else:**

```
if score >= 90:  
    print("A grade")  
elif score >= 75:  
    print("B grade")  
else:  
    print("C grade")
```

**Iterative Computation and Control Flow (पुनरावृत्त गणना और नियंत्रण प्रवाह)****range() function:**

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

**while loop:**

```
i = 0  
while i < 5:  
    print(i)  
    i += 1
```

**for loop:**



```
for char in "Python":  
    print(char)
```

---

### Control Flow Statements (नियंत्रण प्रवाह स्टेटमेंट्स):

Statement	Description
<b>break</b>	Exit the loop immediately
<b>continue</b>	Skip to the next iteration
<b>pass</b>	Placeholder; does nothing
<b>else</b>	Runs if no break occurred
<b>assert</b>	Debugging aid to test conditions

### Example using break, continue, pass:

```
for i in range(5):  
    if i == 2:  
        continue # Skips 2  
    elif i == 4:  
        break # Exits on 4  
    print(i)
```

### # Pass Example

```
for i in range(3):  
    pass # No action here
```

---



## One Liner

1. Arithmetic: +, -, \*, /, //, %, \*\*  
अंकगणितीय ऑपरेटर: +, -, \*, /, //, %, \*\*
2. Logical: and, or, not  
लॉजिकल: and, or, not
3. Bitwise: &, |, ^, ~, <<, >>  
बिटवाइज़: &, |, ^, ~, <<, >>
4. Conditional: if, if-else, elif  
कंडीशनल: if, if-else, elif
5. Loops: for, while  
लूप्स: for, while
6. Loop control: break, continue, pass  
लूप नियंत्रण: break, continue, pass

## MCQ

1. What is the output of: `print(2 + 3 * 4)`?

- A) 20
- B) 14
- C) 24
- D) 18

Correct Answer: B) 14

2. Which of these is not a logical operator in Python?

- A) and
- B) or
- C) not
- D) xor

Correct Answer: D) xor

**3. What will be the output?**

a = 5

b = 3

print(a &amp; b)

A) 1

B) 2

C) 7

D) 0

**Correct Answer: A) 1****4. Which loop will definitely execute at least once?**

A) for loop

B) while loop

C) do-while loop

D) None

**Correct Answer: A) for loop***(Python has no do-while loop)***5. What is the output?**

x = 10

if x &gt; 5:

print("Yes")

else:

print("No")

A) Yes

B) No

C) Error

D) None

**Correct Answer: A) Yes****1. Which of the following is a relational operator in Python?**

निम्नलिखित में से कौन सा रिलेशनल ऑपरेटर है?

A) =

B) ==

C) +

D) !

**Correct Answer: B) ==****Explanation:** == checks for equality between two values.**विवरण:** == दो मानों की समानता की जांच करता है।**2. Which of these has the highest precedence?**

इनमें से किसकी प्राथमिकता सबसे अधिक होती है?



- A) +
- B) and
- C) \*
- D) ==

**Correct Answer: C) \* (Multiplication)**

**Explanation:** Arithmetic operators like \* have higher precedence than logical and relational operators.

**विवरण:** गणितीय ऑपरेटर जैसे \* की प्राथमिकता उच्च होती है।

### 3. What does the following code print?

```
x = 10
y = 5
print(x > y and y < 3)
```

- A) True
- B) False
- C) Error
- D) None

**Correct Answer: B) False**

**Explanation:**  $x > y$  is True,  $y < 3$  is False → True and False = False.

**विवरण:** एक सत्य और एक असत्य होने पर and False देता है।

### 4. What will print(3 << 1) return?

**print(3 << 1) का आउटपुट क्या होगा?**

- A) 6
- B) 1
- C) 5
- D) 7

**Correct Answer: A) 6**

**Explanation:** Bitwise left shift:  $3 (011) << 1 = 6 (110)$

**विवरण:** बिटवाइज बाएं शिफ्ट से संख्या दोगुनी हो जाती है।

### 5. Which statement is used as a placeholder?

**कौन सा स्टेटमेंट प्लेसहोल्डर की तरह प्रयोग होता है?**

- A) continue
- B) break
- C) pass
- D) assert

**Correct Answer: C) pass**

**Explanation:** pass does nothing; it acts as a placeholder.



**विवरण:** pass कोई क्रिया नहीं करता, केवल एक प्लेसहोल्डर है।

### 6. Which keyword is used to terminate a loop immediately?

किस कीवर्ड का उपयोग लूप को तुरंत समाप्त करने के लिए होता है?

- A) break
- B) stop
- C) exit
- D) return

**Correct Answer: A) break**

**Explanation:** break exits the loop as soon as it is encountered.

**विवरण:** break से लूप तुरंत समाप्त हो जाता है।

### 7. What is the output of this code?

for i in range(3):

```
    if i == 1:
        continue
    print(i)
```

- A) 0 1 2
- B) 0 2

C) 1 2

D) 0 1

**Correct Answer: B) 0 2**

**Explanation:** When  $i == 1$ , continue skips the print.

**विवरण:** continue  $i == 1$  के लिए प्रिंट स्किप कर देता है।

### 8. What does assert do in Python?

Python में assert क्या करता है?

- A) Checks syntax
- B) Acts like a breakpoint
- C) Tests conditions and raises error if False
- D) Terminates program

**Correct Answer: C)**

**Explanation:** assert is used to debug by testing conditions.

**विवरण:** assert एक परीक्षण स्टेटमेंट है, जो असत्य होने पर एरर देता है।

### 9. Which loop is best when the number of iterations is known?



760 741 8817



New Ideas YT



www.examjila.com

जब पुनरावृत्ति की संख्या इकात हो तो कौन सा लूप  
सबसे उपयुक्त है?

