Submitted by:. Sana Iqbal

Roll No:. SP20M2BB037

(Assignment)

Q) Introduction of python?

Language Introduction

Python is a dynamic, interpreted (bytecode-compiled) language. There are no type declarations of variables, parameters, functions, or methods in source code. This makes the code short and flexible, and you lose the compile-time type checking of the source code.

Python is a dynamic, interpreted (bytecode-compiled) language. There are no type declarations of variables, parameters, functions, or methods in source code. This makes the code short and flexible, and you lose the compile-time type checking of the source code. Python tracks the types of all values at runtime and flags code that does not make sense as it runs.

An excellent way to see how Python code works is to run the Python interpreter and type code right into it. If you ever have a question like, "What happens if I add an int to a list?" Just typing it into the Python interpreter is a fast and likely the best way to see what happens. (

Q) Modules:

In Python, Modules are simply **files with the ". py" extension containing Python code that can be imported inside another Python Program**. In simple terms, we can consider a module to be the same as a code library or a file that contains a set of functions that you want to include in your application. To create a module, we have to save the code that we wish in a file with the file extension **".py"**. Then, the name of the Python file becomes the name of the module.

For Example,

In this program, a function is created with the name **"welcome"** and save this file with the name **mymodule.py** i.e. name of the file, and with the extension **".py"**.

We saved the following code in a file named **mymodule.py**

```
def. welcome(name):
  print("Hello, " + name +"
```

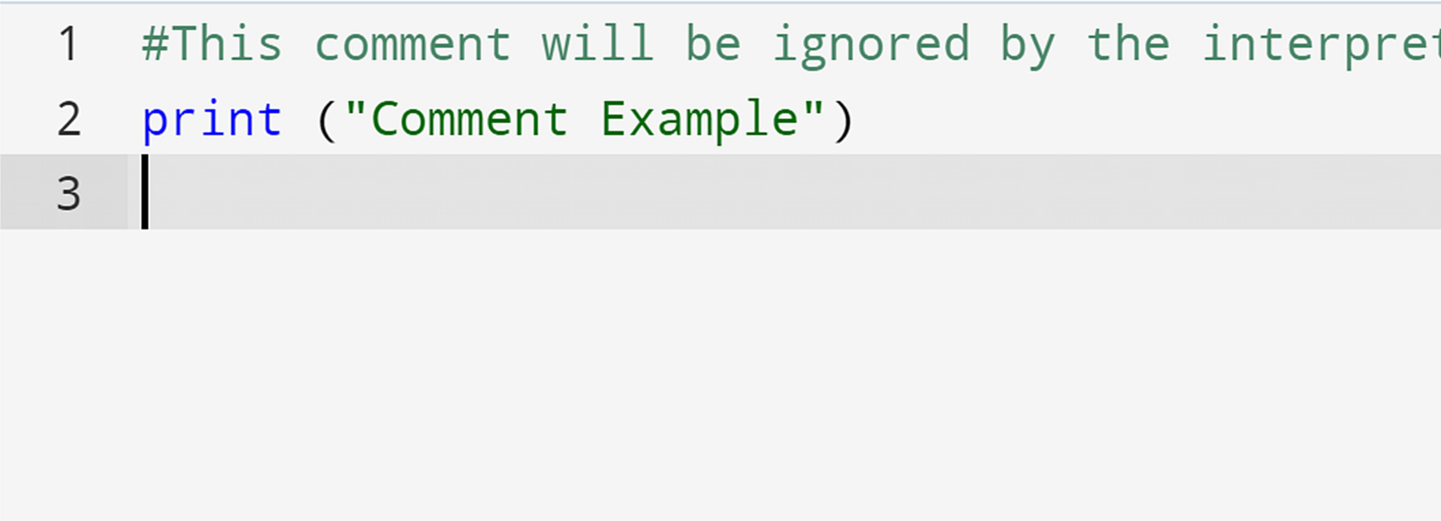
Example:

```
marks1=int(input("Enter your marks :"))
marks2=int(input("Enter your marks :"))
marks3=int(input("Enter your marks :"))
marks4=int(input("Enter your marks :"))
if(marks1 <33 or marks2<33 or marks3<33):
    print("you have less than 33 marks")
elif (((marks1+marks2+marks3)/3)<40):
    print("Your average is less than 40")
else:
    print("Pass")
```

