

Filter files by name

/ work /

Name	Last Modified
• Assignment1.ipynb	6 minutes ago
• README.md	2 months ago
• sample-cod...	2 months ago

Launcher Assignment1.ipynb

Python 3 (ipykernel)

Question 1 - Create one variable containing following type of data

[1]:

```
#String
a = "Ammar"
```

[2]:

```
#List
b = [3, "Ammar", 5.6]
```

[3]:

```
#Float
c = 4.5
```

[4]:

```
#Tuple
d = (3, "Ammar", 5.6)
```

Question 2 - Given are some following variables containing data. What will be the data type of the above given variable.

[5]:

```
var1 = ' '
```

[6]:

```
print(type(var1))

<class 'str'>
```

[7]:

```
var2 = '[ DS , ML , Python]'
```

Filter files by name

/ work /

Name	Last Modified
Assignment1.ipynb	6 minutes ago
README.md	2 months ago
sample-cod...	2 months ago

Launcher x Assignment1.ipynb x +

Python 3 (ipykernel)

```
[8]: print(type(var2))
    <class 'str'>

[9]: var3 = [ 'DS' , 'ML' , 'Python' ]

[10]: print(type(var3))
    <class 'list'>

[11]: var4 = 1.

[12]: print(type(var4))
    <class 'float'>

Q3. Explain the use of the following operators using an example:

(i) / : Used for dividing

[13]: 4/2

[13]: 2.0

(ii) % : It is the Modulus operator, it returns the remainder.
```

Filter files by name

/ work /

Name	Last Modified
Assignment1.ipynb	6 minutes ago
README.md	2 months ago
sample-cod...	2 months ago

Launcher Assignment1.ipynb

Python 3 (ipykernel)

(ii) % : It is the Modulus operator, it returns the remainder.

```
[14]: 5 % 3
```

```
[14]: 2
```

(iii) // : Used for getting how many times a number will be divided

```
[15]: 40 // 5
```

```
[15]: 8
```

(iv) ** : Used for exponential.

```
[16]: 5 ** 4
```

```
[16]: 625
```

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.

```
[17]: Q4_List = [2, "Grey", 8.9, "Cat", (34, 56), False, 0.87, 4, 7, 'Phone']
```

```
[18]: Q4_L = len(Q4_List)
```

+

+

↑

↺

Filter files by name

/ work /

Name	Last Modified
Assignment1.ipynb	7 minutes ago
README.md	2 months ago
sample-cod...	2 months ago

Launcher

Assignment1.ipynb

+

+

✂

📄

📄

▶

■

↺

▶▶

Markdown

Python 3 (ipykernel)

```

[19]: print(Q4_L)
      10

[20]: for item in Q4_List:
      print(f"{item} - {type(item)}")

      2 - <class 'int'>
      Grey - <class 'str'>
      8.9 - <class 'float'>
      Cat - <class 'str'>
      (34, 56) - <class 'tuple'>
      False - <class 'bool'>
      0.87 - <class 'float'>
      4 - <class 'int'>
      7 - <class 'int'>
      Phone - <class 'str'>

      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.

[23]: No_A = int(input("Enter number A: "))
      No_B = int(input("Enter number B: "))

      print(f"Entered Numbers are: \n A = {No_A} \n B = {No_B}")
      print("Calculating.....")
  
```

Filter files by name

/ work /

Name	Last Modified
Assignment1.ipynb	7 minutes ago
README.md	2 months ago
sample-cod...	2 months ago

Launcher Assignment1.ipynb

Python 3 (ipykernel)

```
[23]: No_A = int(input("Enter number A: "))
      No_B = int(input("Enter number B: "))

      print(f"Entered Numbers are: \n A = {No_A} \n B = {No_B}")
      print("Calculating.....")

      if No_A < No_B:
          print("B should be smaller than A, try again")
      else:
          count = 0
          while No_A % No_B == 0 and No_A != 0:
              No_C = No_A // No_B
              count += 1

          if count > 0:
              print(f"A is purely divisible by B {count} times {No_C}")
          else:
              print("A is not purely divisible by B.")
```

Enter number A: 90
Enter number B: 100
Entered Numbers are:
A = 90
B = 100
Calculating.....
B should be smaller than A, try again

