

1/10/22

Page No.	
Date	

* Programming Languages *

→ What is Programming ?

- It is the process of the converting ideas into instructions that computer can understand and execute. These instructions are specific and sequential.

eg: consider a tea which requires different ingredients like sugar, water, tea leaves, ginger which is put in sequential and specific amount.

→ Bug - When something unexpected happens is called bug.

→ Crash - When programs or softwares stop working or freeze.

→ "Orders and Sequence matters in the programming."

→ Programming can also be defined as giving computers the right instructions and the correct sequence of steps to produce the desired result.

→ There are tons and thousands of programming languages and each comes with its own set of strengths and weaknesses.

→ Computers only understands '0 & 1' i.e binary code / language and all the code we write in english is converted to machine language so that computers can understand our instructions.

→ Source Code - It is a code written by programmers for the softwares and products.

→ There are 3 methods to convert source code into machine code

1) Compile it 2) Interpret it 3) Both

1) Compiler:- It takes high level programming languages and turn it into an executable that contains low level machine code.

2) Interpreter:- The interpreter process out source code each times its run - line by line. The main difference b/w interpreter & compiler is that the interpreter can execute every line of source code each time.

⇒ Compiled Languages:- C, C++ and Objective-C

⇒ Interpreted Languages:- Php & JavaScript

⇒ Combination:- Java, C# & Python

→ IDE's (Integrated Development Environment) is used to write source code. - It is an application that provides the special tools needed to write, debug and compile code.

* Python

→ Syntax:- It is a rules of a programming language to be followed.

→ Statements are the foundations of any program and source code.

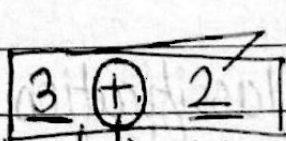
Each statements can be made up of:

Keywords + Expressions + Operators

Operators are the statements that tell the computers to perform an action with some input.

Symbol	Operators
+	Addition
-	Subtraction
*	Multiplication
/	Division

↳ This are called Arithmetic Operators

eg:  Input/operands

Arithmetic operator

↳ This is an expression

eg: $2 + 3 * 6 \neq 30$

$= 20$

→ Because all the arithmetic operators follow the rule of (BODMAS)

⇒ Errors and Bugs are very common in programming

→ Errors Categories :-

- 1) Syntax :- Languages rules broken
- 2) Runtime :- The computer was unable to execute
- 3) Semantic :- The output of the program is not what you have expected

→ Variables - It is any characteristics value like number or characters

Variables in Python can be declared by giving them a name and assigning the value and the assignment of the value is done by the assignment operator which is "=" (equal to)

eg: $\text{age} = 36$
 ↓ ↗
 variable assignment

⇒ Data Types :- It is the classification of data items which represents the kind of value that tells what operations can be performed on a particular data

→ Integer :- It is a number with no decimal values.

→ Strings :- It is a combination of letters, numbers, symbols and spaces.

eg: age = 36 → integer
print(age)

email = jk@gmail.com → string
print(email)

→ To identify which data-type a function is we use type operator.

eg: print(type(age)) & print(type(email))
= int = str

→ Rules for variables:-

1) The variable name should only contain letters, numbers and underscores. Also the name shouldn't start with numbers.

2) Spaces are not allowed in variable name
eg: account balance (X)
account_balance (✓)

3) The variable names are case-sensitive
eg: Cookies ≠ cookies

4) Variables cannot be reserved keywords.

→ Arithmetic Operators

3 + 2 = 5 → addition

3 - 2 = 1 → subtraction

3 / 2 = 1.5 → division

3 * 2 = 6 → multiplication

$3 ** 2 = 3^2 = 9 \rightarrow$ exponent

$3 \% 2 = 1 \rightarrow$ modulus (remainder)

$3 // 2 = 1 \rightarrow$ integer division

→ Float:- Any number with decimal points.

eg: 3.45, 7.42

→ Strings: `print("Hello World")`
(strings are put up in " " or ' ' marks)

eg: `print("Let's get started")` (X)

`print('Let's get started')` (✓)

OR

`print("She said, 'Fantastic' to him")` (X)

`print('She said, "Fantastic" to him')` ✓

→ Whitespace:- It is a space given in program to make it human readable and make the program free from errors.

→ Comments:- It is used to make the every line of code in a understandable manner.

Syntax:- `# This is a comment`

→ Problem 1

Code:- `print("Challenge 1:")`

```
# A message for user  
message = "This is going to be very tricky"  
Message = "Very tricky!"  
print(message) # show the message
```

```
# Perform mathematical operation  
result = 2 * 3  
print("2 * 3 =", result)  
result = 5 - 3  
# print("5 - 3 =", result)  
print("Challenge completed!")
```

Soln.

① | This is going to be ~~very~~ tricky | message

② | Very Tricky | Message

`print(1)`

③ | `2 * 3 = 8` | result

`print(↓)` ⇒ `2 * 3 = 8`

④ | `2` | result is now updated to