Python Assignment

August 9, 2024

Basics Python Assignment 1. Explain the key features of Python that make it a popular choice of programming. Answer – Many features of Python make it a popular choice of programming some of them are: a. It has a wide library, making it more user-friendly and easier to use and many pre-defined words are already stored there. b. It is used for every type of operation or area, such as frontend, backend, data analysis, and many more. c. Python is easy to read and learn, it becomes easy for the non-technical person to understand it and can learn it in a short period. d. Python have a huge active community that makes vast libraries and framework for the programmer to work on it. e. Python has image processing which helps the programmers enhance the analysis and code.2. Describe the role of predefined keywords in Python and provide examples of how they are used in a program. Answer- Predefined words are words that have some special meaning or reserved meaning by the python and each predefined word is unique have its purpose. Predefined words cannot be used as an identifier. The role of a predefined function is to control the flow of the program and apply conditions, loop and control statements to the program.

Some of the predefined keywords are: if, if- else, if- elif -else, nested- if-else, input, for, while, break, continue, print, True, False.

[ ]:
```
#Ex- input means to make the other person enter their details
name = input ("Enter your name")
email = input ("enter your email")
contact = int(input ("Enter your contact"))
```

```
Enter your name Mansi
enter your email gargmansi@gmail.com
Enter your contact 777777777777
```

[77]:
```
#Ex- True/print means to check the boolean value if True or False
value = 100
if (value ==100):
    print ("True")
else:
    print ("False")
```

```
True
```

[82]:
```
#in not means the value/object is not present in the program
item = ["perryperi", "Burger", "Pizza", "cold drink"]
if ("Burger" and "Pizza" in item):
    print ("Bring it to mee")
```

```
else:
    print ("Just bring coke for me")
```

```
Bring it to mee
```

3. Compare and contrast mutable and immutable objects in Python with examples. Answer – Mutable objects or containers are the ones that can be changed after they have been created such as lists, dictionaries, and sets. These have better performance as they can be modified without creating a new one. Immutable objects/ containers are ones whose state or values cannot be changed after they are created like integers, floats, and strings. They are safer as one cannot change they have to create a new program.

[16]:
```
#Ex - mutable object:
list_cont = [1,2, "mansi",4.8, "learner"]
list_cont [2] = "pwskills"
list_cont [2]
```

[16]:
```
'pwskills'
```

[19]:
```
#Ex - immutable object:
b = "pwskills"
b[0] = "i"
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[19], line 3
      1 #Ex - immutable object:
      2 b = "pwskills"
----> 3 b[0] = "i"

TypeError: 'str' object does not support item assignment
```

4. Discuss different types of operators in Python and provide examples of how they are used. Answer- There are mainly seven types of operators in Python that are: a. Athematic operators - are used to do mathematical operations like addition, subtraction, multiplication, division, modulus, and exponentiation.

b. Comparison operators – are used to compare two values like equal to, not equal to, greater than, less than, greater than equal to, and smaller than equal to.

c. Logical operators- are used to combine operators like logical AND, logical OR, and Logical NOT.

d. Bitwise operators – are used to do bit-level codes like bitwise AND, bitwise OR, bitwise NOT, bitwise XOR, Left shift, and Right shift.

e. Assignment operators- are used for assigning the values to variables/ objects like addition assignment, subtraction assignment, multiplication assignment, and division assignment.

f. Membership operators- are used to test the values in the variables like in, not in.

g. Identity operators- are used to compare the memory location of two objects like is, is not.

[22]:
```
#Ex Athematic operators
a = 15
b = 30
a*b
```

[22]: 450

[26]:
```
#Ex Comparision operators
10 != 10
```

[26]: False

[27]:
```
#Logical operators
True and False
```

[27]: False

[28]:
```
#Logical operators
True or False
```

[28]: True

[29]:
```
#Bitwise operators
18 & 3
```

[29]: 2

[30]:
```
#Bitwise operators
5^3
```

[30]: 6

[36]:
```
#Assignment operators
i = 20
i = i/4
i
```

[36]: 5.0

[41]:
```
#Membership operators
j = ["data", "analysis", "program"]
"program" in j
```

[41]: True

[42]:
```
#Membership operators
j = ["data", "analysis", "program"]
"science" in j
```

[42]: False

[43]:
```
#Identity operators
a = 100000
b = 100000
a is b
```

[43]: False

[44]:
```
#Identity operators
a = 10
b = a
a is b
```

[44]: True

5. Explain the concept of type casting in Python with example Answer- The process of changing the data type of a variable/value/object. Type casting is done for computing the variable by using different operators in python, as the data is mismatched many times. Some types of type casting are string, float, integer, list.

[47]:
```
#EX of type casting in Python
a = "200"
b = 200
type (a)
```

[47]: str

[48]:
```
type (b)
```

[48]: int

[50]:
```
#to add a and b use type casting so that they are be added -EX of type casting␣
 ↪in Python
a = int(200)
type (a)
```

[50]: int

[51]:
```
a+b
```

[51]: 400

6. How do conditional statments work in Python? Illustrate with examples Answer- Conditional statements are the statements or programs that executed when the condition is true or false. There are four types of conditional statements that are if, if- else, if- elif -else, nested- if-else.

[52]:
```
#Ex- If - else means if one condition is not fulfilled will try to another␣
 ↪condition.
a=45
if (a%2 == 0):
    print ("The number is even")

else:
    print ("The number is odd")
```

```
The number is odd
```

[55]:
```
#Ex- elif means more that two conditions.
marks = 80
if (marks >= 90):
    print ("Excellent")
elif (80 <= marks <90):
    print ("Very good")
elif (65 <= marks <80):
    print ("Good")
elif (50 <= marks <65):
    print ("Average")
```

```
Very good
```

[71]:
```
#Ex- nested if else means there can be multiple if else conditions inside if␣
 ↪else.
voucher = True
Income = 150000
if voucher:
    if (Income >= 50000):
        if (Income <= 100000):
            print ("Your income level is good")
        else:
            print ("Your income is very high")
    else:
        print ("Your income is very low")
else:
    print ("you are not eligible")
```

```
Your income is very high
```

7. Describe the different types of loops in Python and their use cases with examples. Answer- Loop statements are the statements that allows you to excute a block of code repetitively. In this there are two types of loop that are for loop and while loop.

[75]:
```
#Ex- while / continue means repeatedly executed a block of code until a␣
 ↪condition is met.
n = 10
i= 2
while (i < n):
    i=i+1
    if (i == 5):
        continue
    print (i)
else:
    print ("There is no break, your program has run successfully")
```

```
3
4
6
7
8
9
10
There is no break, your program has run successfully
```

[76]:
```
#Ex- for/ break means to iterate over a sequence of elements it is mainly for␣
 ↪string and list.
m = [1,5,"myrules", 4.8]
for m in m:
    if (m == "myrules"):
        break
    print (m)
else:
    print ("there is no break")
```

```
1
5
```