- A) while loop
- B) for loop
- C) do-while loop
- D) infinite loop

**Correct Answer: B) for loop**

**Explanation:** for loop is preferred when range is predefined.

**विवरण:** for लूप तब उपयोगी होता है जब रेंज पहले से पता हो।

**10. What is the output of:**

```
x = 5  
x += 3  
print(x)
```

- A) 5
- B) 8
- C) 3
- D) 53

**Correct Answer: B) 8**

**Explanation:**  $x += 3$  is equivalent to  $x = x + 3$ .

**विवरण:**  $+=$  एक शॉर्टहैंड असाइनमेंट ऑपरेटर है।



## Chapter 5

### Sequence Data Types (क्रमिक डेटा प्रकार)

#### What are Sequence Data Types? (क्रमिक डेटा प्रकार क्या हैं?)

Sequence data types in Python are data structures that store multiple items in an ordered way.

Python में क्रमिक डेटा प्रकार वे डेटा स्ट्रक्चर हैं जो कई आइटम्स को क्रमबद्ध (ordered) रूप में संग्रहित करते हैं।

#### 1. List (सूची)

**Definition:** A list is an ordered, **mutable** collection of items.

**परिभाषा:** लिस्ट एक क्रमबद्ध और परिवर्तनशील (**mutable**) संग्रह होता है।

```
fruits = ['apple', 'banana', 'cherry']
print(fruits[1]) # banana
```

#### List Features:

- Mutable (can change elements)
- Allows duplicates
- Indexing, Slicing supported

#### Operations on Lists:

```
fruits.append("orange") # Add
fruits.remove("banana") # Remove
```



```
fruits[0] = "grapes"      # Modify  
print(fruits[1:3])       # Slicing
```

---

## 2. Tuple (ट्यूपल)

**Definition:** A tuple is an ordered, **immutable** collection of items.

**परिभाषा:** ट्यूपल एक क्रमबद्ध लेकिन **अपरिवर्तनशील** (**immutable**) संग्रह होता है।

```
colors = ('red', 'green', 'blue')  
print(colors[2])  # blue
```

### Tuple Features:

- Immutable (cannot change after creation)
- Faster than lists
- Allows indexing and slicing

## 3. Dictionary (डिक्शनरी)

**Definition:** Dictionary stores **key-value** pairs.

**परिभाषा:** डिक्शनरी में डेटा कुंजी-मूल्य (**key-value**) के रूप में संग्रहीत होता है।

```
student = {'name': 'John', 'age': 20}  
print(student['name'])  # John
```

### Dictionary Features:

- Unordered (as of older versions), now ordered (3.7+)
- Keys must be unique and immutable



- Values can be of any type
- 

## Indexing and Slicing (अनुक्रमण और स्लाइसिंग)

```
nums = [10, 20, 30, 40, 50]
print(nums[2])    # Indexing → 30
print(nums[1:4])  # Slicing → [20, 30, 40]
```

---

## Concatenation (संकलन):

```
a = [1, 2]
b = [3, 4]
print(a + b)  # [1, 2, 3, 4]
```

---

## Concept of Mutability (परिवर्तनशीलता का विचार):

- **List** is mutable → list[0] = value works
  - **Tuple** is immutable → tuple[0] = value gives error
  - **Dictionary** keys are immutable, values are mutable
- 

## Common Examples:

### Finding Maximum, Minimum, Mean:

```
nums = [5, 10, 15, 20]
print(max(nums))  # 20
```



760 741 8817



New Ideas YT



www.examjila.com

```
print(min(nums)) # 5  
print(sum(nums)/len(nums)) # Mean → 12.5
```

### Linear Search in List/Tuple:

```
nums = [3, 7, 2, 9]
```

```
x = 7
```

```
found = False
```

```
for i in nums:
```

```
    if i == x:
```

```
        found = True
```

```
        break
```

```
print("Found:", found)
```

### Counting Frequency using Dictionary:

```
items = ['apple', 'banana', 'apple', 'cherry', 'banana']
```

```
freq = {}
```

```
for item in items:
```

```
    if item in freq:
```

```
        freq[item] += 1
```

```
    else:
```

```
        freq[item] = 1
```

```
print(freq) # {'apple': 2, 'banana': 2, 'cherry': 1}
```

### One Liner



1. List: Mutable, ordered collection

लिस्ट: परिवर्तनीय, क्रमबद्ध संग्रह

2. Tuple: Immutable list

ट्यूपल: अपरिवर्तनीय लिस्ट

3. Dictionary: Key-value pairs

डिक्शनरी: कुंजी-मूल्य जोड़

4. String: Immutable character sequence

स्ट्रिंग: अपरिवर्तनीय अक्षर शृंखला

5. Slicing and indexing allow access to elements.

स्लाइसिंग और इंडेक्सिंग तत्वों तक पहुंच प्रदान करते हैं।

## MCQ

**1. Which of these is mutable in Python?**

इनमें से कौन Python में परिवर्तनशील है?

- A) List
- B) Tuple
- C) String
- D) All of the above

**Correct Answer: A) List**

C) Error

D) 4

**Correct Answer: B) 3**

**3. Which method is used to add an item to the list?**

- A) insert()
- B) append()
- C) add()
- D) put()

**Correct Answer: B) append()**

**2. What will len((1,2,3)) return?**

- A) 2
- B) 3



4. What is the result of 'a' in ['a', 'b', 'c']?

- A) False
- B) True
- C) Error
- D) None

**Correct Answer: B) True**

---

5. Which data type does not support item assignment?

- A) list
- B) tuple
- C) dictionary
- D) set

**Correct Answer: B) tuple**

---

6. What will be the output of:

```
d = {'x': 1, 'y': 2}  
print(d['z'])
```

- A) 2
- B) None
- C) Error
- D) 0

**Correct Answer: C) Error**

---

7. How do you get all keys from a dictionary?

- A) dict.values()
- B) dict.items()
- C) dict.keys()
- D) dict.get()

