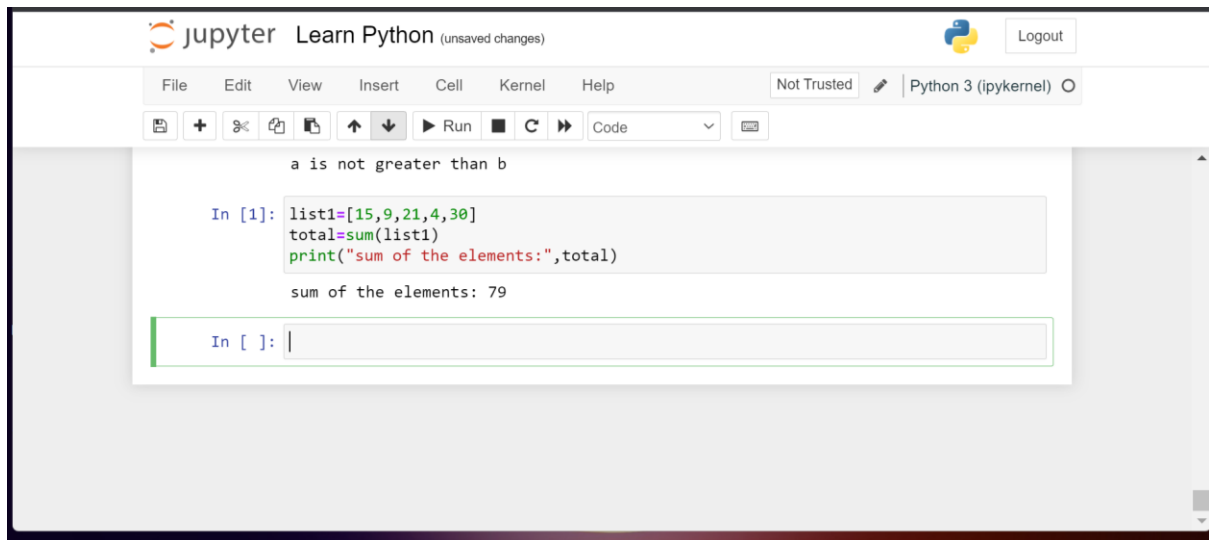


## 1. Write a Python program to sum all the items in a list.



A screenshot of a Jupyter Notebook interface. The top bar shows the Jupyter logo, the text "Learn Python (unsaved changes)", a Python logo, and a "Logout" button. Below the top bar is a menu bar with "File", "Edit", "View", "Insert", "Cell", "Kernel", and "Help". To the right of the menu bar are "Not Trusted" and "Python 3 (ipykernel)" with a dropdown arrow. Below the menu bar is a toolbar with icons for saving, adding, deleting, and running cells, along with a "Run" button and a "Code" dropdown. The main area shows a code cell with the following code:

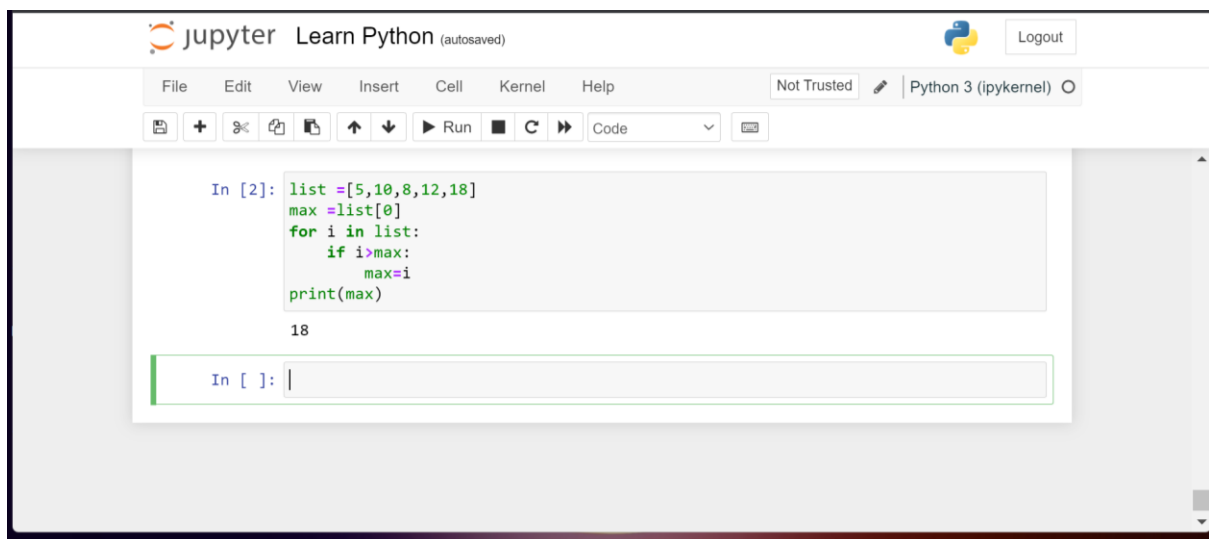
```
a is not greater than b

In [1]: list1=[15,9,21,4,30]
        total=sum(list1)
        print("sum of the elements:",total)

        sum of the elements: 79

In [ ]: |
```

## 2. Write a Python program to get the largest number from a list.



A screenshot of a Jupyter Notebook interface. The top bar shows the Jupyter logo, the text "Learn Python (autosaved)", a Python logo, and a "Logout" button. Below the top bar is a menu bar with "File", "Edit", "View", "Insert", "Cell", "Kernel", and "Help". To the right of the menu bar are "Not Trusted" and "Python 3 (ipykernel)" with a dropdown arrow. Below the menu bar is a toolbar with icons for saving, adding, deleting, and running cells, along with a "Run" button and a "Code" dropdown. The main area shows a code cell with the following code:

```
In [2]: list =[5,10,8,12,18]
        max =list[0]
        for i in list:
            if i>max:
                max=i
        print(max)

        18

In [ ]: |
```

## 3. Write a Python program to count the number of strings from a given list of strings. The string length is 2 or more and the first and last characters are the same.

The original list is: [5, 6, 2, 3, 9]  
The sliced list is: []

```
In [30]: list=("578","eliminate","away","upon","everyday")
len(list)

Out[30]: 5
```

In [ ]:

Page 1 of 2 32 of 82 words English (India)

#### 4. Write a Python program to remove duplicates from a list.

```
[2, 8, 7, 5, 4]
```

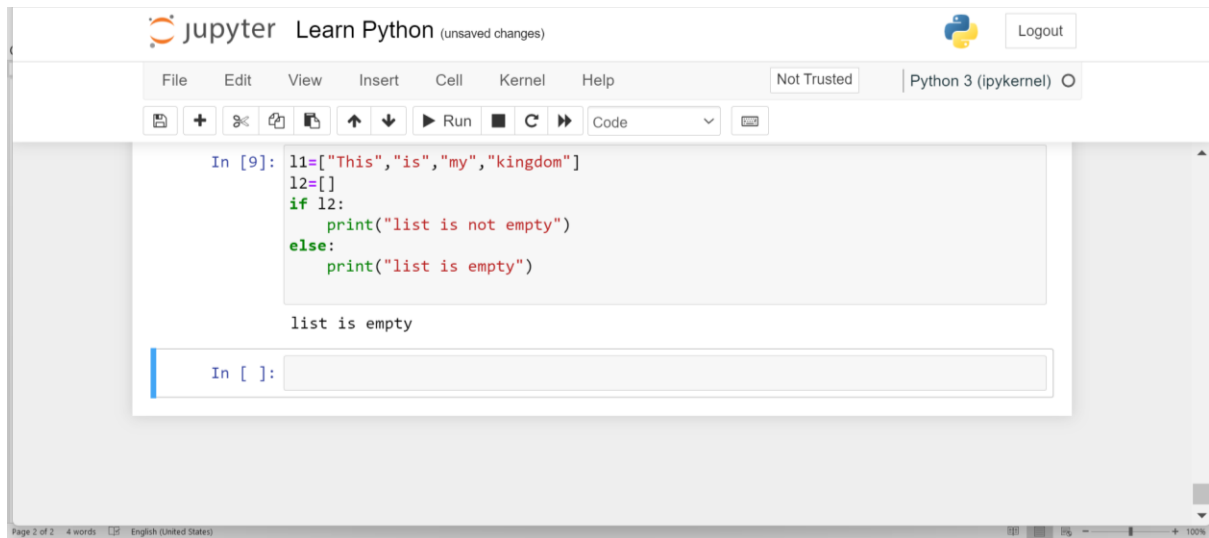
```
In [7]: list1=[8,7,12,12,7,10,9,7,10]
list2=[]
[list2.append(i) for i in list1 if i not in list2]
print(list2)

[8, 7, 12, 10, 9]
```

