

Python Basics (Part 2)

- SHREYAS S

Agenda

- Recap of previous session
- Operators
- Conditionals (if/else, nested if/else) and indentations
- Loops (for/while/nested loops)
- Basic operation (string traversal and string slicing)
- Object oriented Programming concepts

Recap of previous section

- Tools for writing python code
- Setting up python on your machine
- Why python and its benefits.
- Variables
- Data types (string, int, float, Boolean)
- Data structures (List, Tuple, Set, Dictionary)
- Basic operation on data structures

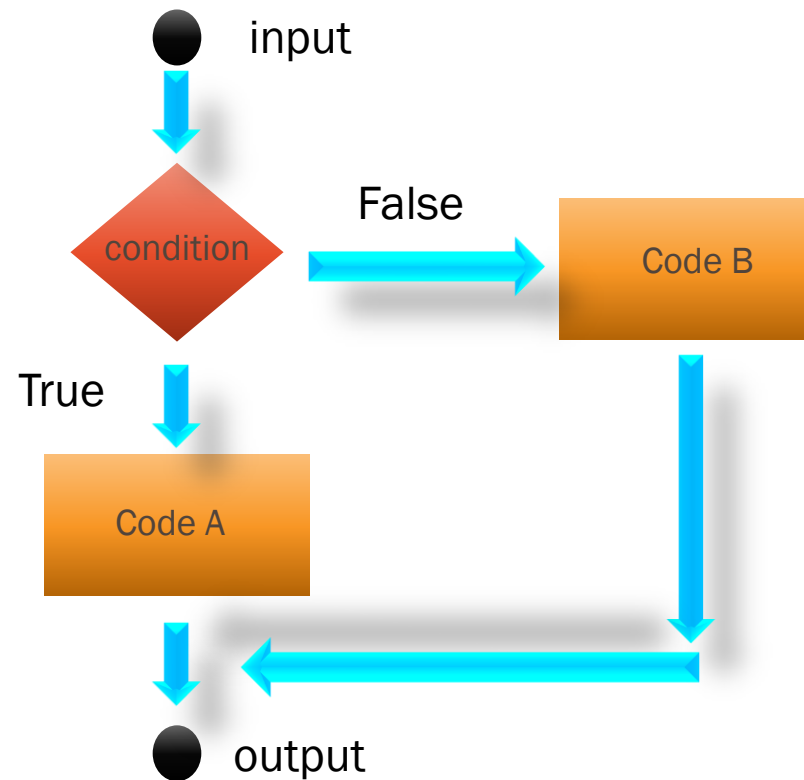
Operators in python

- We use operators to perform any particular operation on variables and values in it.
- We have different type of operators:
 - a. Arithmetic Operators: To perform common mathematical operations.
 - b. Comparison operators: To compare two values.
 - c. Assignment operators: To assign values to variables.
 - d. Logical operators: Used to combine conditional statements.
 - e. Identity operators: Used to compare memory locations of variables.
 - f. Membership operators: To test membership of a sequence (string, list, tuple).
 - g. Bitwise operator : Used to compare binary numbers.

Conditionals and Indentation

- Python program is executed sequentially. The first statement is executed first, then second then third and so on.
- If you want a particular part of code to execute only when particular condition occurs then that's when the concept of conditionals are used.
- In python you can use if/else to direct your execution flow based on some conditions.
- You can also use nested if else. Here if you want to add condition to else part as well then you need to use elif.
- You can also add multiple conditions to one if statement.

