

Assignment: 2

Investigate the following features or constructs as they pertain to Python:

1. Interpretation :-

Python:

Python program runs directly from the source code rather than machine code.

The .pyc file first compiled from python .py source file. Python is designed to be used interpretively.

Java:

Programs written in Java must be explicitly compiled into byte codes (.class files), though an IDE do this automatic.

2. Boolean expressions :-

```
>>>print(19>1)
```

```
True
```

```
>>>print(8==8)
```

```
False
```

A Boolean expression evaluates to either true or false. Python provides the Boolean type that can be either set to False or True.

Conditional operators (<, <=, >, >=, ==, !=) compare the values of 2 objects and returns True or False. In Python, all operators have same priority but C doesn't have same priorities for each operator.

3. short circuit evaluation :-

```
>>>a=6
```

```
>>>b=2
```

```
>>>if (a==5 and b==2):
```

```
...     print("inside")
```

```
... else:
```

```
...     print("outside")
```

```
>>>outside
```

We can use and, or and not for compound Boolean expressions.

In above example, for 'AND' if first expression is false, then it won't check other and it will return false. For 'OR', if the first expression is true then it won't check second expression and it will return True.

Java also has Short Circuit Evaluation for '&&' and '||' operator but not for '&' and '|'.

In 'or' if the second condition has method call and first condition is true then it will not check for second and method call will skip execution.

4. numeric types :-

```
>>>a=15.5
```

```
>>> type(a)
```

```
<class 'float'>
```

```
>>>b=32
```

```
>>> type(b)
```

```
<class 'int'>
```

```
>>>c=9 + 2j
```

```
>>> type(c)
```

```
<class 'complex'>
```

In python, we don't need to declare type before assigning to a variable.

Python has three distinct numeric types which are Integer, Float and Complex.

In Java, we need to define type while declaring a new variable.

5. strings :-

```
>>> s='This is example of a String. '
```

```
>>> print(s)
```

```
This is example of a String.
```

We can declare string almost same way in both java and python.

Python strings can be enclosed in either single or double quotes (' or '').

6. arrays :-

```
>>>sub = ["PLC", "OSD"]
>>>print(sub)
["PLC", "OSD"]
>>>sub.append(2021)
>>>print(sub)
["PLC", "OSD", 2021]
```

Python doesn't have built-in array type but it can be implemented by using list.

In java array is a group of like-typed variables that are referred to by a common name. we can not add values with different types.

7. lists :-

```
sub = ["PLC", "OSD", "AdvAlgo"]
>>> print(sub)
(["PLC", "OSD", "AdvAlgo"])
>>>print(type(sub), len(sub))
(<type 'list'>, 3)
```

A list is a collection which is ordered and mutable. In Python lists are written with square brackets.

Lists can never be used as dictionary keys, because lists are mutable.

Java has 4 types of list ArrayList, LinkedList, Vector and Stack in Collection interface.

8. tuples :-

```
>>>sub=("PLC", "OSD")
>>>print(sub, sub[0], len(sub))
>>> ((‘PLC’, ‘OSD’), ‘PLC’, 2)
```

A tuple is a collection which is ordered and unchangeable.

In Python tuples are written with round brackets.

Java doesn't have any such inbuilt data structure to support tuples, but it can be implemented by using javatuple library.

9. Slices :-

```
>>> cars = ("BMW", "Rolls-Royce", "Audi", "Tesla", "Volvo", "Jaguar", "Ferrari",
"Range-Rover")
>>> c = slice(2)
>>> a = slice(1,3)
>>> print(cars[c] , cars[a])
>> (('BMW','Rolls-Royce'),('Rolls-Royce','Audi'))
```

Slice object is used to slice the given sequence such as list, array, string or range.

Syntax of slice is as follows :

slice(stop)

slice(start,stop,step)

Java doesn't have this kind of functionality directly. Instead we can achieve slicing of arrays or lists with the help of ArrayUtils library or Java lambda expressions.

10. index range checking :-

```
>>>a = ["Hey", "Hello"]
>>>print(a [-1], a[-2], a[0])
hello hey hey
>>> print(a.index('Hey'))
0
```

Python also supports negative indexing whereas Java doesn't support. Also, we can check index of array object by index object on python.

11. dictionaries :-

```
>>> a={1:"a",2:"b",3:"c"}
>>>print(a.values(), a, type(1), a[1],len(a))
>>> ['a', 'b', 'c'], {1: 'a', 2: 'b', 3: 'c'}, <type 'int'>, 'a', 3)
```

A dictionary is a collection of key-value pairs which is unordered and changeable.

C language doesn't have this feature but Java does have Key-value pair data structure called Map.

12. if statement :-

```
>>> x = 1
>>> y = 2
>>> if x > y :
```

```

...     print('If statement tested')
...elif x == y:
...     print("elif statement tested ")
...else :
...     print("else statement tested ")
>>>If statement tested

```

Most of the language has same type of if statement but implementation in python is easy.

If the condition is true, then it will execute indented block otherwise skip the indented block.

13. switch statement :-

Python doesn't have a switch statement. But C, C++ and Java have this and it executes the matched case.

In order to implement switch case in Python, we can use several else-if statements.

14. for loop :-

```

>>>for sub in "PLC":
...     print (sub)
P L C

```

For statement iterates all elements of objects. Implementation of for loop is easy in python.

This is less like the 'for' keyword in other programming languages, and works more like an iterator method as found in other object-orientated programming languages.

15. while loop :-

```

>>> x = 5
>>> while x > 0 :
... print(x)
... x=x-1
...
5 4 3 2 1

```

While loop continues to iterate until condition statement become false.

while loop can have else statement in python but not in java.

It can be trapped into infinite loop if condition never get false value.

16. indentation to denote code blocks :-

```
>>>if 3>1:
... (Whitespace or Tab)print ("Indentation")
>>>(Indentation)
```

To indicate a block of code in Python, you must indent each line of the block by the same amount. But in Java, Program code in a Java program is contained in a block. A block begins with an open curly brace symbol and ends with a close curly brace symbol.

17. type binding :-

```
>>>x=10
>>>print(type(x))
<class 'int'>
```

```
>>>y='Hello'
>>>print(type(y))
<class 'str'>
```

Python use dynamic type binding, type of a variable is determined at the execution time. while java use static type binding.

18. type checking :-

```
>>> x=10
>>>print(type(x))
>>> <type 'int'>
>>> x="Hello"
>>>print(type(x))
>>> <type 'str'>
```

This is declaration of variable two times. But python accept it and do not generate error. But we cannot do this same thing in java.

In Python no type checking is done. Instead the type is assigned dynamically. Also, the user can reassign a once declared variable to a different data type.

19. functions :-

```
>>> def me(name):
...   return "Hello,"+ name
...
>>> print(me("Sidhdharth"))
>>>Hello, Sidhdharth
```

Function is a block of reusable code defined using the def keyword, only runs when called.

Python function support default parameter while java not.

There is no declared return type, sometime create difficulty to understand without comments.

20. one other feature - your choice :-

```
>>>a = [1,2,3,4,5]
>>>print (list(map(lambda x: x**2, a)))
>>> [1,4, 9, 16, 25]
```

Lambda is anonymous functions with any number of arguments and only one expression. The expression is evaluated and returned answer.

Well-named function with def keyword are more readable and understandable than a lambda function Java doesn't have lambda until java 8. However anonymous classes use to create lambda in java.

.