In [ ]:

Page 2 of 2 3 words English (United States)

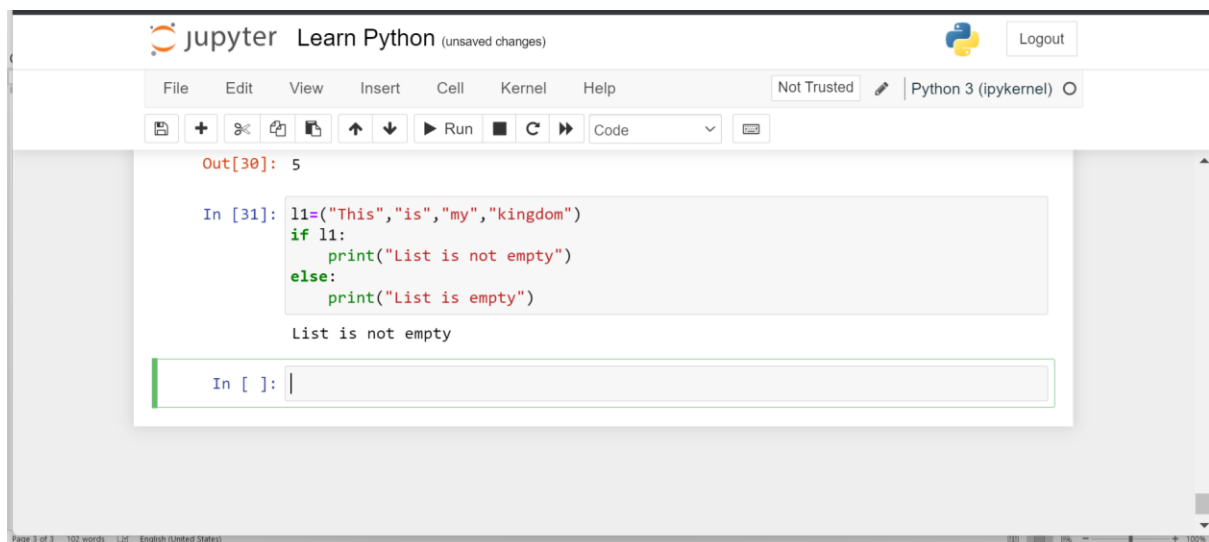
## 5. Write a Python program to check if a list is empty or not.



A screenshot of a Jupyter Notebook interface. The top bar shows the Jupyter logo, the text "Learn Python (unsaved changes)", a Python logo, and a "Logout" button. Below this is a menu bar with "File", "Edit", "View", "Insert", "Cell", "Kernel", and "Help". To the right of the menu bar are "Not Trusted" and "Python 3 (ipykernel)". Below the menu bar is a toolbar with icons for saving, adding, deleting, and running cells, along with a "Code" dropdown menu. The main area contains a code cell with the following Python code:

```
In [9]: l1=["This","is","my","kingdom"]
        l2=[]
        if l2:
            print("list is not empty")
        else:
            print("list is empty")
```

The output of the code cell is "list is empty". Below the code cell is an input field with the text "In [ ]:".



