Untitled

August 27, 2023

Q1. Create one variable containing following type of data: (i) string (ii) list (iii) float (iv) tuple

[23]:
```
# string
string='Physics walla'
#list
lis=[1,2,3,2,1,3]
#float
floa=4.5
#tuple
tup=(1,2,3,4)
print(type(string),type(lis),type(floa),type(tup))
```

```
<class 'str'> <class 'list'> <class 'float'> <class 'tuple'>
```

Q2. Given are some following variables containing data: (i) var1 = ' ' (ii) var2 = '[DS , ML , Python]' (iii) var3 = ['DS' , 'ML' , 'Python'] (iv) var4 = 1. What will be the data type of the above given variable.

[10]:
```
var1 = ' '
var2 = '[ DS , ML , Python]'
var3 = [ 'DS' , 'ML' , 'Python' ]
var4 = 1
print(type(var1),type(var2),type(var3),type(var4))
```

```
<class 'str'> <class 'str'> <class 'list'> <class 'int'>
```

Q3. Explain the use of the following operators using an example: (i) / (ii) % (iii) // (iv) **

[14]:
```
# /
#divion
print(5/2)
# % extracts the remainder
print(5%2)
#// floor divisiom
print(5//2)
# ** power
print(5**2)
```

```
2.5
1
```

```
2
25
```

Q4. Create a list of length 10 of your choice containing multiple types of data. Using for loop print the element and its data type.

[24]:
```
lis.append(string)
lis.append(floa)
lis.append(var1)
lis.append(var4)
print(lis)
for i in lis:
    print(type(i))
```

```
[1, 2, 3, 2, 1, 3, 'Physics walla', 4.5, ' ', 1]
<class 'int'>
<class 'int'>
<class 'int'>
<class 'int'>
<class 'int'>
<class 'int'>
<class 'str'>
<class 'float'>
<class 'str'>
<class 'int'>
```

Q5. Using a while loop, verify if the number A is purely divisible by number B and if so then how many times it can be divisible.

[6]:
```
A=8
B=2
count=0
while A!=1:
    if A%B==0:
        A=A//B
        count+=1
print(count)
```

```
3
```

Q6. Create a list containing 25 int type data. Using for loop and if-else condition print if the element is divisible by 3 or not.

[7]:
```
lis=[]
for i in range(0,25):
    lis.append(i)
for i in lis:
    if i%3==0:
        print(i,'True')
```

```
    else:
        print(i,'False')
```

```
0 True
1 False
2 False
3 True
4 False
5 False
6 True
7 False
8 False
9 True
10 False
11 False
12 True
13 False
14 False
15 True
16 False
17 False
18 True
19 False
20 False
21 True
22 False
23 False
24 True
```

Q7. What do you understand about mutable and immutable data types? Give examples for both showing this property.

[8]:
```
#Mutable data types can be changed even after declaration like Lists
#example:
lis=[1,2,3,4]
print(lis)
lis[1]=5
print(lis)
```

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

[12]:
```
#immutable data types cannot be changed once declared like tuple
#example
tup=(1,2,3,4)
print(tup[1])
tup[1]=5
print(tup)
```

2

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[12], line 5
      3 tup=(1,2,3,4)
      4 print(tup[1])
----> 5 tup[1]=5
      6 print(tup)

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

[ ]: