Why we create Program?

To build Software. Software is a collection of Program which achieve common objective.

SDLC

Software Development Life Cycle

1) Requirement Gathering

2) Analysis

3) Designing ( ER Diagram , DFD Diagram , UML Diagram )

4) Development or Coding

5) Testing

6) Deployment and Training

Language Basics

All Language Basics are same

1) Variable , Operator and constant

2) Decision Construct like if else

3) Looping Construct

Organizational Concept

In Software, there is so many lines of code so we need to organize this code. Every language have a standard approach to organize code.

1) Functional Approach

2) OOPS Approach

Python Advantage

Its Learning Cycle is short.

Sound in Syntax Rule

You have to memorize Syntax Rule of Python Language and for achieving you have to repeat it again and again.

Step to Create Program

Fact :- Computer can solve only mathematical problem. All mathematical problem have pre-defined way to solve it.

Step 1 :- Find the Programming Problem Solution into mathematical context

Step 2 :- Convert this mathematical solution into selected programming language

Tip :- Always try different programming problem

Python

Python has 30 keyword.

Python is a case sensitive language

Python is a purely OOPS language. Here each and every term is a First Class Object.

Why OOPS?

In a Software, there is so many lines of code. Then we need some way to organize code. In C , we used Functional Approach to manage code.

Functional Approach focus upon Action So called Action Oriented Approach. It follow Top Down Approach to manage code.

Software Design of Functional Approach is tightly couple.

Coupling :- interdependency between modules

Cohesion :- specificness of a module

Tight Coupling increases Software Complexity.

To share data between function there is concept of Global Data and this global data cannot be restricted to specific function which raises the issue of Data Security.

Functional Approach is best if functionality of Software are fixed like Hardware Related Software.

Functional Approach is not suited if functionality change are volatile like in real life software.

OOPS

It is a way to organize code.

Object Oriented Programming System :- It focus upon Object. It is a real world approach.

Object :- Any real thing which is capable to show its presence to us.

Well Defined Object :- Object which contain given four feature

1) State :- property of an object which can be change by some action

2) Attribute :- property of an object whose value help us describing an Object

3) Behavior :- action performed by object to change of its state

4) Identity :- attribute whose value are unique among different object

Object can communicate with each other.

Sender Object --> Message Pass --> Receiver Object --> Perform an Action called Method.

Behavior :- action controlled by object itself

Method :- action controlled by another object

Class :- It is a conceptual representation of same state, attribute, behavior and method of similar kind of object.

Class is a blue print of Object

Class is a model of Object

Object is a reality of Class

Object is an instance of Class

In OOPS, we search Object and classify them into Class.

Object focus upon data and their operation.

Feature of OOPS :-

1) Encapsulation :- process of hiding unnecessary details from an end user with the help of logical or physical packaging

2) Abstraction :- process of retrieving necessary details for an end user

3) Inheritance :- It is a back bone of OOPS. Child derive feature from their parent. Term used in Inheritance.

Super - Sub Class , Parent - Child Class , Base - Derived Class

Generalization :- process of analyzing common feature between classes and build their parent classes according to common features. Due to this, OOPS follow Bottom Up Approach.

Benefits :-

i) Reusability

ii) Ease of Maintenance

iii) Resilience to Change or Flexible to Change

After performing Generalization, we achieve a top most base class which is sole purpose to act as parent for another classes. Such class does not exists in reality. This class is called Abstract Class.

Child is capable to redefine feature provided by its Parent. This feature is called OVERRIDING.

4) Polymorphism :- Poly ( Many ) + Morphs ( Form ) An Object can exists in many forms. There are two type of Polymorphism

i) Static Polymorphism :- An object form depend upon its context.

ii) Dynamic Polymorphism :- Action of an Object depend upon its value.

Python

Python is a case sensitive language

All Python program have extension py

Python is an interpreted language

print()

This is a pre-defined function in Python which accept a String and display it on screen and move the cursor to next line

Any term enclose in double quote or single quote is called String

Escape Character or Escape Sequence

They begin with backslash ( \ ) and contain special meaning in Python

|  |  |
| --- | --- |
| Escape Character | Meaning |
| \n | New Line |
| \t | Tab |
| \\ | \ |
| \" | " |
| \' | ' |
| \b | Backspace |
| \f | Form Feed |
| \r | Carriage Return Move the cursor at beginning of line |
| \newline | Used to split String in multiple Line |
| \a | Bell Sound |
| \v | Vertical Tab |
| \ooo | Represent Octal Number |
| \xhh | Represent HExa Decimal Number |
| \u0000 | Unicode Character |

Character Encoding

Computer cannot directly understand character then we need some process which convert character into computer understandable form. This process is called Character Encoding. There are two type of Character Encoding present

1) ASCII character take 8 bit in memory and it represent English language symbol.

2) Unicode character take 16 bit in memory and it represent all language symbol.

Variable

It is a memory location which store the constant value and this location is refer by a valid identifier and called variable name.

Identifier :- name given to identify element in Python language.

There is no concept of declaration of variable. Here in Python, concept of Dynamic Type Initialization which decide the type of variable.

Type of Value :-

1) str :- represent String

2) int :- number without decimal point

3) float :- number with decimal point

4) bool :- represent by True and False

Operator :- It is symbolic function.

Assignment Operator :- It assign right side value to its left

In Python, there is a type() function used to get type of a variable

Arithmetic Operator :- + , - , \* , / (divison) , // ( floor division ) , % , \*\* ( exponent )

How to read value from keyboard

In Python, input() function is used to read value from keyboard and it read value in form of String. So here is requirement of conversion function

1) int(string) :- convert string to int

2) float(string) :- convert string to float

3) bool(string) :- convert string to bool

4) str(value) :- convert any type value into string

Number System

1) Binary Number System :- Base 2 Allowed Digit 0 and 1

2) Decimal Number System :- Base 10 Allowed Digit 0 to 9

3) Octal Number System :- Base 8 Allowed Digit 0 to 7

4) Hexa Decimal Number System :- Base 16 Allowed Digit 0 to 9 , A , B, C, D, E,F

15687

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 5 | 6 | 8 | 7 |
| 10000th Place | 1000th Place | 100th Place | 10th Place | Unit Place |

15687 = 1 \* 10000 + 5 \* 1000 + 6 \* 100 + 8 \* 10 + 7 \* 1

15687 = 1 \* 104 + 5 \* 103 + 6 \* 102 + 8 \* 101 + 7 \* 100

15687 // 10 = 1568 ( It remove the last digit of given number )

15687 % 10 = 7 ( It give us last digit of given number )

Internal of Python

In Python, each and every value or term will be treated as an object. Every object have a unique id and we get this id with the help of id() function.

In Python, data value are immutable which means we cannot change their value. If there are some change it will be done by creating new object.

In Python, int , float , str and bool are immutable data type.

That process takes background resources of Python and degrade performance.

Arithmetic assignment Operator

+= , -= , \*= , //= , /= , %= , \*\*=

a \*\*= 3

a = a \*\* 3

Relational Operator or Comparison Operator

> , < , >= , <= , != , ==

They used to compare value and return result in boolean form.

Decision Construct

Normally Python execute its program statement in sequential manner. We can control the execution of statement based upon given condition with the help of decision construct.

1) if statement :- In Python, there is no block statement. We use indentation for creating blocks.

if boolean\_expression :  
 if\_block\_statement

If boolean expression gives True it execute if block statement

2) if else statement :-

if boolean\_expression :  
 if\_block\_statement  
else :  
 else block statement

If boolean expression gives False it execute else block statement

else block never exists without if block

3) Nested if else statement :- we can define if block or if else block inside another else block or if block

4) elif statement :- If we found only if block or if else block inside another else block then we can use elif statement

Python Comment

These are line in code which is ignored by Python Interpreter and there are two type of comment

1) Single Line Comment  
 # First Line Comment

2) Multi Line Comment

''' First Line  
 Second Line '''

Logical Operator

and

|  |  |  |  |
| --- | --- | --- | --- |
| A > 10 | and | B > 10 | Result |
| True |  | True | True |
| True |  | False | False |
| False |  | True | False |
| False |  | False | False |

If first condition gives False then it never check second condition and return False as a result. This behavior make it Short Circuit Logical Operator

or

|  |  |  |  |
| --- | --- | --- | --- |
| A > 10 | or | B > 10 | Result |
| True |  | True | True |
| True |  | False | True |
| False |  | True | True |
| False |  | False | False |

If first condition gives True then it never check second condition and return True as a result. This behavior make it Short Circuit Logical Operator

not

|  |  |
| --- | --- |
| not A > 10 | Result |
| not True | False |
| not False | True |

Loop

It is an arrangement of repeating some statement more than once. Loop focus upon repetition or Iteration. There are two type of Loop.

1) Fixed Loop :- Where number of repetition or iteration is controlled by developer of program

2) Variable Loop or Odd Loop :- Where number of repetition or iteration is controlled by end user of program

Step to Manage Repetition in Loop :-

1) Initialize the counter variable

2) Check the Loop Execution Condition based upon Counter Variable and this condition decide Loop Body execute or not. Here Loop Body means set of statement which you want to repeat.

3) Reinitialize the counter variable in such a way it approaches to un-satisfy the Loop Execution Condition after giving desired iteration

Looping Construct :- In Python, there exists only one Looping Construct i.e. while Loop.

Programming Trick

Like math while we solving a theorem we assume something and according to this assumption we solve it. It's another name is flagging concept.

Efficiency of Program

We mainly focus upon two factor

1) How much time will take by our program to solve it

2) How much memory will take by our program to solve it

Memory can be easily increases But Time cannot.

So our focus upon Time Factor of Program. This is called Time Complexity of Program.

We assume a RAM ( Random Access Machine ) model and this machine takes unit time to solve a statement. So we simple count the number of statement execution according to input size and analyze the growth of number of statement.

Jump Statement

break :- This statement is used as an exit statement from a Loop Block.

continue :- This statement is used as an exit statement from current iteration of loop

math module

math module provide us readymade mathematical function

sqrt() :- give us square root

pow() , ceil() , floor() , log2() , log10()

Infinite Loop

Loop which execute infinite times. There are official syntax of creating infinite Loop

while True :  
 action

In such loop break statement is used for exit