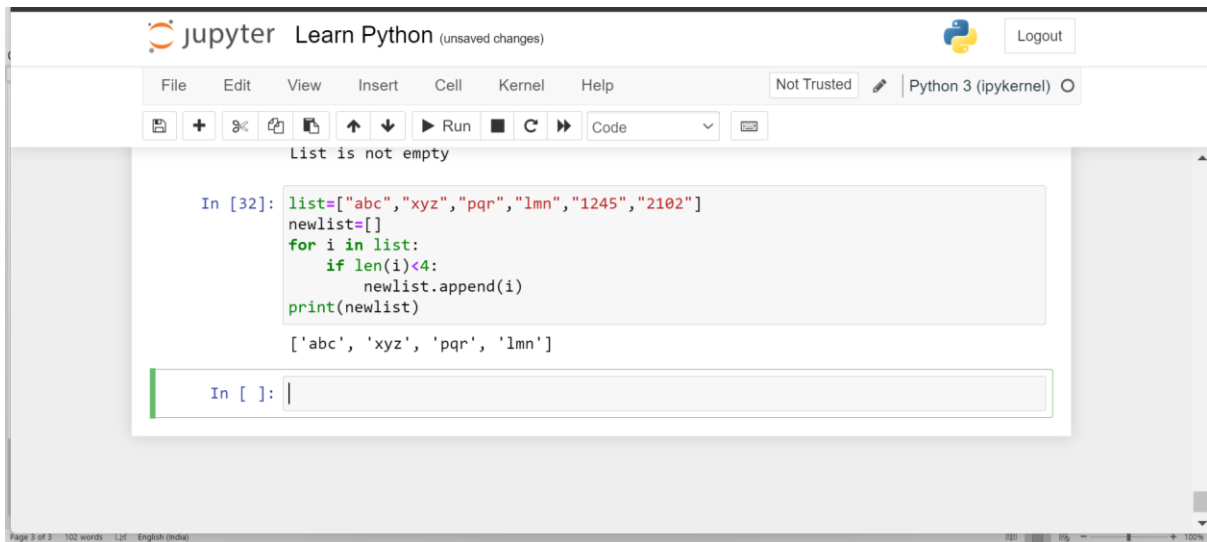
A screenshot of a Jupyter Notebook interface. The top bar shows the Jupyter logo, the text "Learn Python (unsaved changes)", a Python logo, and a "Logout" button. Below this is a menu bar with "File", "Edit", "View", "Insert", "Cell", "Kernel", and "Help". To the right of the menu bar are "Not Trusted" and "Python 3 (ipykernel)". Below the menu bar is a toolbar with icons for saving, adding, deleting, and running cells, along with a "Code" dropdown menu. The main area contains a code cell with the following Python code:

```
Out[30]: 5

In [31]: l1=("This","is","my","kingdom")
        if l1:
            print("List is not empty")
        else:
            print("List is empty")
```

The output of the code cell is "List is not empty". Below the code cell is an input field with the text "In [ ]:".