**Correct Answer: C) dict.keys()**

---

8. Which of the following creates a tuple?

- A) t = (1, 2)
- B) t = [1, 2]
- C) t = {1:2}
- D) t = 1, 2

**Correct Answer: A) t = (1, 2)**

---

9. What does slicing do?

- A) Edits an element
- B) Deletes an item



- C) Returns a portion of the sequence
- D) None

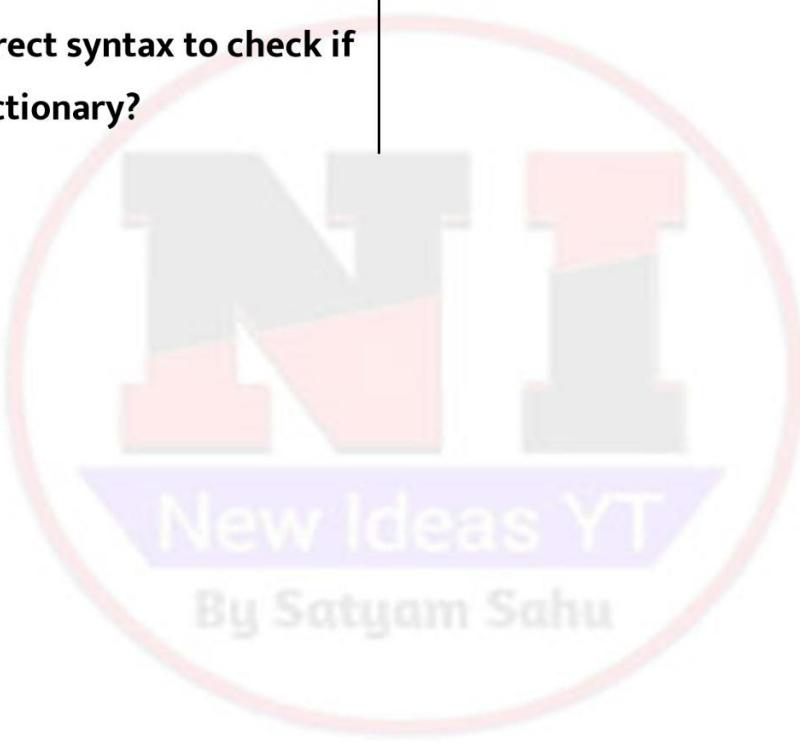
**Correct Answer: C) Returns a portion of the sequence**

- A) if key in dict:
- B) if dict.has(key):
- C) if key.exists(dict):
- D) if key == dict:

**Correct Answer: A) if key in dict:**

---

**10. What is the correct syntax to check if a key exists in a dictionary?**





## Chapter 6

### Functions in Python (Python में फ़ंक्शन)

#### 1. Top-down approach of problem solving (समस्या समाधान की टॉप-डाउन विधि)

A problem is broken into smaller sub-problems or modules which are solved independently and then integrated.

एक बड़ी समस्या को छोटे-छोटे हिस्सों (modules) में बाँट कर हल किया जाता है और फिर उन्हें एक साथ मिलाया जाता है।

#### Example:

To calculate the area of a rectangle:

- Step 1: Take input
- Step 2: Multiply length and breadth
- Step 3: Display result

#### 2. Modular programming and functions (मॉड्यूलर प्रोग्रामिंग और फ़ंक्शन)

Code is divided into independent units (functions). Each function does a specific task and can be reused.

कोड को अलग-अलग हिस्सों (functions) में बाँटा जाता है, जिससे उन्हें बार-बार उपयोग किया जा सके।

#### Example:



```
def add(a, b):  
    return a + b
```

---

### 3. Function parameters, Local variables (पैरामीटर और स्थानीय वेरिएबल्स)

Parameters are inputs to functions. Local variables are declared inside the function and accessible only inside it.

पैरामीटर वो मान होते हैं जो फ़ंक्शन को दिए जाते हैं। लोकल वेरिएबल्स फ़ंक्शन के अंदर बनाए जाते हैं और वहीं तक सीमित रहते हैं।

#### Example:

```
def greet(name):  
    msg = "Hello " + name # msg is a local variable  
    return msg
```

### 4. The return statement (वापसी स्टेटमेंट)

return sends back a value from the function to the caller.

return स्टेटमेंट के माध्यम से फ़ंक्शन एक मान को वापस भेजता है।

#### Example:

```
def square(n):  
    return n * n
```

---

### 5. Doc Strings (डॉक्युमेटेशन स्ट्रिंग्स)



Docstrings describe what the function does. They are written as the first statement in the function using triple quotes.

डॉकस्ट्रिंग्स से हमें पता चलता है कि फ़ंक्शन क्या करता है। इसे ट्रिपल कोट्स के बीच लिखा जाता है।

#### **Example:**

```
def add(a, b):
    """This function adds two numbers"""
    return a + b
```

#### **6. Global statement (वैश्विक वेरिएबल्स का उपयोग)**

Use the global keyword to modify a variable defined outside the function.

फ़ंक्शन के बाहर बने वेरिएबल को बदलने के लिए global कीवर्ड का उपयोग किया जाता है।

#### **Example:**

```
x = 10
def change():
    global x
    x = 20
```

#### **7. Default argument values (डिफॉल्ट मान)**

If a parameter is not passed, the default value is used.

यदि कोई मान नहीं दिया गया हो, तो डिफॉल्ट मान का उपयोग होता है।

#### **Example:**



python

```
def greet(name="User"):
    print("Hello", name)
```

## 8. Keyword arguments (की-वर्ड आर्ग्युमेंट्स)

You can pass arguments by name.

आप पैरामीटर को नाम से पास कर सकते हैं।

**Example:**

```
def student(name, age):
    print(name, age)
```

```
student(age=18, name="Rahul")
```

## 9. VarArgs Parameters (Variable arguments - एक से ज्यादा मान)

Use \*args to accept a variable number of arguments.

