Task - 12/06/'25

FEBRUARY 2025

01 Saturday

Week 05
032-333

1. History of python programming language

Guido Van Rossum - Creator of python programming language.

Python is a high-level, interpreted, general-purpose programming language.

Why it is called `python'?

Guido named it after his favourite comedy sketch series from an old BBC television called "Monty python's flying circus".

Timeline of Python's history:

1989 - (Birth of python)
Guido started working on python during Christmas Holidays in December 1989.

02 Sunday

1991 - First official release (0.9.0 Released)

features - functions, exceptions, modules, strings, lists and basic I/O.

FEB 2025	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	*	*	*

1994 - Python 1.0 (with more features like lambda, map, filter, reduce).

2000 - Python 2.0 (List comprehension, Garbage collection)

2008 - Python 3.0 (Improved Unicode support and simplified syntax).

2020 - Python 2 ended (officially discontinued on Jan 1, 2020.

present - (Python 3.x) 3.10, 3.11, 3.12, 3.13, 3.14 & ongoing.
Continues to be popular in web development, data science, AI, ML, automation & education.

Python Today:
- One of the most popular programming lang.
- Used by tech giants like Google, YouTube, Instagram, NASA, Dropbox & more.

NOTE: Anaconda, not an IDE itself, but a python distribution. It includes IDE like Spyder & Jupyter

NOTE Notepad not an IDE. It is a plain text editor eg. no debugging tools

FEBRUARY 2025

04 Tuesday Week 06 035-330

2. IDE in Python.

IDE - Integrated Development Environment.

It is a software application that provides tools to write, test and debug python programs easily.

Main features of an IDE:

Features	Purpose
1. Code Editor	Write or edit python code
2. Syntax Highlighting	Colors for keywords, variables
3. Auto-completion	Suggest code while typing
4. Debugger	Helps to find & fix errors
5. Output Console	Shows program output
6. File Management	Manage & save multiple files.

Popular python IDE's:

IDE Name	Description
1. IDLE	Default IDE
2. Pycharm	Powerful & feature-rich (JetBrains)
3. VS code	Light weight editor with extension
4. Jupyter Notebook	Best for Data Science & ML
5. Thonny	Simple IDE for beggners
6. Spyder	Popular in scientific computing.

FEB 2025

S	M	T	W	T	F	S
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28

(IDE helps you write better python code faster with less error, esp. useful for beginners & Professionals alike

3. TOKENS in Python

- In Python, tokens are the smallest units of a program that have meaning.
- It is a basic building blocks of python code.

Types of Tokens:

1. Keywords:
   - These are reserved words that have special meaning in python.
   - Cannot use them as variable names.
   - lower case except True, False.

if, else, elif, while, for, break, continue, def, return, import, class, True, False, None

2. Identifiers:

These are names given to variables, functions, classes.

Rules:

* Must start with a letter (A-Z) or (a-z) or (_) underscore
* Cannot start with a number or symbols (except _ underscore)
* Cannot used reserved keyword.
* Can contain letters, numbers and underscore.

3. Literals : Constant values assigned to variables.

Types:

Literal Type	Example
String	"hello", 'hi'
Integer	10, -5
Float	3.14, -2.5
Boolean	True, False
NONe	None.

Numbers: int, float, complex

Collections: list, tuple, set, dictionary

4. Operators : Symbols that perform operations on variables or values.

Type	Example	Description
Arithmetic	+, -, *, /, //, %, **	Math operations
Assignment	=, +=, -= etc.	Assign values
Comparison	==, !=, >, <	Compare values
Logical	and, or, not	Logic operations
Bit wise	&, |	
Membership	in, not in	Check membership
Identity	is, is not	Check memory identity

doubt

4. Data Types in Python

Data types define the kind of value a variable holds.

Built-in Data Types in Python

1. Numeric Types

Type	Example	Description
int	10	Integer numbers.
float	3.14	Decimal numbers
complex	2+3j	Complex numbers

2. Text Type

str	"hello"	Text string

3. Boolean Type

bool	True, False	Represents truth values.

4. Sequence Types

list	[1, 2, 3]	Ordered, changeable, allows duplicates
tuple	(1, 2, 3)	Ordered, unchangeable
range	range (5)	Sequence of numbers.

5. set Types

Type	Example	Description
set	{1, 2, 3}	Unordered, no duplicate values.