## 6. Write a Python program to filter the list if the length of the character is < 4



A Jupyter Notebook interface with the title "Learn Python (unsaved changes)". The menu bar includes File, Edit, View, Insert, Cell, Kernel, and Help. The status bar shows "Not Trusted" and "Python 3 (ipykernel)". The code cell contains the following Python code:

```
List is not empty

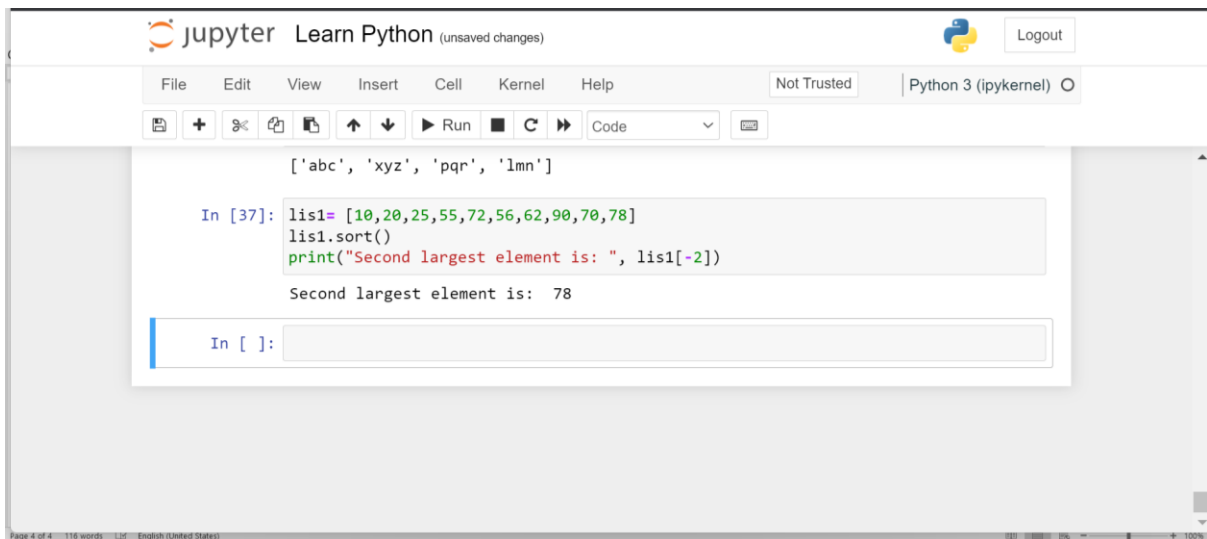
In [32]: list=["abc","xyz","pqr","lmn","1245","2102"]
newlist=[]
for i in list:
    if len(i)<4:
        newlist.append(i)
print(newlist)

['abc', 'xyz', 'pqr', 'lmn']

In [ ]: |
```

The output of the code is displayed above the input prompt: `['abc', 'xyz', 'pqr', 'lmn']`. The bottom status bar indicates "Page 3 of 3", "102 words", and "English (media)".

**7. Write a Python program to find the second largest number in a list.**



A Jupyter Notebook interface with the title "Learn Python (unsaved changes)". The menu bar includes File, Edit, View, Insert, Cell, Kernel, and Help. The status bar shows "Not Trusted" and "Python 3 (ipykernel)". The code cell contains the following Python code:

```
['abc', 'xyz', 'pqr', 'lmn']

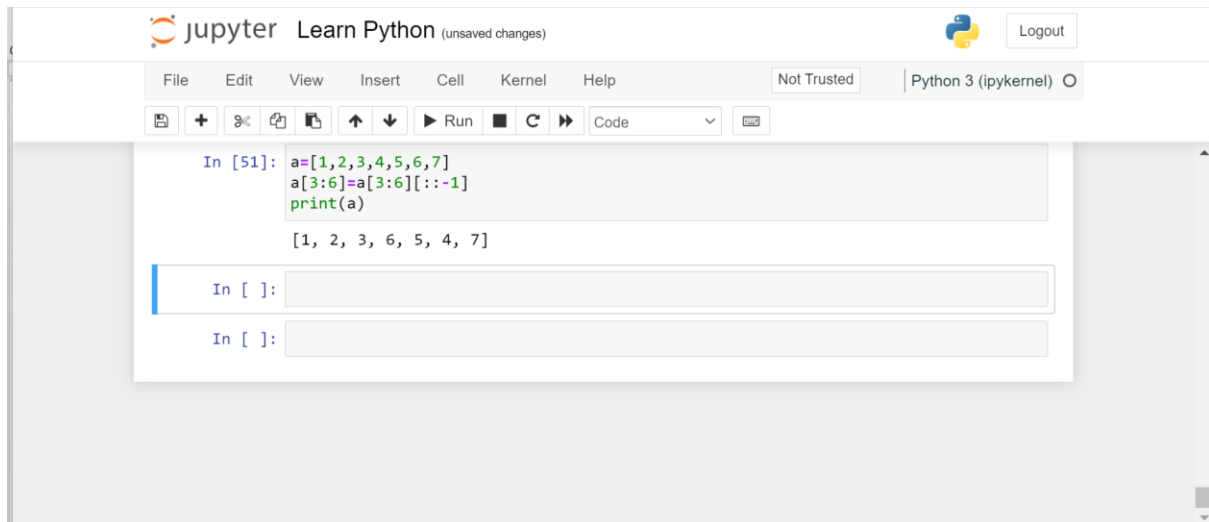
In [37]: lis1=[10,20,25,55,72,56,62,90,70,78]
lis1.sort()
print("Second largest element is: ", lis1[-2])

Second largest element is: 78

In [ ]: |
```

