Day1 - Day4 Handwritten

1. What is Python?

↳ Python is a high-level, interpreted programming languages known for its simplicity and readability. It was created by Guido Van Rossum and first released in 1991.

2. Key features of python?

* Easy to learn & use
* Versatile
* Interpreted
* Dynamically typed
* Extensive libraries & frameworks
* Cross platform.

3. Is python ~~is~~ compiled or Interpreted?

Python is both compiled & Interpreted.

→ It is compiled to bytecode,

→ The bytecode is interpreted at runtime by PVM.

4. Main application of python?

→ Python is incredibly versatile & used in many areas. Here are some of its main application.

1. Web development.
2. Data scienc & Analytics
3. Machine Learning & AI
4. Automation & Scripting
5. Scientific Computing
6. Game development
7. Desktop GUI Applications
8. Networking & Cybersecurity

5. How to download python on your system?

For windows:

1. Download python:
   * Go to the official Python website:
     http://www.python.org/downloads/
   * Click on the lastest window installer

2. Run the installer:
   * Double click the downloaded file.
   * Important: Check the box "Add python 3.x to PATH" at the bottom.
   * Click Install now & follow the prompts.

3. Verify Installation:
   * Open command prompt (search for cmd)
   * Type python --version or pythor & press Enter.
   * You should see the installed python version.

6 Difference b/w Python 2 & Python 3?

→ The key difference b/w python 2 & Python 3 is that Python 3 is the future, while Python 2 is now obsolete (officially ended on January 1, 2020).

Feature	Python 2	Python 3
Print Statement	print "Hello"	print ("Hello") (function)
Integer Division	5/2 = 2 (truncates)	5/2 = 2.5 (true division)
Unicode support	Not default; use u-text	Strings are Unicode by default.
xrange()	Exists; more memory-efficient	Removed, range() behaves like xrange()
Library support	Most new libraries dont support it	Full support for all modern libraries.
End of life	Support ended in 2020	Actively maintained & updated.

7. How can you check the installed python version?

* You can check the installed python version from your terminal or command prompt using the following commands, depending on your system setup.

→ python -- version (or) python -V

8. What is the role of print() function?

print() function is used to display output to the console. It is one of the most basic & commonly used functions to show information, debug programs or interact with users.

9. What is an IDE?

* An IDE (Integrated Development Environment) is a software application that provides comprehensive facilities to programmers for software development.

Commonly used Python IDEs:

1. Pycharm
2. Visual Studio Code
3. Spyder
4. Jupyter Notebook.
5. IDLE.

10. How do you run python from the terminal?

1. Open your terminal.
   * On windows: Use command prompt or PowerShell
   * On MacOS/Linux: Use Terminal.
2. Navigate to the folder where your Python file is saved.

   cd /path/to/your/folder.

3. Run the python file using one of the following commands:

* python3 filename.py (If using python3)

  (or)

  python filename.py

  (or)

  py filename.py (Use only Python launcher is installed)

Variable in Python (11-20).

11. What is a variable in Python?

* A variable in Python is a named storage location used to hold a value that your program can use or change later.

12. How do you declare a variable in python?

You declare a variable simply by assigning a value to a name using the = operator. No need to specify the type.

Syntax: variable_name = value.

13. Is it necessary to declare the type of a variable in python?

* No, it is not necessary to declare the type of a vairable.
* Python is dynamically typed language, which means
  → You do not need to specify a variable's data type when declaring it.
  → Python automatically infers the type based on the value you assign.

```
name = "Alice"  # Python knows this is a string.
```

14. What are the rules for naming a Variable?

1. Start with a letter or underscore (_)
   * Valid : name, _name
   * Invalid : 1name, @value.
2. Can contain letters, numbers, & underscores.
   * Valid : user1, total_amount, max_Speed
   * Invalid : user-name, price$
3. Case-Sensitive :
   * score, Score, SCORE are different variables.
4. Cannot be a Python keyword.
   * Invalid : if, class, for, def, True etc.

15. Global variable & local variable differences?

Global variable	Local variable

* Declared outside of all functions
* Can be accessed anywhere in the program
* To modify them inside a function, you must use the global keyword.
`x = 10 # global variable`
`def update():`
`  global x`
`  x = 20 # modifies global x`
`update()`
`print(x) # o/p : 20.`
	* Declared inside a function
	* Can only be used within the function.
	* They are created when the function is called & destroyed when its ends.
	`def greet():`
	`  message = "Hello" # local variable`
	`  print(message)`
	`greet()`
	`# print(message) # Error!`

16. Can a Variable start with a number in Python? Why or Whynot?

* No, a variable name cannot start with a number in Python.

```
1st_place : "Anu"  # Syntax Error.
```

Correct Way:

```
first_place : "Anu"
```

Why not?

* Python follows strict naming rules for identifiers (like variable names) and one of those rule is

→ A variable must begin with a letter (A-Z or a-z) or an underscore (_), not a digit.