\*args का उपयोग करके हम एक फ़ंक्शन में कई मान पास कर सकते हैं।

**Example:**

```
def total(*numbers):
    return sum(numbers)
```



## Library Functions (लाइब्रेरी फँक्शन्स)

### Input and Output:

```
name = input("Enter name: ")
age = eval(input("Enter age: "))
print("Hello", name)
```

### String Functions (स्ट्रिंग फँक्शन्स)

Function	Explanation	Explanation (Hindi)
<b>count()</b>	Count occurrences	किसी अक्षर की गिनती
<b>find()</b>	First index	पहली बार पोजीशन
<b>replace()</b>	Replace substrings	स्ट्रिंग बदलना
<b>strip()</b>	Remove whitespace	किनारों की स्पेस हटाना
<b>upper()</b>	Convert to uppercase	कैपिटल लेटर में बदलना

### Example:

```
s = "python programming"
print(s.upper()) # PYTHON PROGRAMMING
print(s.find("pro")) # 7
```

### String: Slicing, Membership, Pattern Matching

```
text = "Hello Python"
print(text[0:5]) # Hello
print('P' in text) # True
```



## Numeric Functions (संख्यात्मक फँक्शन्स)

```
import math  
print(max(2, 5, 1)) # 5  
print(pow(2, 3)) # 8  
print(round(3.67)) # 4  
print(math.sqrt(16)) # 4.0
```

## Random & Date-Time Functions

```
import random  
print(random.randint(1, 10)) # 1 to 10
```

```
import datetime  
now = datetime.datetime.now()  
print(now.strftime("%d-%m-%Y %H:%M"))
```

## Recursion (पुनरावृत्ति)

A function calling itself is recursion.

जब एक फँक्शन खुद को दोबारा कॉल करता है, तो इसे पुनरावृत्ति कहते हैं।

### Example:

```
def fact(n):  
    if n == 1:  
        return 1
```



return n \* fact(n - 1)

## One Liner

1. Function = reusable block of code  
फंक्शन = कोड का पुनः उपयोग योग्य खंड
2. Use def keyword to define functions  
def का उपयोग फंक्शन बनाने के लिए होता है
3. Parameters can be default, keyword, or variable-length  
पैरामीटर डिफॉल्ट, कीवर्ड, या वैरिएबल लेंथ हो सकते हैं
4. return sends a value back to the caller  
return मान को कॉलर को वापस भेजता है
5. Docstrings provide documentation inside functions  
डॉकस्ट्रिंग्स फंक्शन के भीतर विवरण देती हैं

## 7. Built-in Functions & Libraries (इनबिल्ट फंक्शन और लाइब्रेरी)

1. Common functions: input(), print(), eval()  
सामान्य फंक्शन: input(), print(), eval()
2. String methods: find(), replace(), upper(), etc.  
स्ट्रिंग मेथड्स: find(), replace(), upper() आदि
3. Math functions: pow(), round(), sqrt(), etc.  
गणितीय फंक्शन: pow(), round(), sqrt() आदि
4. random module provides random number functions  
random मॉड्यूल यादचिक संख्या फंक्शन देता है
5. Recursion is when a function calls itself  
रीकर्शन वह होता है जब फंक्शन स्वयं को कॉल करता है



## MCQ

### 1. What does def keyword do in Python?

Python में def कीर्वड़ क्या करता है?

- A) Defines a variable
- B) Deletes a variable
- C) Defines a function
- D) Declares a class

**Answer: C**

### 2. What is the default return type of a Python function if no return statement is used?

अगर कोई return स्टेमेंट नहीं है, तो डिफॉल्ट रिटर्न वैल्यू क्या होती है?

- A) 0
- B) Null
- C) None
- D) Error

**Answer: C**

### 3. Which method changes all characters to uppercase?

कौन सा method सभी अक्षरों को uppercase में बदलता है?

- A) upper()
- B) lower()
- C) capitalize()
- D) title()

**Answer: A**

### 4. What does \*args do in a function?

फ़ंक्शन में \*args क्या करता है?

- A) Accepts string
- B) Accepts one argument
- C) Accepts any number of arguments
- D) Throws error

**Answer: C**

### 5. Which function is used to find square root in Python?

Python में स्क्वायर रूट निकालने के लिए कौन सा फ़ंक्शन है?



- A) sqrt()
- B) root()
- C) power()
- D) square()

**Answer: A**

---

**6. Which keyword is used to refer to global variables inside a function?**

फँक्शन के अंदर वैश्विक वेरिएबल का उपयोग करने के लिए कौन सा कीवर्ड है?

- A) global
- B) extern
- C) main
- D) static

**Answer: A**

---

**7. What is the output of print('abc'.isdigit())?**

इसका आउटपुट क्या होगा? `print('abc'.isdigit())`

- A) True
- B) False
- C) abc

- D) Error

**Answer: B**

---

**8. What does eval("2+3\*4") return?**

`eval("2+3*4")` का क्या आउटपुट होगा?

- A) 20
- B) 14
- C) 10
- D) Error

**Answer: B**

**MCQ 9: What is the purpose of the return statement in a function?**

फँक्शन में `return` स्टेटमेंट का उद्देश्य क्या है?

- A) It stops the function. / यह फँक्शन को रोक देता है।
- B) It skips the function. / यह फँक्शन को छोड़ देता है।
- C) It returns a value to the caller. / यह कॉल करने वाले को मान लौटाता है।
- D) It prints output. / यह आउटपुट को प्रिंट करता है।

**Answer: C**

---



**MCQ 10: Which of the following is a correct function definition in Python?**

निम्न में से कौन सा Python में सही फ़ंक्शन डिफिनिशन है?

- A) function add(a, b):
- B) def add(a, b):
- C) def add[a, b]:
- D) add def(a, b):

**Answer: B**

**MCQ 11: What is the output of the following code?**

```
def test():
    return 5
print(test())
```

- A) 0
- B) 5
- C) None
- D) Error

**Answer: B**

**MCQ 12: Which string function is used to check if all characters are alphabets?**

कौन सा स्ट्रिंग फ़ंक्शन यह जांचता है कि सभी अक्षर वर्णमाला हैं या नहीं?

- A) isdigit()
- B) isalpha()
- C) isalnum()
- D) isspace()

**Answer: B**

**MCQ 13: Which of the following will create a function that takes any number of arguments?**

निम्नलिखित में से कौन सा एसा फ़ंक्शन बनाएगा जो कोई भी संख्या में आर्ग्युमेंट्स ले सकता है?

- A) def func(args):
- B) def func(\*args):
- C) def \*func(args):
- D) def func[args]:

**Answer: B**

**MCQ 14: What is the output of print("Python".upper())?**

print("Python".upper()) का आउटपुट क्या है?



- A) PYTHON
- B) python
- C) Python
- D) Error

**Answer: A**

---

**MCQ 15: Which function can evaluate string expressions in Python?**

Python में कौन सा फ़ंक्शन स्ट्रिंग एक्सप्रेशन को Evaluate कर सकता है?

- A) eval()
- B) exec()
- C) input()
- D) str()

**Answer: A**

---

**MCQ 16: What does the global keyword do?**

global कीवर्ड क्या करता है?

- A) Creates a local variable
- B) Deletes a variable
- C) Declares a variable as global inside a

- function
- D) Defines a constant

**Answer: C**

---

**MCQ 17: Which of the following functions returns the length of a string?**

निम्नलिखित में से कौन सा फ़ंक्शन स्ट्रिंग की लंबाई लौटाता है?

- A) len()
- B) length()
- C) size()
- D) count()

**Answer: A**

---

**MCQ 18: What is the use of pass statement in Python functions?**

Python में pass स्टेटमेंट का क्या उपयोग है?

- A) It exits the function.
- B) It does nothing; used as a placeholder.
- C) It passes a value.
- D) It skips an error.

**Answer: B**



## Chapter 7

### File Processing (फाइल प्रोसेसिंग)

#### Concept of Files (फाइल की संकल्पना):

A **file** is a named location on disk to store related data. It allows data to be stored permanently and retrieved when needed.

फाइल एक नामित स्थान होती है जहाँ संबंधित डेटा को स्थायी रूप से संग्रहित किया जाता है और आवश्यकता पड़ने पर पुनः प्राप्त किया जा सकता है।

#### File Opening and Closing (फाइल खोलना और बंद करना):

Python uses the `open()` function to open a file and `close()` to close it.

#### Syntax (सिंटैक्स):

```
file = open("filename.txt", "mode")
file.close()
```

#### Modes in File Opening (फाइल खोलने के मोड):

Mode	Meaning (अर्थ)
'r'	Read mode – पढ़ने के लिए
'w'	Write mode – लिखने के लिए (पुराना डेटा मिटा देता है)
'a'	Append mode – डेटा जोड़ने के लिए
'r+'	Read + Write – पढ़ने और लिखने के लिए
'b'	Binary mode – बाइनरी डेटा के लिए (e.g., images, videos)



## Reading from a File (फाइल से पढ़ना):

```
f = open("data.txt", "r")
content = f.read()
print(content)
f.close()
```

Other methods:

- `read()`: reads the entire file
- `readline()`: reads one line
- `readlines()`: reads all lines in a list

## Writing to a File (फाइल में लिखना):

```
f = open("data.txt", "w")
f.write("Hello, Python!")
f.close()
```

- `write()`: writes a single string
- `writelines()`: writes a list of strings

## File Cursor Management: `tell()` and `seek()`

- `tell()`: Returns the current cursor position in file
- `seek(offset)`: Changes the cursor position



Example:

```
f = open("data.txt", "r")
print(f.tell())
f.seek(0)
print(f.read())
f.close()
```

---

### Command Line Arguments (कमांड लाइन आर्ग्युमेंट्स):

Python allows passing arguments from the command line using the sys module.

```
import sys
print("Filename:", sys.argv[0])
print("Argument1:", sys.argv[1])
```

To run:

```
bash
python program.py arg1 arg2
```

---

### Examples (उदाहरण)

Example 1: Writing user input to a file

```
f = open("user.txt", "w")
name = input("Enter your name: ")
f.write("Name: " + name)
f.close()
```



## Example 2: Reading all lines and printing

```
f = open("user.txt", "r")
lines = f.readlines()
for line in lines:
    print(line)
f.close()
```

## One Liner

1. Open a file using `open()`  
`open()` से फाइल खोलते हैं
2. Modes: `r, w, a, b, x`  
मोड़स: `r, w, a, b, x`
3. Read with `read()`, `readline()`, `readlines()`  
पढ़ने के लिए `read()`, `readline()`, `readlines()`
4. Write with `write()`, `writelines()`  
लिखने के लिए `write()`, `writelines()`
5. Use `close()` to close the file  
`close()` से फाइल को बंद करें
6. Use `tell()` and `seek()` to manage file pointer  
`tell()` और `seek()` से फाइल प्वाइंटर नियंत्रित करें

## MCQ



**MCQ 1: Which function is used to open a file in Python?**

Python में फ़ाइल खोलने के लिए कौन सा फ़ंक्शन उपयोग किया जाता है?

- A) open()
- B) read()
- C) write()
- D) load()

**Answer: A**

**MCQ 2: What does the 'w' mode do while opening a file?**

जब फ़ाइल 'w' मोड में खोली जाती है तो वह क्या करता है?

- A) Appends content
- B) Only reads
- C) Deletes existing content and writes new
- D) Opens file in binary mode

**Answer: C**

**MCQ 3: Which function returns the current cursor position in a file?**

कौन सा फ़ंक्शन फ़ाइल में वर्तमान कर्सर पोजिशन लौटाता है?

- A) seek()
- B) read()
- C) tell()
- D) getpos()

**Answer: C**

**MCQ 4: What will readlines() return?**

readlines() क्या लौटाता है?

- A) A single line
- B) Entire file as string
- C) List of lines
- D) Error

**Answer: C**

**MCQ 5: Which module is used to handle command line arguments?**

कमांड लाइन आर्ग्युमेंट्स को संभालने के लिए कौन सा मॉड्यूल उपयोग किया जाता है?

- A) os
- B) cmd
- C) sys
- D) file

**Answer: C**

**MCQ 6: What happens if we try to read a file that doesn't exist?**

यदि हम ऐसी फ़ाइल को पढ़ने की कोशिश करें जो मौजूद नहीं है तो क्या होगा?

- A) It returns None
- B) It creates the file
- C) It raises an error (FileNotFoundException)
- D) It reads an empty string

**Answer: C**

**MCQ 7: What does the write() function do in Python file handling?**

Python फ़ाइल हैंडलिंग में write() फ़ंक्शन क्या करता है?

- A) Reads a file
- B) Writes content to a file
- C) Deletes content
- D) Appends content

**Answer: B**

**MCQ 8: What is the default mode when opening a file using open()?**

open() का उपयोग करते समय डिफॉल्ट मोड क्या होता है?