The output of the code is displayed above the input prompt: `Second largest element is: 78`. The bottom status bar indicates "Page 4 of 4", "116 words", and "English (United States)".

**8. Write a Python program to reverse a list at a specific location.**



The image shows a Jupyter Notebook interface with the title "jupyter Learn Python (unsaved changes)". The top bar includes a "Logout" button and a "Python 3 (ipykernel)" label. The menu bar contains "File", "Edit", "View", "Insert", "Cell", "Kernel", and "Help". Below the menu is a toolbar with icons for saving, adding, deleting, and running code. The main area displays a code cell with the following Python code:

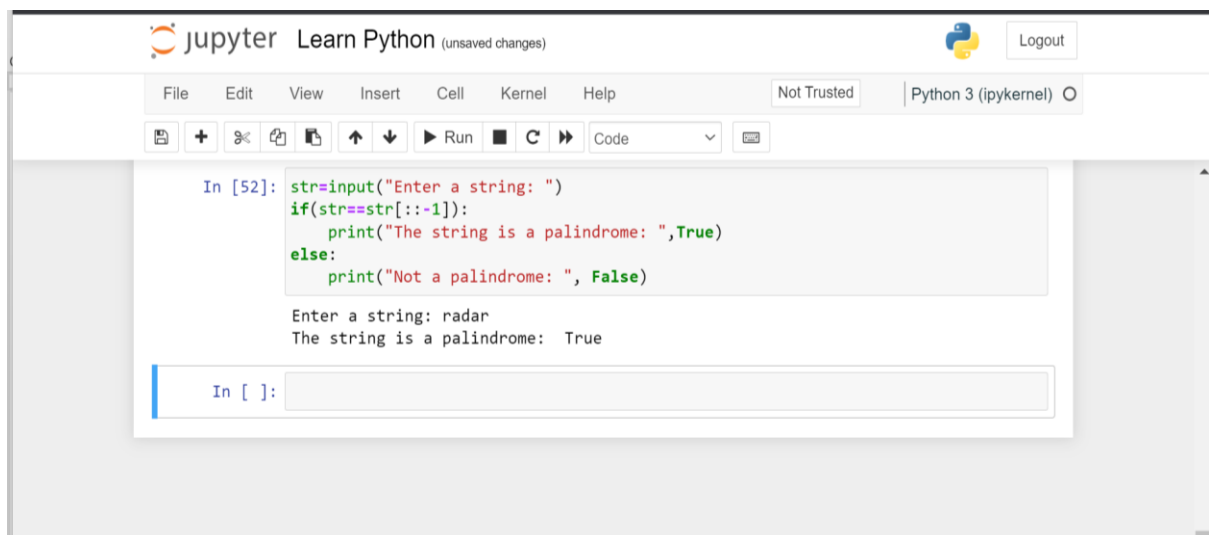
```
In [51]: a=[1,2,3,4,5,6,7]
a[3:6]=a[3:6][::-1]
print(a)
```

The output of the code is displayed below the code cell:

```
[1, 2, 3, 6, 5, 4, 7]
```

Below the output, there are two empty input fields for the next code cells, each labeled "In [ ]:".