→ This rule helps Python avoid confusion with numeric literals and maintain clear, predictable syntax.

17. What happens if you use a variable w/o assigning a value?

* If you use a variable w/o assigning a value you will get a NameError, beoz python doesn't know what the variable refers to.

Eg: `print (score)  # NameError: name 'score' is not defined`

Correct Usage:

```
score : 100
print (score)
```

18. How is memory managed for variables in Python?

* Memory management for variables in Python is automatic. But it's still structured and efficient under the hood.

19. Can Python Variable names contain special characters?

* No, Python variable names cannot contain special characters like $, @, #, %, or !

* It allows only,
  - → Letters (A-Z or a-z)
  - → Digits (0-9) But not at the beginning
  - → Underscore (_)

* Not allowed, because will cause a SyntaxError.

20. What is the difference b/w = & == operator?

= (Single Equals) Assignment	== (Double Equals) Comparison
* Used to assign a value to a variable.	* Used to compare two values for equality.
* Stores data in a variable.	* Return True if values are equal, otherwise False
x = 10 ~~Prin~~name = "Anu"	x = 5 print (x == 5) # True print (x == 10) # False.

Data Types in Python (21-30)

21. Basic Data Types in Python?

In Python, there are several built in data types that that you use to store different kinds of data.

1. Numeric Types.

* int	- Integers	- 1, 2
* float	- Decimal numbers	- 3.14, 2.5
* complex	- Complex numbers	- 2+3j

2. Text type:

Str	- String (Text)	- "Hello"

3. Boolean Type:

bool	- Logical True/ False	- True, False.

4. Sequence Type:

list	- Ordered, mutable.	[1,2,3] ['a', 'b']
tuple	- Ordered, immutable.	(10,20), ('a', 'b')
range	- Sequence of numbers.	range(5) (0 to 4)

5. Set type:

set	- Unordered, Unique values	{1,2,3}
frozenset	- Immutable set	frozenset({1,2,3})

6. Mapping Type:

dict . key value Pairs {"name": "Anu", "age" : 20}

7. None Type:

None Type Represents no value None.

22. Difference b/w int, float, complex?

int (Integer)	float (Floating point)	Complex (Complex number
* Represents whole number (no decimal point) * Can be +ve, -ve or 0. * Eg: 5, -12, 0.	* Represents decimal numbers. * Used when precision is needed * Eg: 3.14, -0.5	* Represents a number with real & imaginary part Eg: a+bj (j is the imaginary unit).

23 What is the difference b/w list & tuple?

List	Tuple.
* Mutable: You can add, remove or change. * created using square brackets [ ] * Slower, but more flexible. fruits = ['apple', 'banana', 'cherry'] fruits[1] = 'orange'	* Immutable: Once created, cannot be changed. * Created using parantheses ( ) * Faster & use less memory. Colors = ('red', 'green', 'blue') Colors[1] = 'Yellow' # Type Error

24. Difference b/w dictionary & list?

List	Dictionary.
* Sequence of ordered items * Accessed using index number starts from 0 * Created using square brackets [] fruits = ['Apple', 'banana', 'Cherry']	* Collection of key value pairs * Accessed using keys, not index numbers. * Created using curly braces {} person = {"name": "Anu", "age": 25}

25. What is a set? How is it different from list?

* A set is collection of unordered, unique elements

set	List
* Unordered: Items have no index or order. * No duplicates allowed. * created using curly braces {} or set() function. my_set = {1,2,3,2,1} print(my_set) # O/P {1,2,3} Duplicates are removed	* Ordered: Items are stored in a specific order * Duplicates are allowed * created using square brackets. my_list = [1,2,3,2,1] print(my_list) #O/P [1,2,3,2,1]

26. Difference b/w Mutable & Immutable data types?

Mutable	Immutable
* Can be change after they are created.	* Cannot be changed after creation.
* You can add, remove or modify elements	* If you try to change them, Python creates a new object
Eg: list, dict, set.	Eg: int, float, str, tuple, bool.

27. What will type() function return if the variable is String?

* If the variable is String, the type() function will return <class 'str'>

Eg: name = "Anu"
print (type(name)) # o/p <class 'str'>

28 What are the Boolean data types?

* Boolean (or) bool is a basic (built-in) data type in python that can hold only two possible values.
  -> True,
  -> False.

29. How do you covert data from one type to another?

* In Python, you can convert data from one type to another using type conversion functions also known as type casting.

Type conversion formulas:

Function	Converts to	Example.
int()	Integer	int("5") → 5
float()	Float	float("3.5") → 3.5
str()	String	str(100) → "100"
bool	Boolean	bool(0) → False.
list	List	list("abc") → ['a', 'b', 'c']
tuple	Tuple	tuple([1,2]) → (1,2)
set	Set	set([1,2,2]) → {1, 2}

30. What does the len() function do for different data types?

* The len() function return the number of items (length) in a container-type object such as strings, list, dictionaries etc..

Eg: len("hello") → 5

len([1,2,3]) → 3