

# python-basics-assignment

October 10, 2024

1) Explain key features of python that make it a popular choice for coding? ANS - Python is simple and used to learn which has a huge community, it contains a lot of useful libraries and packages which are used for data science and AI. It is a growing community. It has many important libraries useful for data manipulation, visualization, and machine Learning . Some of these include numpy, pandas , matplotlib, tensorflow. 2) Describe the role of predefined keywords in python and provide examples of how they are used in a program ANS - predefined keywords are reserved words in python which play an important role in programming . every keyword has a particular purpose and functionality in python. Examples include input() to take input from the user, print() to print output on terminal, While to create a loop, In (membership operator), Is (identity operator) 3) compare and contrast mutable and immutable objects in python with examples ANS - mutable objects are objects which can be changed or modified according to the scenario examples are list and dictionary immutable objects are objects which cannot be changed or modified according to the scenario examples are string and tuple

```
[45]: #list
li=[1,2,3,4,5]
print(li)
li[2]=34
li[4]=32
print(li)
```

```
[1, 2, 3, 4, 5]
[1, 2, 34, 4, 32]
```

```
[47]: # dictionary
dic={'one':1,'two':2,'three':3}
print(dic)
dic["one"]=23
dic["two"]=34
print(dic)
```

```
{'one': 1, 'two': 2, 'three': 3}
{'one': 23, 'two': 34, 'three': 3}
```

```
[49]: #string
a="Saabir"
print(a)
a[0]="e"
```

```

-----
TypeError                                Traceback (most recent call last)
Cell In[49], line 3
      1 #string
      2 a="Saabir"
----> 3 a[0]="e"

TypeError: 'str' object does not support item assignment

```

```

[51]: #tuple
      t=(1,2,3,4)
      print(t)
      t[0]=23

```

(1, 2, 3, 4)

```

-----
TypeError                                Traceback (most recent call last)
Cell In[51], line 4
      2 t=(1,2,3,4)
      3 print(t)
----> 4 t[0]=23

TypeError: 'tuple' object does not support item assignment

```

##4) Discuss different types of operators in python and provide examples of how they are used  
 there are 7 different types of operators: they are assignment operators comparison operators arithmetic operators membership operators identity operators logical operators bitwise operators

```

[53]: #assignment operators
      #this is used to assign a value to a variable
      name="saabir"
      email="shaik.saabir2003@gmail.com"
      print(name, email)

```

saabir shaik.saabir2003@gmail.com

```

[61]: #comparison operators
      # it is used to compare various various objects and return a boolean value(True or False)
      a1=1
      a2=2
      print(a1==a2)
      print(a1<=a2)

```

False

True

```
[71]: #arithmetic operators
      #it is used to perform mathematical operations on variables
      de=1
      df=2
      de+=1
      print(de)
      de*=df
      print(de)
      de*=5
      print(de)
```

2  
4  
20

```
[77]: #membership operators
      # it is used to check wheather a object is part of list or tuple, etc
      # it is used to iterate over a list or string they are in and not in
      lis=[1,2,3,4]
      for i in lis:
          print(i,end=" ")
```

1 2 3 4

```
[81]: #identity operators
      # it is used to check or compare the location or references of the variable on
      ↪ object
      # is or not is keywords are used
      s1=1
      s2=2
      print(s1 is s2)
      s2=1
      print(s1 is s2)
```

False

True

```
[94]: #logical operators
      # these are used to perform "and" or "or" operations on various statements on
      ↪ conditions
      # they are & and |
      if ((1 == 1) and (2==2)):
          print("yes")
      if ((1==1) or (2==1)):
          print("or operator is used")
```

yes  
| operator is used

```
[102]: #bitwise operators  
#these are operations that are performed at bitwise level or bit  
# they are bitwise and &, bitwise or |, XOR ^, left shift <<, right shift >>  
b1=100  
b2=34  
print("bitwise and",b1&b2)  
print("bitwise or",b1|b2)  
print("left shift", b1 << 3)
```

bitwise and 32  
bitwise or 102  
left shift 800

5) explain the concept of type casting in python with example there are two types of type casting in python they are implicit or explicit type casting implicit type casting is done automatically by python explicit type casting is done manually by developer

```
[114]: #implicit type casting  
a=10  
print(type(a),a)  
#automatically type casts "int"  
b=1.0  
print(type(b),b)  
#automatically type casts "float"  
c=a+b  
print(type(c),c)  
#automatically type casts "float" after adding "int" and "float"
```

<class 'int'> 10  
<class 'float'> 1.0  
<class 'float'> 11.0

```
[116]: #explicit type casting  
a=float(10)  
print(type(a),a)  
#manually type casts as "float"  
b=int(1.0)  
print(type(b),b)  
#manually type casts as "int" even though it is "float"  
c=a+b  
print(type(c),c)  
#automatically type casts "float" after adding
```

<class 'float'> 10.0  
<class 'int'> 1  
<class 'float'> 11.0

6)how do conditional statements work in python ?Illustate with examplesconditional statements work in such a way that if the above if the condition is True then the below block of code is executed there are several way to combine and write conditional statements using if , else , elif , nested if-else

```
[122]: arr=[1,2,3,4,5,6]
no=int(input("enter a positive number to search"))
if no in arr:
    print("number is found")
elif no < 0:
    print("negative numbers not allowed")
else:
    print("number not found")
```

enter a positive number to search -1

negative numbers not allowed

```
[149]: arr1=["saabir",34,"student"]
n=int(input("enter index of the list positive number"))
if ((-n-1)<=(n)):
    if (arr1[n]):
        if type(arr1[n]) == int :
            print("The element is number")
        else:
            print("The element is not a number")
else:
    print("enter valid index")
```

enter index of the list positive number 1

The element is number

7)describe the different types loops in python an there use cases with examplethere are two different types of loops in python they are while loop and for loop we can use break and continue to control the flow of command

while loop block of code is repeatedly executed until the given condition is satisfied for loop block of code is repeatedly executed until the given iteration of a condition

```
[155]: #use cases and examples of while loop
#while loop is used when we dont know the number of times block of code much be
↪executed
#or a given condition is satisfied
# Example of a while loop
c = 0
while c < 3:
    print(c)
    c += 1
```

0  
1  
2

```
[157]: #use cases and examples of for loop
#for loop is used when we know the number of times block of code much be
↳executed and iterate over a elements of lists
#or a given executing a condition specific no of times

# Example of a for loop iterating over a list
names=["saabir","ayesha","ram"]
for i in names:
    print(i)
```

saabir  
ayesha  
ram

[ ]: