

Launch Meeting x Home Page - x ML-Assignment x ML-Assignment x ML-Assignment x M-Assignment x

localhost:8888/notebooks/M-Assignment-2.1.ipynb

jupyter M-Assignment-2.1 Last Checkpoint: an hour ago (unsaved changes)

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel)

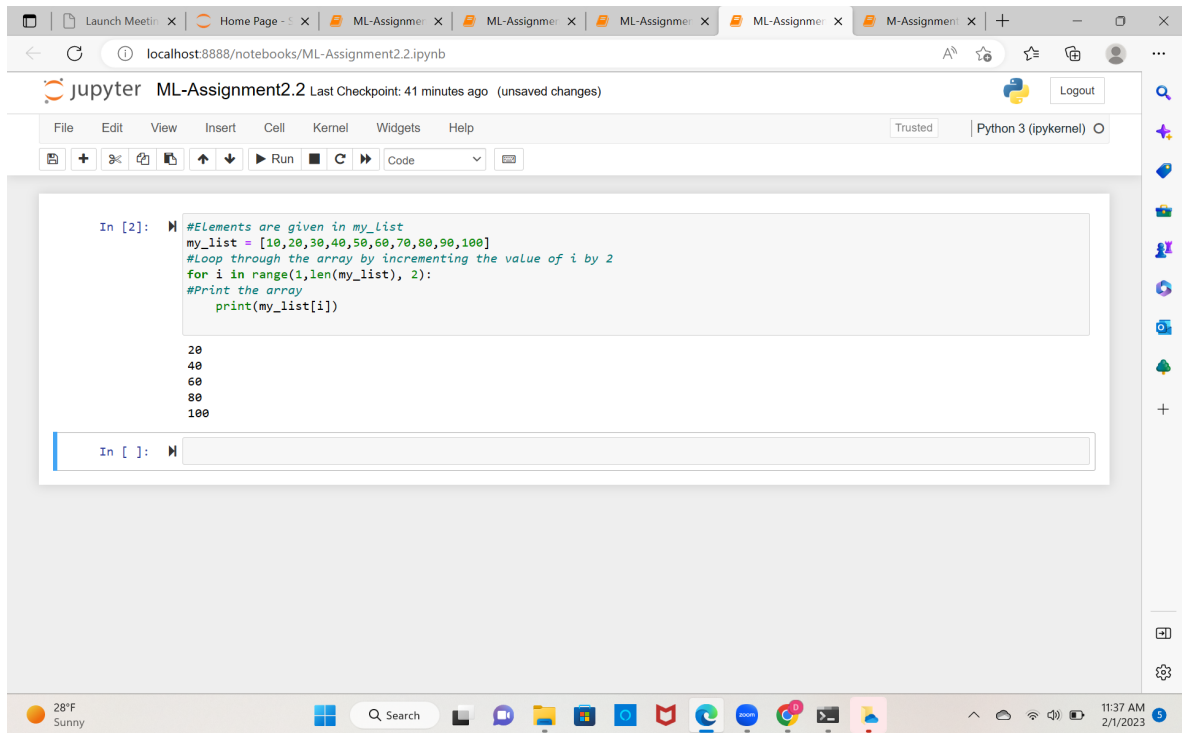
```
In [3]: #Number of rows
rows = 5
for i in range(0,rows):
    # Nested Loop for each column
    for j in range(0,i+1):
        print(" ",end=" ")
    #Print Star
    print("\n")

for i in range(rows,0, -1):
    #Nested Loop for each column
    for j in range(0,i - 1):
        print(" ",end=" ")
    #Print Star
    print("\n")

*
* *
* * *
* * * *
* * * * *
* * * * *
* * * *
* * *
* *
*
*
```

In []:

2. The elements are written in a array named as my_list. By using loop , array incremented the value of i by 2. Finally by print statement the odd indexes are printed.



The screenshot shows a Jupyter Notebook window titled "ML-Assignment2.2" with a last checkpoint of 41 minutes ago. The code cell contains the following Python code:

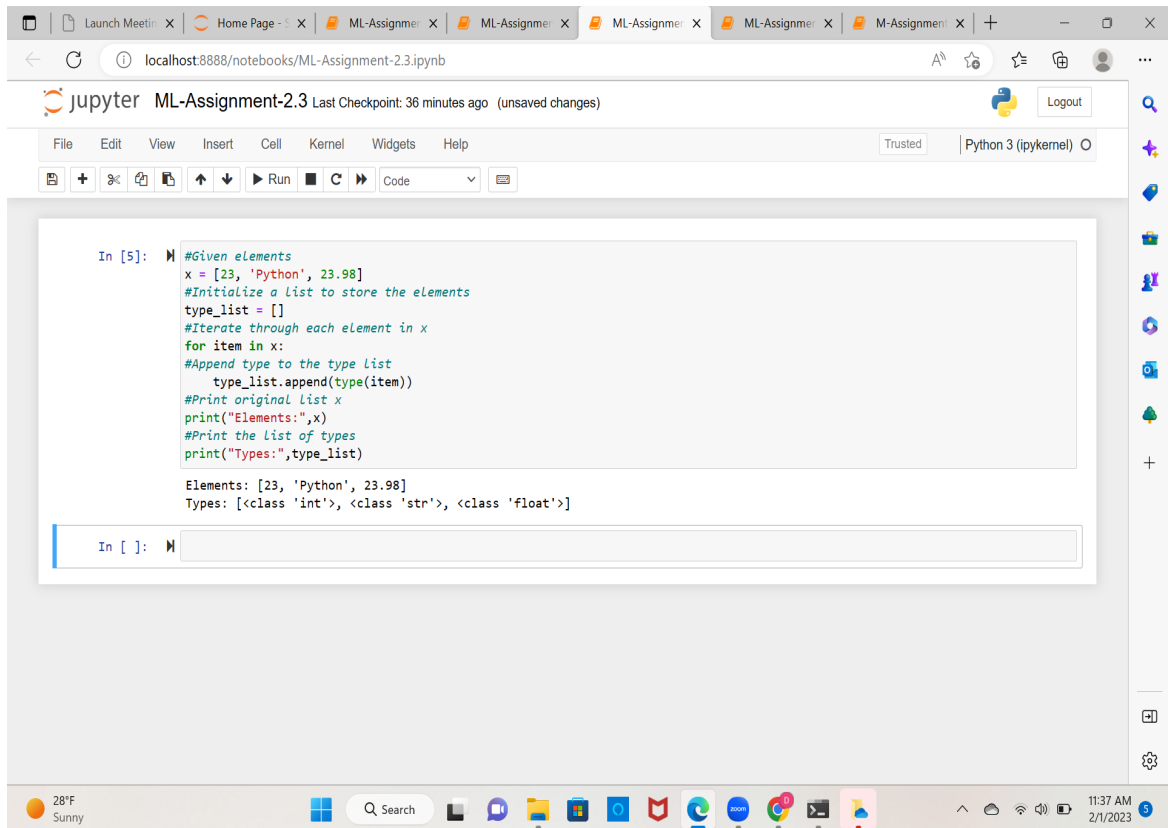
```
In [2]: #Elements are given in my_list
my_list = [10,20,30,40,50,60,70,80,90,100]
#Loop through the array by incrementing the value of i by 2
for i in range(1,len(my_list), 2):
    #Print the array
    print(my_list[i])
```

The output of the code cell shows the elements at odd indices of the list: 20, 40, 60, 80, and 100.

```
20
40
60
80
100
```

The Jupyter Notebook interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help), a toolbar with icons for file operations and execution, and a sidebar with various tool icons. The bottom status bar shows the system temperature as 28°F, Sunny, and the time as 11:37 AM on 2/1/2023.

3. Given elements in list. The type_list statement is used to store the elements. Iteration through each element in x is done by using for loop. Now i have add type(item) to type_list by using append function. By using print statement the elements in x are printed and type of the element is printed as output.



The screenshot shows a Jupyter Notebook titled "ML-Assignment-2.3" running on a local host. The code cell contains the following Python code with comments:

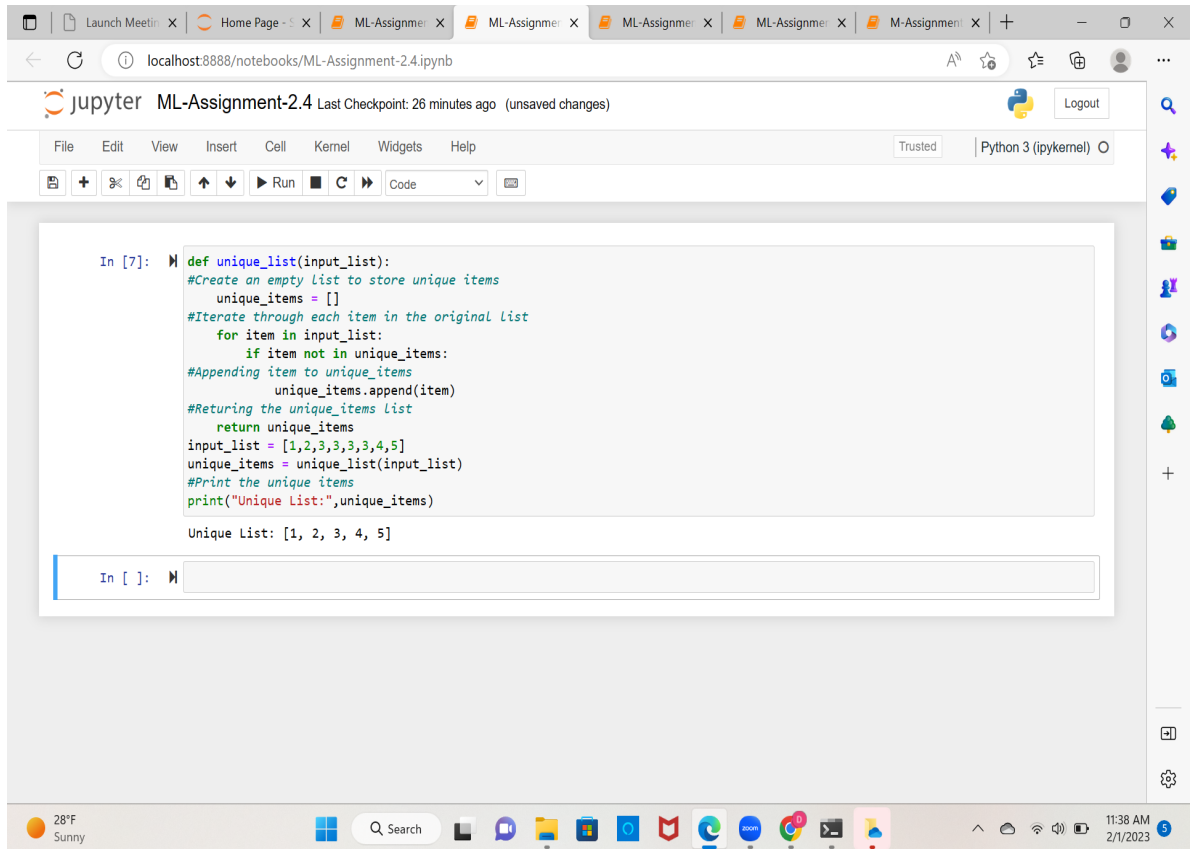
```
In [5]: #Given elements
x = [23, 'Python', 23.98]
#Initialize a list to store the elements
type_list = []
#Iterate through each element in x
for item in x:
    #Append type to the type list
    type_list.append(type(item))
#Print original list x
print("Elements:",x)
#Print the list of types
print("Types:",type_list)
```

The output of the code is displayed below the cell:

```
Elements: [23, 'Python', 23.98]
Types: [<class 'int'>, <class 'str'>, <class 'float'>]
```

The Jupyter Notebook interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help), a toolbar with icons for file operations and execution, and a status bar at the bottom showing the system clock and weather.

4. Firstly Input_list is defined in the unique_list. Now i have created an empty list to store unique items as unique_items. For iteration through each item in original list i have used for loop . If the item is not in unique_items ,we can add items by append function. And then unique_items are returned. Finally printed the unique items from unique_list.



5. Define `case_count` in a string. Check if the character is in upper case by using `if` statement and check if the character is in lower case by using `elif` statement. Then return the number of uppercase letters and number of lowercase letters from the given string. Then call the functions. Finally print the number of uppercase characters and lower case character by using `print` statement.

Launch Meeting x Home Page - x ML-Assignmen x ML-Assignmen x ML-Assignmen x ML-Assignmen x M-Assignment x +

localhost:8888/notebooks/ML-Assignmen2.5.ipynb

jupyter ML-Assignmen2.5 Last Checkpoint: 20 minutes ago (unsaved changes) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel)

Run Code

```
In [2]: def case_count(string):
        upper_count = 0
        lower_count = 0
        #Loop through the string
        for char in string:
            #Check if the character is in upper case
            if char.isupper():
                upper_count += 1
            #Check if the character is in Lowe case
            elif char.islower():
                lower_count += 1
        #Return the number of upper-case and Lower-case Letters
        return (upper_count, lower_count)
        #Given input string
        string = 'The quick Brow Fox'
        #Call the function
        upper_count, lower_count = case_count(string)
        #Print the results
        print("Number of Upper-case characters:",upper_count)
        print("Number of Lower-case characters:",lower_count)
```

Number of Upper-case characters: 3
Number of Lower-case characters: 12

In []:

28°F Sunny 11:38 AM 2/1/2023