Filter files by name

/ work /

Name	Last Modified
Assignment1.ipynb	7 minutes ago
README.md	2 months ago
sample-cod...	2 months ago

Launcher x Assignment1.ipynb x +

Python 3 (ipykernel)

```
[24]: 50%5

[24]: 0

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.

[25]: I_List = [ 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25 ]

for i in I_List:
    if i % 3 == 0:
        print(f"{i} is divisible by 3")
    else:
        print(f"{i} is NOT divisible by 3")

1 is NOT divisible by 3
2 is NOT divisible by 3
3 is divisible by 3
4 is NOT divisible by 3
5 is NOT divisible by 3
6 is divisible by 3
7 is NOT divisible by 3
8 is NOT divisible by 3
9 is divisible by 3
10 is NOT divisible by 3
11 is NOT divisible by 3
12 is divisible by 3
13 is NOT divisible by 3
14 is NOT divisible by 3
15 is divisible by 3
16 is NOT divisible by 3
17 is NOT divisible by 3
18 is divisible by 3
19 is NOT divisible by 3
20 is NOT divisible by 3
21 is divisible by 3
22 is NOT divisible by 3
23 is NOT divisible by 3
24 is divisible by 3
25 is NOT divisible by 3
```

Filter files by name

/ work /

Name	Last Modified
Assignment1.ipynb	8 minutes ago
README.md	2 months ago
sample-cod...	2 months ago

Launcher Assignment1.ipynb

Python 3 (ipykernel)

```
print([i for i in range(1, 26) if i % 3 == 0])
```

1 is NOT divisible by 3
2 is NOT divisible by 3
3 is divisble by 3
4 is NOT divisible by 3
5 is NOT divisible by 3
6 is divisble by 3
7 is NOT divisible by 3
8 is NOT divisible by 3
9 is divisble by 3
10 is NOT divisible by 3
11 is NOT divisible by 3
12 is divisble by 3
13 is NOT divisible by 3
14 is NOT divisible by 3
15 is divisble by 3
16 is NOT divisible by 3
17 is NOT divisible by 3
18 is divisble by 3
19 is NOT divisible by 3
20 is NOT divisible by 3
21 is divisble by 3
22 is NOT divisible by 3
23 is NOT divisible by 3
24 is divisble by 3
25 is NOT divisible by 3

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

Filter files by name

/ work /

Name	Last Modified
Assignment1.ipynb	8 minutes ago
README.md	2 months ago
sample-cod...	2 months ago

Launcher x Assignment1.ipynb x +

Python 3 (ipykernel)

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

Mutable: These are the ones in which there elements can be altered / changed anytime example: Lists

```
[26]: I_L = [1,2,3,4]
      print(I_L)
      I_L[1] = 20
      print(I_L)
```

```
[1, 2, 3, 4]
[1, 20, 3, 4]
```

Immutable: Once made, elements cannot be changed example: Tuples

```
[27]: A = (2,3,4)
      print(A)
      A[1] = 7 #Will give error
```

```
(2, 3, 4)
```

```
-----
TypeError                                Traceback (most recent call last)
Cell In[27], line 3
      1 A = (2,3,4)
      2 print(A)
----> 3 A[1] = 7 #Will give error
```


+ + + +

Filter files by name

/ work /

Name	Last Modified
• Assignment1.ipynb	8 minutes ago
• README.md	2 months ago
• sample-cod...	2 months ago

Launcher x Assignment1.ipynb x +

save + copy paste undo redo Markdown v

Python 3 (ipykernel)

```
-----
TypeError                                Traceback (most recent call last)
Cell In[27], line 3
      1 A = (2,3,4)
      2 print(A)
----> 3 A[1] = 7 #Will give error

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

[28]: 10//5

[28]: 2

```
[29]: sum = 0
      for i in range(1,6):
          sum += i
      print (sum)
```

15

```
[30]: count = 0
      while count < 5:
          print(count)
          count += 1
```

0
1
2

Filter files by name

/ work /

Name	Last Modified
Assignment1.ipynb	8 minutes ago
README.md	2 months ago
sample-cod...	2 months ago

Launcher Assignment1.ipynb

Python 3 (ipykernel)

```
print (sum)

15

[30]: count = 0
      while count < 5:
        print(count)
        count += 1

0
1
2
3
4
```