6. Mapping Type

dict	{"name": "xxx", "age": 25}	Key - value pairs.

7. None Type

None Type	None	Represents no value or null value.

5. Membership Operators

- Used to test whether a value is present in a sequence
- Returns boolean value (True or False)

Operator	Meaning	Example
in	Returns True if value is present	1. 'a' in "apple" - True 2. 'z' in apple - False
not in	Returns True if value is not present	1. 'a' not in apple - False 2. 'z' not in apple - True

09 Sunday

6. Identity Operators

- Identity operators are used to compare the memory locations of two objects.
(ie) Whether two variables refer to the same object.
(returns boolean)

Operator	Meaning	Example
is	Returns True if both variables point to the same object in memory	a is b (a & b refer to the same object)
is not	Returns True if they do not refer to the same object	a is not b (a & b refer to different objects)

7. How to find data type? Explain.

Every value and every variable has a data type. The type() function tells you the class (type) of any value or variable.

Examples:

Data	Code Example	output
10	type(10)	<class 'int'>
"hello"	type("hello")	<class 'str'>
[1,2,3]	type([1,2,3])	<class 'list'>

11 Tuesday 8. Convert a='154' to integer. Week 07 042-323

To convert a string to an integer, use int() function.

```
a = '154'          # This is a string (coz it is in quotes)
b = int(a)         # convert string to integer
print(b)           # output 154
print(type(b))     # <class 'int'>
```

9. Output:

hello hello hello hello hi hi hi hi

```
print("hello" * 4 + hi * 4)
```

output coming without space.

10. Write Programme for the output:

Name : RC Total : 453
Age : 27 Average : 90.6
Tamil : 88
English: 93
Math : 100
Science : 82
Social : 90

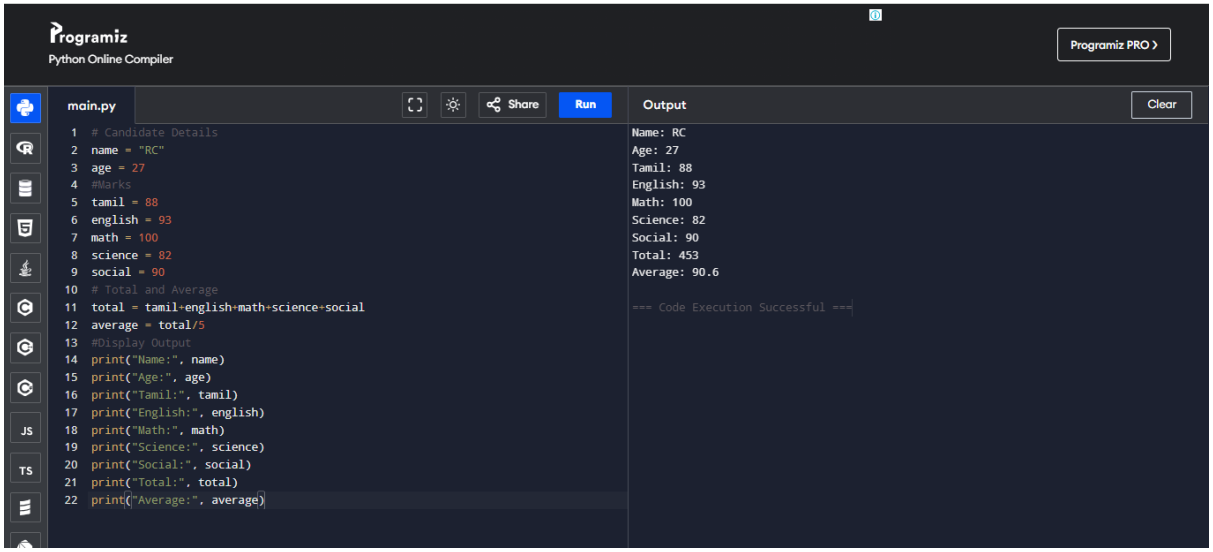
```
# Candidate details
name = "RC"
age = 27

# Marks
tamil = 88
english = 93
math = 100
science = 82
social = 90

# Total and Average
total = tamil + english + math + science + social
average = total / 5

# Display output
print ("Name :", name)
print ("Age :", age)
print ("Tamil:", tamil)
print ("English:", english)
print ("Science :", science)
print ("Social:", social)
print ("Total:", total)
Print ("Average :", average)
```

Programme code



String to integer