Conditionals and Indentation



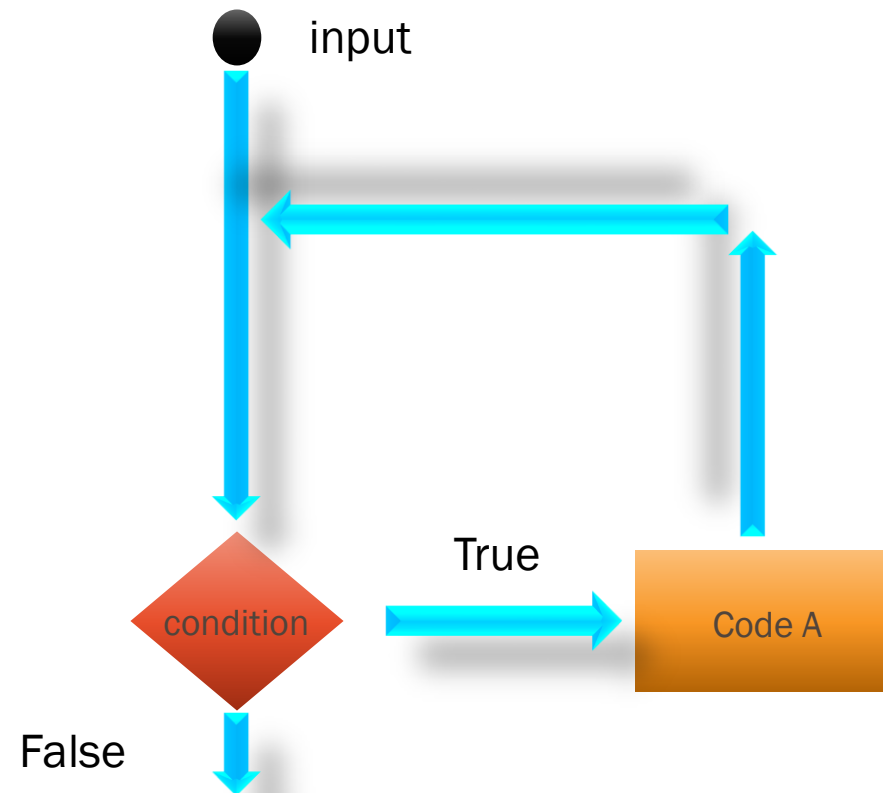
Conditionals and Indentation

- Indentations are blank spaces at the start of each line of code.
- In other programming languages indentation is to improve the code readability. But in python indentations are very important part of the syntax.
- Indentations help us and the machine to understand a block of code.
- Indentations are used in if/else, for, while, functions, classes.
- Python by default uses 4 spaces as it make the code more structured and more readable.
- If indentations are used in a wrong way python may throw indentation error.

Loops

- Like we said before, statements are executed sequentially in python. We have considered the case where we want a code to execute only when a particular condition is satisfied (if/else).
- But consider a case where we want a code to execute n number of times based on a condition.
- This is where the concept of loops come in. A loop allows us to execute a set or block of code n number of times.
- When the condition is not satisfied the loop breaks and the code after the loop is executed.
- In python you can use for loops or else while loops based on your requirement.

Loops



For loops

- For loops are used to iterate over the items in any type of sequence like list, tuple.
- The statement of for loop starts with the keyword *for* and this should all be lowercase.
- When you mention a sequence in for loop, that sequence is evaluated.
- Then each item in the sequence is assigned to the variable you mention after the for keyword one by one.
- Consider first element in the sequence is assigned to the variable, then the code inside for loop is executed. This is one iteration of for loop. This process goes on until all the items are iterated over in the sequence.

While loops

- When you want a block of code to be executed until a condition is not true then you should use while loop here.
- The while loop starts with the while keyword and then the expression based on which you execute the code below it.
- This expression returns Boolean values such as true or false.
- When the expression is evaluated to true then the code block below it is executed and when the expression evaluates to false the loop breaks itself and the pointer to the code comes out of the loop.

Break, continue and else in loops

- While using loops you can use break keyword to break the loop when a particular condition is satisfied. Breaking the loop means the pointer comes out of the loop and the code after the loop block starts executing.
- In similar way you can use continue to skip a block of code that needs to be executed when the condition is satisfied.
- You can use else command in while and for loop to execute a block of code when the expression in while part is evaluated to false. (similar to else in if else).

Basic operations on strings

- Strings can be treated as arrays in python.
- You can perform multiple operations on strings like indexing, traversing and slicing.
- Strings can be visualized as arrays of characters.
- Just like indexing, and slicing in list we can do the same in string using the square brackets.
- You can also traverse the string using for loop.
- Let look into code on how to perform these operations on string.

