Assignment - 2

What are the data types in python? Explain.

python has six standard data types

1. Numbers
2. String
3. List
4. Tuple
5. Set
6. Dictionary

Numbers

In numbers, these are mainly 3 types which include integer, float and complex These are defined as a class in python, inorder to find to which class the variable belongs to you can use type() function

Eg:
```
a=5
print (a)
print (type(a))
```

output
```
5
<class 'int'>
```

String

A string is an ordered sequence of characters

we can use single (or) double quotes

Eg: 'welcome'
 "welcome"

```
st1 = "hi"
st2 = "sir"
print(st1+st2)
```

output : hi sir

List

A list can contain a series of values

List variables are declared by using brackets [ ]. It is mutable - we can modify

Eg:
```
List = [2,4,5.5,"Hi"]
print("List[2] = ", List[2])
```

output : List[2] = 5.5

```
print("List[0:3] = ", List[0:3])
```

output : List[0:3] = [2,4,5.5]

Tuple

It is a sequence of python objects. separated by commas
It is immutable. It is defined using parenthe

Eg:

```
Tuple = (50, 15, 25.6, "python")
print (" Tuple [1] = ", Tuple [1])
```

output Tuple[1] = 15

Set

A set is an unordered collection of items
It is defined by values separated by comma inside braces { }

Eg:

```
set = {5, 1, 2.6, "python")
print (set)
```

output : { 'python', 1, 5, 2.6 }.

Dictionary

Dictionaries are most flexible built-in data types in python

Items are stored and fetched by using the key. They are used to store huge amount of data

To retrieve the value we must know the key. dictionaries are defined within braces {}

Eg: Dict = { 1:'Hi', 2:7.5, 3:'class'}

print (Dict)

output : {1:'Hi', 2:7.5, 3:'class'}

2 Briefly explain history of python

Python was conceived in the late 1980s by Guido Van Rossum at Centrum Wiskunde & Informatica (CWI) in the Netherlands as a successor to the ABC language, capable of exception handling and interfacing with Amoeba operating system. Its implementation began in December 1989. Van Rossum shouldered sole responsibility for the project, as the lead developer, until 12 july 2018.

when he announced his permanent vacation from his responsibilities as python's Benevelent Dictator for Life, a title the python community bestowed upon him to reflect his long term commitment as the project's chief decision-maker.

He now shares his leadership as a member of a five-person steering council. In jan 2019, active python core developers elected Brett Cannon, Nick Coghlan, Barry Warsaw, Carol Willing and Van Rossum to a five member Steering Counil to lead the project.

Python 2.0 was released on 16 oct 2000 with major new features, including a cycle-detecting garbage-collector and support for unicode

Python 3.0 was released on 3 Dec 2008. It is not completely backwar compatible. Many of its features were backported to python 2.6x and 2.7.x versio

Releases of python 3 include the 2 to 3 utility. the translation of python 2 code to 3

python 2.7's end of life date was initially set at 2015 then postponed to 2020 out of concern that a large body of existing code could not easily be forward-ported to python 3.

3 Explain all the Operators in Python

python operators falls into 7 categories

1. Python Arithmetic Operator:

These operators include python operators for basic mathematical operations.

a. Addition (+) adds the value

eg 3 + 4

out 7

b Subtraction (-) subtracts the value

Eg: 3-4

-1

c. Multiplication (*)

multiplies the value.

Eg: 3 * 4

output: 12

d. Division (/)

divides the value

Eg: 3/4

0.75

e. Exponential (**)

Raises the first number to the power of second number

Eg: 3 ** 4

output: 81

f. Floor Division (//)

It dumps the digits after the decimal

Eg: 3//4

1

g. modulus (%)

returns the value of remainder

Eg: 3%4

3

Scanned with CamScanner

2. Relational Operators :

It carries out the comparison of operands

a) Less than (<)

Eg: 3<5

True

b) Greater than (>)

Eg: 3>4

False

c) Less than or equal to (<=)

Eg: 7<=7

True

d) Greater than or equal to (>=)

Eg: 0>=0

True

e) Equal to (==)

Eg: 3==3.0

True

f) Not equal to (!=)

Eg: 1!=1.0

False

3. Assignment Operators

Assigns a value to a variable

a. Assign (=)

Eg: a=7

b.) Add and Assign (+=)

Eg: a+=2

c) -=

x-=5

d) *=

x*=5

e) /=

x/=5

f) %=

x%=5

g) **=

x**=5

h) &=

x&=5

i) |=

x|=5

j) >>=

x>>=5

k) <<=

x<<=5

4. Logical Operators

These are conjunctions that you can use to combine more than one ~~combine~~ condition.

a) and

~~xy~~ x and y

True if both the operands are true

b) or

True if either of the operands is true

x or y

c) not

True if operand is false

not x.

5) Identity Operator

These operators test if the two operands share an identify. we have two identity operators - 'is' and 'is not'.

a) is

2 is 20	'2' is "2"
false	True

b) is not

2 is not '2'

True

6) Bitwise Operator

a) Binary AND (&)

Eg: 2&3

output 2

b) Binary OR(|)

Eg: 2|3

output: 3

c) Binary XOR(^)

Eg: 2^3

output 1

d) Binary One's Complement(~)

Eg: ~-3

output: 2

e) Binary Left-shift (<<)

Eg: 2<<2

8

f) Binary Right shift (>>)

eg: 3>>2

output: 1

7) Membership python Operator

These operators test whether a value is a member of a sequence. The sequence may be a list, a string, a tuple we have two operators 'in' and 'not in'

a) in

True if value/variable is found in sequence

Eg: 5 in x

b) not in

True if vaule/variable is not found in the sequence

Eg: 5 not in x

4-Explain featuyes of python

Python provides lots of featuyes :

1) Easy to Learn and Use

Python is easy to learn and use. It is developer friendly and high level language

2) Expyessive language

It is moye expyessive means that it is moye undesstandable and yeadable

3) Interpyeted language

Python is an interpyeted language. it executes the code line by line at a time. debugging is easies

4) Cross-platform Language

Python can sun equally on different platfoyms such as Windows, Linux, Unix etc.,

5) It is a portable Language

6) Free and Open Source
It is freely available at offical web address. The source code is also available.

7) Object Orient Language
It supports object oriented language and concepts of classes and objects come into existence

8) Extensible
It implies that other languages such as c/c++ can be used to compile the code and thus it can be used further in our python code

9) Large Standard Library
It has large and broad library and provides rich set of modules and functions for rapid application development

10) GUI Programming support
Graphical user Interfaces can be developed using python

5. Justify why python is interactive interpreted language

Python is interactive. When a python statment is entered, and is followed by the Return key, if appropriat, the result will be printed on the screen., immediately, in the next line. This is particularly advantageous in debugging process. In Interactive mode of operation, python is used in a similar way as a Unix command line or the terminal

Interactive python is very much helpful for debugging. It simply returns prompt or the corresponding output of statment if appropriat and returns error for incorrect statments.

Python is an interpreted objectoriented language. By interpreted it is meant that each time a program is run the interpreter checks through code for errors and then interprets the instructions into machine-readable bytecode

An interpreter is a translator in computer's language which translates the given code line by line in machine readable bytecodes. If any error is encountered it stops the translation until the error is fixed.

Thus the python is an interactive and interpreted language