Q) Comments:

Comments in Python are **identified with a hash symbol, #, and extend to the end of the line**. Hash characters in a string are not considered comments, however. There are three ways to write a comment - as a separate line, beside the corresponding statement of code, or as a multi-line comment block.

Comments in python is the inclusion of short descriptions along with the code to increase its readability. A developer uses them to write his or her thought process while writing the code. It explains the basic logic behind why a particular line of code was written. They are just meant for the coders themselves or other developers to understand a piece of code, especially since the Python interpreter completely ignores comments in Python. You can see this in the following example.



```
#This comment will be ignored by the interpret
print ("Comment Example")

```

What Are Comments in Python Used For?

Comments in Python are identified with a hash symbol, #, and extend to the end of the line. Hash characters in a string are not considered comments, however. There are three ways to write a comment - as a separate line, beside the corresponding statement of code, or as a multi-line comment block.

There are multiple uses of writing comments in Python. Some significant uses include:

- Increasing readability
- Explaining the code to others
- Understanding the code easily after a long-term
- Including resources
- Re-using the existing code

IF-Else Statement:

In computer programming, we use the `if` statement to run a block code only when a certain condition is met.

For example, assigning grades **(A, B, C)** based on marks obtained by a student.

1. if the percentage is above **90**, assign grade **A**
2. if the percentage is above **75**, assign grade **B**
3. if the percentage is above **65**, assign grade **C**

In Python, there are three forms of the `if...else` statement.

1. `if` statement
2. `if...else` statement
3. `if...elif...else` statement

Example:

```
def hello():
    marks=int (input("Enter your marks :"))
    if(marks>=80):
        print("Grade is A")
    elif(marks>=70):
        print("Grade is B")
    elif(marks>=50):
        print("Grade is C")
hello()
```

Getting Values from User:

```
var1=int (input("Enter the value :"))
var2=int (input("Enter the value :"))
var3=int (input("Enter the value :"))
var4=int (input("Enter the value :"))
print(max(var1,var2,var3,var4))
```

What Are the Advantages of Using Comments in Python?

Comments in Python provide numerous advantages. Their primary benefits include:

- Makes the code easily understandable by other Programmers.

- The code becomes self-explanatory

- Helps remember why we used a specific command, method, or function in the code

- Enables the interpreter to ignore some part of the code while testing

Q) Pip

PIP is **a package manager for Python packages, or modules if you like**. A package contains all the files you need for a module.

Modules are Python code libraries you can include in your project. Pip is **a recursive acronym that can stand for either "Pip Installs Packages" or "Pip Installs Python"**. Alternatively, pip stands for "preferred installer program". Python 2.7. 9 and later (on the python2 series), and Python 3.4 and later include pip (pip3 for Python 3) by default.

Data variables

- Variables are nothing but reserved memory locations to store certain values. This means that when you create a variable you reserve some space in memory.
- Based on the data type of a variable, the interpreter allocates memory and decides what can be stored in the reserved memory. Therefore, by assigning different data types to variables, you can store integers, decimals or characters in these variables.
- While learning about Python variable types it is highly recommended that you try out all statements in the Python shell and see the results. You must also explore and experiment with the shell.

Q)Python Variable

Variables are the most important aspect of any programming language including python. In simple words, variables are names given to memory locations. The values stored in these memory locations can be altered throughout the programming life cycle. Variables are stored in the main memory and are generally required to carry forward various processes and transactions. Since we are talking about memory allocated to variables, it is quite obvious that once the variables are defined they will get associated with a physical address in the main memory. If we start accessing these values by physical addresses, the code will get complicated and that is why the addresses are associated with variable names.