Object Oriented Programming concepts (Theory)

- OOPs is one of the most important concept in any programming language. Most of the interview questions usually revolve around this concept.
- Before moving on to OOPs concept first lets look into what is procedural programming.
- When you declare variables and functions you divide the whole logic into multiple functions and the data stored in variables.
- The logic in those functions are performed on the data stored in variables. This type of programming is very simple and easy to follow.
- But say for example you have lot of functions and variables in your program and they are interconnected with each other.

Object Oriented Programming concepts (Theory)

- In such cases where function are interdependent and there are lot of function there its bit difficult to manage the code.
- This problem of interdependency and code management is solved by Object oriented programming.
- In OOPs we group related variables and functions. These groups are called objects where variables are properties of objects and functions are methods of object.
- The way you construct the object is through class which is also called as blueprint for creating objects.
- Next we will look into the key concepts of object oriented programming.

Object Oriented Programming concepts (Theory)

➤ Following are the four important concepts of OOPs:

a) Encapsulation

b) Abstraction

c) Inheritance

d) Polymorphism

➤ Encapsulation means to wrap the data and methods related to an object into a single unit.

➤ This restricts access to methods and variables of a particular object by any other object. This helps us to ensure that the data is not being modified by accident by other objects.

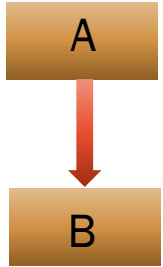
➤ The variables can only be modified by objects method. These type of variables are called private variables.

Object Oriented Programming concepts (Theory)

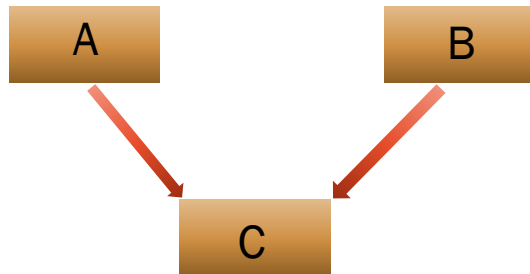
- Abstraction means to hide the details of internal implementation and show only few functionalities. This makes the user experience simpler by hiding the complex logic and details.
- Inheritance is the concept where in the child objects of a particular object acquires all the properties and methods of the parent object as well as they can have their own properties and methods.
- Inheritance provides reusability of code.
- There are 4 types of inheritance:

Object Oriented Programming concepts (Theory)

- Single inheritance

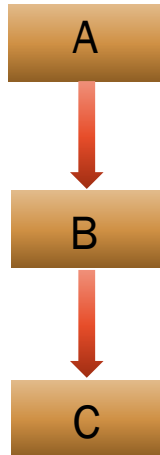


- Multiple inheritance

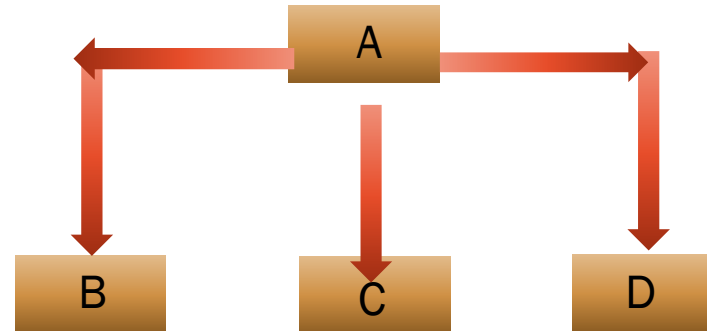


Object Oriented Programming concepts (Theory)

- Multilevel inheritance



- Hierarchical Inheritance



Object Oriented Programming concepts (Theory)

- Polymorphism : Poly = Many and Morph = Form
- Polymorphism concept helps you to have same function name for different objects but have different logic with respect to each other.
- In case of Animal object you can have different objects of animal like dog, cheetah, fish, bird. All these objects can have functions named move but the way each animal moves would be different. Like dog walks, cheetah runs, fish swims and birds fly.

Agenda for Next Session

- ❑ Classes and objects
- ❑ Type conversion
- ❑ Functions and lambda
- ❑ Function argument
- ❑ Scope

QUESTIONS?

THANK YOU