- A) write
- B) append
- C) read
- D) execute

**Answer: C**

**MCQ 9: Which function reads one line at a time from a file?**

कौन सा फ़ंक्शन फ़ाइल से एक बार में एक लाइन पढ़ता है?

- A) read()
- B) readline()
- C) readlines()
- D) fetchline()

**Answer: B**

**MCQ 10: What does seek(0) do in a file?**

फ़ाइल में seek(0) क्या करता है?

- A) Moves the cursor to the end
- B) Closes the file
- C) Moves cursor to the beginning



D) Deletes file contents

**Answer: C**

**MCQ 11: What will happen if we do not close the file after operations?**

अगर फ़ाइल ऑपरेशन के बाद हम उसे बंद नहीं करते तो क्या होगा?

- A) Nothing happens
- B) Data may not be saved properly
- C) File gets deleted
- D) File becomes read-only

**Answer: B**

**MCQ 12: Which of the following is used to write multiple lines to a file at once?**

निम्न में से कौन फ़ाइल में एक साथ कई लाइनें लिखने के लिए उपयोग किया जाता है?

- A) write()
- B) writelines()
- C) writeall()
- D) appendlines()

**Answer: B**

**MCQ 13: Which file mode allows both reading and writing without truncating the file?**

कौन सा फ़ाइल मोड फ़ाइल को बिना मिटाए पढ़ने और लिखने की अनुमति देता है?

- A) r+
- B) w+
- C) a+
- D) rb

**Answer: A**

**MCQ 14: How do you open a file for appending in text mode?**

टेक्स्ट मोड में डेटा जोड़ने के लिए फ़ाइल कैसे खोलते हैं?

- A) open("file.txt", "w")
- B) open("file.txt", "r")
- C) open("file.txt", "a")
- D) open("file.txt", "r+")

**Answer: C**

**MCQ 15: Which of these functions is used to close an open file?**



760 741 8817



New Ideas YT



www.examjila.com

इनमें से कौन सा फ़ंक्शन एक खुली फ़ाइल को बंद करने के लिए उपयोग किया जाता है?

- A) stop()
- B) close()

C) exit()

D) end()

**Answer: B**





## Chapter 8

### Scope of Objects and Names (ऑब्जेक्ट और नाम का स्कोप)

In Python, **scope** refers to the area in a program where a **variable or object is accessible**. It defines the **visibility** and **lifetime** of a variable.

Python में स्कोप यह निर्धारित करता है कि कोई वैरिएबल/ऑब्जेक्ट प्रोग्राम में कहाँ तक मान्य है (accessible)।

#### Types of Scopes (स्कोप के प्रकार):

Python follows the **LEGB Rule**, which stands for:

Scope	Description	
L	Local	फ़ंक्शन के अंदर डिफ़ाइन्ड वैरिएबल
E	Enclosing	नेस्टेड फ़ंक्शन के बाहरी फ़ंक्शन का स्कोप
G	Global	पूरे प्रोग्राम में मान्य वैरिएबल
B	Built-in	Python द्वारा डिफ़ाइन किए गए प्री-डिफ़ाइन्ड नाम

#### Example – Scope and LEGB Rule:

```
x = "Global"
```

```
def outer():
```

```
    x = "Enclosing"
```

```
    def inner():
```



```
x = "Local"  
print(x) # Output: Local  
inner()  
outer()
```

**Explanation:** Inner function accesses its own local x.

---

### Modules in Python (Python में मॉड्यूल):

A **module** is a file containing Python definitions and functions. It helps in **code reuse** and **modular programming**.

मॉड्यूल एक ऐसा Python फाइल होता है जिसमें फंक्शन और क्लास को रखा जाता है ताकि कोड को दोबारा उपयोग किया जा सके।

### Creating and Using a Module (मॉड्यूल बनाना और उपयोग करना):

#### 1. Create a module:

Save the following as mymodule.py

```
def greet(name):  
    return "Hello, " + name
```

#### 2. Use the module in another file:

```
import mymodule  
print(mymodule.greet("Amit"))
```

---



## Importing Modules (मॉड्यूल इम्पोर्ट करना):

Syntax	Meaning	अर्थ (Hindi)
<b>import module</b>	Import entire module	पूरे मॉड्यूल को इम्पोर्ट करना
<b>from module import func</b>	Import specific function	किसी एक फंक्शन को इम्पोर्ट करना
<b>import module as alias</b>	Use alias	नाम छोटा करके प्रयोग करना

## Reloading Modules (मॉड्यूल को रीलोड करना):

Sometimes, if you modify a module and want the changes without restarting the interpreter:

```
import importlib
importlib.reload(mymodule)
```

### Examples

#### Example 1 – Global and Local Scope

```
x = 5
```

```
def func():
    x = 10
    print("Inside function:", x)
```

```
func()
print("Outside function:", x)
```

#### Example 2 – Using Module



```
# math module  
import math  
print(math.sqrt(25)) # Output: 5.0
```

## One Liner

1. Scope levels follow LEGB: Local, Enclosed, Global, Built-in  
स्कोप के स्तर LEGB नियम का पालन करते हैं
2. global keyword is used to modify global variables  
global से वैश्विक वैरिएबल बदले जाते हैं
3. Use import to bring modules  
import का प्रयोग मॉड्यूल लाने के लिए होता है
4. from module import function imports specific items  
from module import function विशेष फ़ंक्शन लाता है

## MCQ

**MCQ 1: What does 'L' in LEGB rule stand for?**

LEGB नियम में 'L' का मतलब क्या है?

- A) List
- B) Local
- C) Library
- D) Loop

**Answer: B**

**MCQ 2: Which keyword is used to define a global variable inside a function?**

किस कीवर्ड का उपयोग फ़ंक्शन के अंदर ग्लोबल वैरिएबल को परिभाषित करने के लिए किया जाता है?