- A variable name must begin with a letter of the alphabet or an underscore(_) Example: abc=100 #valid syntax. ...

- The first character can be followed by letters, numbers or underscores. ...

- Python variable names are case sensitive. ...

- Reserved words cannot be used as variable names.

Python Variable Types

- Numbers.
- String.
- List.
- Tuple.

Q) String

In computer programming, a string is a sequence of characters. For example, **"hello" is a string containing a sequence of characters 'h' , 'e' , 'l' , 'l' , and 'o'** . Here, we have created a string variable named string1 . The variable is initialized with the string

Python Programming .

Q)List

Lists are used to store multiple items in a single variable. Lists are one of 4 built-in data types in Python used to store collections of data, the other 3 are Tuple, Set, and Dictionary, all with different qualities and usage.A list is an ordered data structure with elements separated by a comma and enclosed within square brackets. For example, list1 and list2 shown below contains a single type of data. Here, list1 has integers while list2 has strings. In Python, list and tuple are **a class of data structures that can store one or more objects or values**. A list is used to store multiple items in one variable and can be created using square brackets. Similarly, tuples also can store multiple items in a single variable and can be declared using parentheses.

Example

Create a List:

thislist = ["apple", "banana", "cherry"]

print(thislist)

```
numbers = [6, 9, 3, 1]
numbers.sort()
print(numbers)
```

Q) tuple

Tuples are used to store multiple items in a single variable. Tuple is one of 4 built-in data types in Python used to store collections of data, the other 3 are List, Set, and

Dictionary, all with different qualities and usage. A tuple is a collection which is ordered and unchangeable.

Python Tuples

```
Mytuple = (“apple”, “banana”, “cherry”)
```

Tuple

Tuples are used to store multiple items in a single variable.

Tuple is one of 4 built-in data types in Python used to store collections of data, the other 3 are List, Set, and Dictionary, all with different qualities and usage.

A tuple Is a collection which is ordered and unchangeable.

Tuples are written with round brackets.

ExampleGet your own Python Server

Create a Tuple:

```
Thistuple = (“apple”, “banana”, “cherry”)
Print(thistuple)
```

Tuple Items

Tuple items are ordered, unchangeable, and allow duplicate values.

Tuple items are indexed, the first item has index [0], the second item has index [1] etc.

Ordered

When we say that tuples are ordered, it means that the items have a defined order, and that order will not change.

Unchangeable

Tuples are unchangeable, meaning that we cannot change, add or remove items after the tuple has been created.ExampleGet your own Python Server

Create a Tuple:

This tuple = (“apple”, “banana”, “cherry”)

Print(thistuple)

Q) Dictionary

Dictionary

Dictionaries are used to store data values in key:value pairs.

A dictionary is a collection which is ordered*, changeable and do not allow duplicates.

Dictionaries are written with curly brackets, and have keys and values:

Example

Create and print a dictionary:

```
theodicy = {
  "brand": "Ford",
  "model": "Mustang",
  "year": 1964
}
print(thisdict)
```

Dictionary Items

Dictionary items are ordered, changeable, and does not allow duplicates.

Dictionary items are presented in key:value pairs, and can be referred to by using the key name.

Example

Print the "brand" value of the dictionary:

```
thisdict = {
  "brand": "Ford",
  "model": "Mustang",
  "year": 1964
}
print(thisdict["brand"])
```

Q) Python Strings

Python Strings with the help of examples

In computer programming, a string is a sequence of characters. For example, "hello" is a string containing a sequence of characters 'h', 'e', 'l', 'l', and 'o'.

We use single quotes or double quotes to represent a string in Python. For example,

```
# create a string using double quotes

string1 = "Python programming"


# create a string using single quotes

string1 = 'Python programming'
```

```
# create string type variables

name = "Python"
print(name)

message = "I love Python."
print(message)
```

Run Code

```
Python
```

```
I love Python
```