- A) global
- B) globalvar
- C) extern
- D) var

**Answer: A**

**MCQ 3: What is a module in Python?****Python में मॉड्यूल क्या होता है?**

- A) A file with random data
- B) A class
- C) A file containing Python definitions and functions
- D) A database

**Answer: C****MCQ 4: Which method is used to reload a module?****किस फ़ंक्शन का उपयोग मॉड्यूल को रीलोड करने के लिए किया जाता है?**

- A) reload()
- B) importlib.reload()
- C) reimport()
- D) refresh()

**Answer: B****MCQ 5: Which of the following is correct syntax to import only one function from a module?****निम्न में से कौन-सा एक फ़ंक्शन इम्पोर्ट करने का सही तरीका है?**

- A) import math.sqrt
- B) from math import sqrt
- C) math.import sqrt
- D) from sqrt import math

**Answer: B****MCQ 6: What will be the output of the following code?**

```
x = 100
def myfunc():
    x = 200
    print(x)
myfunc()
print(x)
```

**उपरोक्त कोड का आउटपुट क्या होगा?**

- A) 200, 200
- B) 100, 100
- C) 200, 100
- D) Error

**Answer: C****Explanation:** Local x is 200 inside function; Global x remains 100.



**MCQ 7: What is the purpose of the global keyword in Python?**

Python में global कीवर्ड का उपयोग क्यों किया जाता है?

- A) To create a new variable
- B) To declare a variable as global
- C) To define a function globally
- D) To end a function

**Answer: B**

**MCQ 8: What does the following code print?**

```
def outer():
    x = "outer"
    def inner():
        print(x)
    inner()
outer()
```

यह कोड क्या प्रिंट करेगा?

- A) outer
- B) inner
- C) Error
- D) None

**Answer: A**

**Explanation:** x is accessed from the enclosing scope.

**MCQ 9: How can you rename a module while importing?**

मॉड्यूल को इम्पोर्ट करते समय उसका नाम कैसे बदल सकते हैं?

- A) import module as newname
- B) from module import \*
- C) import module rename newname
- D) module import as newname

**Answer: A**

**MCQ 10: Which scope is searched last according to LEGB rule?**

LEGB नियम के अनुसार किस स्कोप को सबसे अंतिम में खोजा जाता है?

- A) Local
- B) Global
- C) Built-in
- D) Enclosing

**Answer: C**



**MCQ 11: Which of the following is NOT a valid way to import a module?**

निम्न में से कौन सा मॉड्यूल इम्पोर्ट करने का वैध तरीका नहीं है?

- A) import math
- B) from math import sqrt
- C) from math import \*
- D) import sqrt from math

**Answer: D**

**MCQ 12: Which function helps to check all built-in functions and variables?**

कौन-सा फ़ंक्शन सभी बिल्ट-इन फ़ंक्शन और वेरिएबल्स को दिखाता है?

- A) show()
- B) builtins()
- C) dir(builtins)
- D) info()

**Answer: C**

**MCQ 13: If two modules have functions with same name, how do you resolve it?**

यदि दो मॉड्यूल्स में एक ही नाम के फ़ंक्शन हों तो कैसे हल करें?

- A) Use only one
- B) Use alias during import
- C) Remove one function
- D) Rename the file

**Answer: B**

**MCQ 14: Which file extension is used for a Python module?**

Python मॉड्यूल के लिए कौन-सा फाइल एक्सटेंशन उपयोग किया जाता है?

- A) .txt
- B) .exe
- C) .py
- D) .mod

**Answer: C**

**MCQ 15: What does importlib.reload() do?**

importlib.reload() क्या करता है?

- A) Deletes the module
- B) Reloads the module to get recent



760 741 8817



New Ideas YT



www.examjila.com

- changes
- C) Shuts down the program

D) Creates a new module

**Answer: B**





## Chapter 9

### Introduction to NumPy ndarray (NumPy ndarray का परिचय)

**NumPy** (Numerical Python) is a powerful library in Python used for numerical and scientific computing.

The main feature of NumPy is the **ndarray (n-dimensional array)**, which allows efficient storage and manipulation of large datasets.

**NumPy** एक शक्तिशाली Python लाइब्रेरी है जो संख्यात्मक और वैज्ञानिक गणना (numerical and scientific computation) के लिए प्रयोग होती है।

इसका मुख्य डेटा टाइप **ndarray (n-dimensional array)** है जो बड़े डेटा को कुशलतापूर्वक संग्रहित और प्रोसेस करता है।

#### Key Features (मुख्य विशेषताएँ)

Feature	Description (English)	विवरण (हिंदी में)
<b>ndarray</b>	N-dimensional array object	N-आयामी ऐरे ऑब्जेक्ट
<b>Broadcasting</b>	Operations on arrays of different shapes	विभिन्न आकारों के ऐरे पर ऑपरेशन
<b>Vectorization</b>	Fast execution of operations	तेज ऑपरेशन निष्पादन
<b>Integration</b>	Easily integrates with C/C++ and other libraries	C/C++ और अन्य लाइब्रेरी के साथ आसानी से एकीकृत

#### NumPy Installation (इंस्टॉलेशन)

pip install numpy

#### Importing NumPy (नंपाई को इम्पोर्ट करना)



```
import numpy as np
```

### Creating NumPy Arrays (NumPy ऐरे बनाना):

```
import numpy as np
```

```
a = np.array([1, 2, 3])
```

```
print(a) # Output: [1 2 3]
```

This creates a 1-dimensional array from a Python list.

### Data Types in NumPy Arrays (NumPy ऐरे में डेटा टाइप):

NumPy arrays support multiple data types: int, float, complex, bool, etc.

#### Example:

```
a = np.array([1, 2, 3], dtype=float)  
print(a)
```

### Array Attributes (ऐरे के गुण):

Attribute	Description	विवरण
<b>ndim</b>	Number of dimensions	आयामों की संख्या
<b>shape</b>	Shape of array	ऐरे का आकार
<b>size</b>	Total number of elements	कुल एलिमेंट्स
<b>dtype</b>	Data type of elements	एलिमेंट का प्रकार

**Example:**

```
a = np.array([[1, 2], [3, 4]])
print(a.ndim) # 2
print(a.shape) # (2, 2)
```

---

**Array Creation Routines (ऐसे बनाने की विधियाँ):**

Function	Purpose	उपयोग
<b>np.array()</b>	From list/tuple	लिस्ट से ऐसे
<b>np.zeros()</b>	All zeros	सभी शून्य
<b>np.ones()</b>	All ones	सभी एक
<b>np.eye()</b>	Identity matrix	एकात्मक मैट्रिक्स
<b>np.full()</b>	Fill with constant	निश्चित मान से भरना

**Example:**

```
np.zeros((2, 2)) # [[0. 0.], [0. 0.]]
np.ones((2, 2)) # [[1. 1.], [1. 1.]]
```

---

**Arrays From Numerical Ranges (संख्यात्मक रेंज से ऐसे):**

Function	Description	विवरण
<b>np.arange(start, stop, step)</b>	Like range()	एक रेंज ऐसे
<b>np.linspace(start, stop, num)</b>	Evenly spaced values	समान दूरी के मान

**Example:**

```
np.arange(1, 10, 2) # [1 3 5 7 9]
```



```
np.linspace(0, 1, 5) # [0. 0.25 0.5 0.75 1.]
```

### Indexing and Slicing (इंडेक्सिंग और स्लाइसिंग):

You can access or modify array elements using indexing/slicing.

#### Example:

```
a = np.array([10, 20, 30, 40])  
print(a[2]) # 30  
print(a[1:3]) # [20 30]
```

### One Liner

1. NumPy provides fast, memory-efficient arrays  
नंपाई तेज़ और मेमोरी-कुशल ऐरे प्रदान करता है
2. ndarray is the central object in NumPy  
ndarray नंपाई का मुख्य ऑब्जेक्ट है
3. Array creation: array(), zeros(), ones(), eye()  
ऐरे बनाने के तरीके: array(), zeros(), ones(), eye()
4. Array operations include slicing, indexing, broadcasting  
ऐरे में ऑपरेशन्स में स्लाइसिंग, इंडेक्सिंग और ब्रॉडकास्टिंग शामिल हैं

### MCQ

**MCQ 1: Which library is used for numerical computations in Python?**

Python में संख्यात्मक गणनाओं के लिए कौन-सी लाइब्रेरी उपयोग होती है?

- A) Pandas
- B) NumPy
- C) Matplotlib



D) TensorFlow

**Answer: B**

**MCQ 2: What does ndarray.ndim return?**

ndarray.ndim क्या लौटाता है?

A) Data type

B) Shape of array

C) Number of dimensions

D) Number of rows

**Answer: C**

**MCQ 3: What is the output of np.ones((2, 2))?**

np.ones((2, 2)) का आउटपुट क्या होगा?

A) [[0, 0], [0, 0]]

B) [[1, 1], [1, 1]]

C) [[2, 2], [2, 2]]

D) Error

**Answer: B**

**MCQ 4: Which function is used to create a range of evenly spaced numbers?**

समान दूरी वाले अंकों का ऐरे बनाने के लिए कौन-सा फंक्शन प्रयोग होता है?

A) np.range()

B) np.linspace()

C) np.numbers()

D) np.random()

**Answer: B**

**MCQ 5: What does the following code output?**

```
a = np.array([10, 20, 30, 40])
```

```
print(a[1:3])
```

A) [10, 20, 30]

B) [20, 30]

C) [30, 40]

D) Error

**Answer: B**

**MCQ 6: Which attribute gives the shape of a NumPy array?**

NumPy ऐरे का आकार (shape) जानने के लिए कौन-सा attribute प्रयोग होता है?



- A) shapeof()
- B) size()
- C) shape
- D) form

**Answer: C**

**MCQ 7: What does np.zeros((3, 2)) create?**

**np.zeros((3, 2)) क्या बनाएगा?**

- A)  $3 \times 2$  matrix filled with 1s
- B)  $3 \times 2$  matrix filled with 0s
- C)  $2 \times 3$  matrix filled with 0s
- D) Array with random values

**Answer: B**

**MCQ 8: Which of the following is used to create an identity matrix in NumPy?**

**NumPy में identity matrix बनाने के लिए कौन-सा function प्रयोग किया जाता है?**

- A) np.identity()
- B) np.eye()
- C) np.id()

- D) np.ones()

**Answer: B**

**MCQ 9: What is the data type of np.array([1, 2, 3.0])?**

**np.array([1, 2, 3.0]) का डेटा टाइप क्या होगा?**

- A) int
- B) float
- C) mixed
- D) complex

**Answer: B**

*(Because one element is float, NumPy upcasts the entire array to float.)*

**MCQ 10: Which function returns evenly spaced values within a given interval?**

**किस function का उपयोग किसी दिये गए अंतराल में समान दूरी वाले मान उत्पन्न करने के लिए किया जाता है?**

- A) np.linspace()
- B) np.arange()
- C) np.sequence()



760 741 8817



New Ideas YT



www.examjila.com

D) np.series()

Answer: A





760 741 8817



New Ideas YT



www.examjila.com

इस ई-बुक को पढ़नेकेबाद अब आपको एक सही दिशा, सटीक कंटेंट और सशक्त माग

अब समय हैइस ज्ञान को एक्शन मेंबदलनेका।

अगर आप इस सफर को और गहराई सेसीखना चाहते हैं, तो आज ही हमारे **Paid Course** में शामिल हो जाइए—

जहाँ हर विषय को बसेसेलकर एजाम लेवल तक समझाया जाता है।

जुड़नेकेलिए:

- ❖ YouTube Channel: [NewIdeasYT](#)
- ❖ Telegram Group: <https://t.me/NEWIDEASYT>
- ❖ Batch Registration: [<https://examjila.com/courses/m3r5>](<https://examjila.com/courses/m3r5>)
- ❖ Free Pdf: <https://examjila.com/pdf-notes>

New Ideas YT

By Satyam Sahu

हमारा वादा है— इस बार O Level पास करना है!

आपकी सफलता, हमारी जिम्मेदारी

आपका माग

Satyam Sahu

(Founder & Educator – NewIdeasYT)

© Copyright All Rights Reserved.

Author: Sahu Satyam (Founder)