**9. Write a Python program to check if a list is a palindrome or not. Return true otherwise false.**



The image shows a Jupyter Notebook interface with the title "jupyter Learn Python (unsaved changes)". The top bar includes a "Logout" button and a "Python 3 (ipykernel)" label. The menu bar contains "File", "Edit", "View", "Insert", "Cell", "Kernel", and "Help". Below the menu is a toolbar with icons for saving, adding, deleting, and running code. The main area displays a code cell with the following Python code:

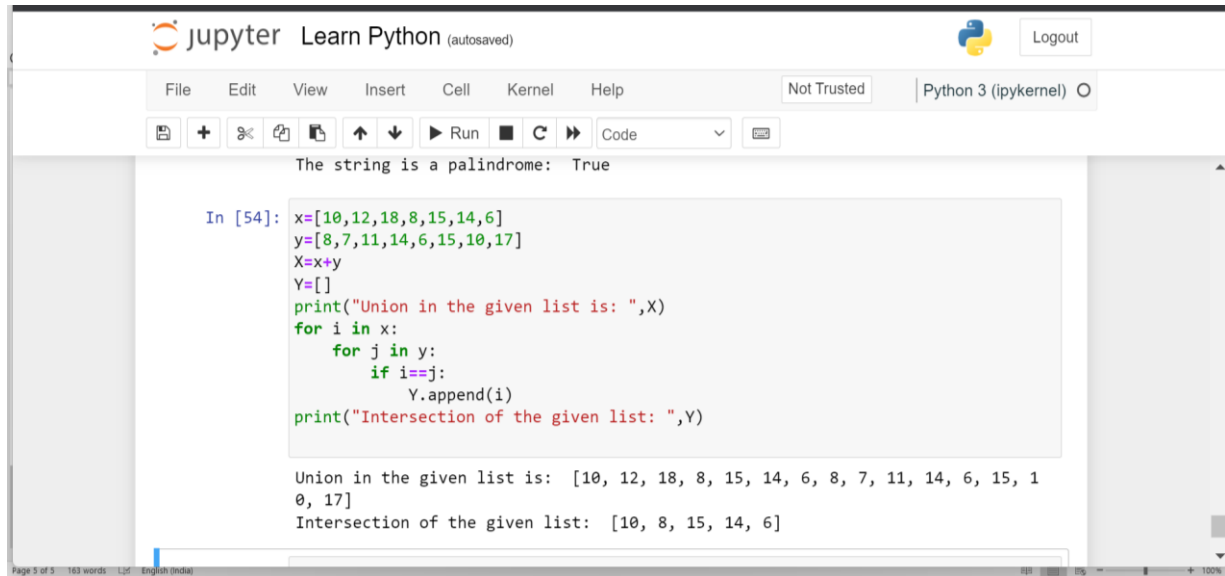
```
In [52]: str=input("Enter a string: ")
if(str==str[::-1]):
    print("The string is a palindrome: ",True)
else:
    print("Not a palindrome: ", False)
```

The output of the code is displayed below the code cell:

```
Enter a string: radar
The string is a palindrome: True
```

Below the output, there is an empty input field for the next code cell, labeled "In [ ]:".

## 10. Write a Python a program to find the union and intersection of two lists.



A screenshot of a Jupyter Notebook interface. The top bar shows 'jupyter Learn Python (autosaved)' and a 'Logout' button. The menu bar includes File, Edit, View, Insert, Cell, Kernel, and Help. The status bar indicates 'Not Trusted' and 'Python 3 (ipykernel)'. The notebook contains a code cell with the following Python code:

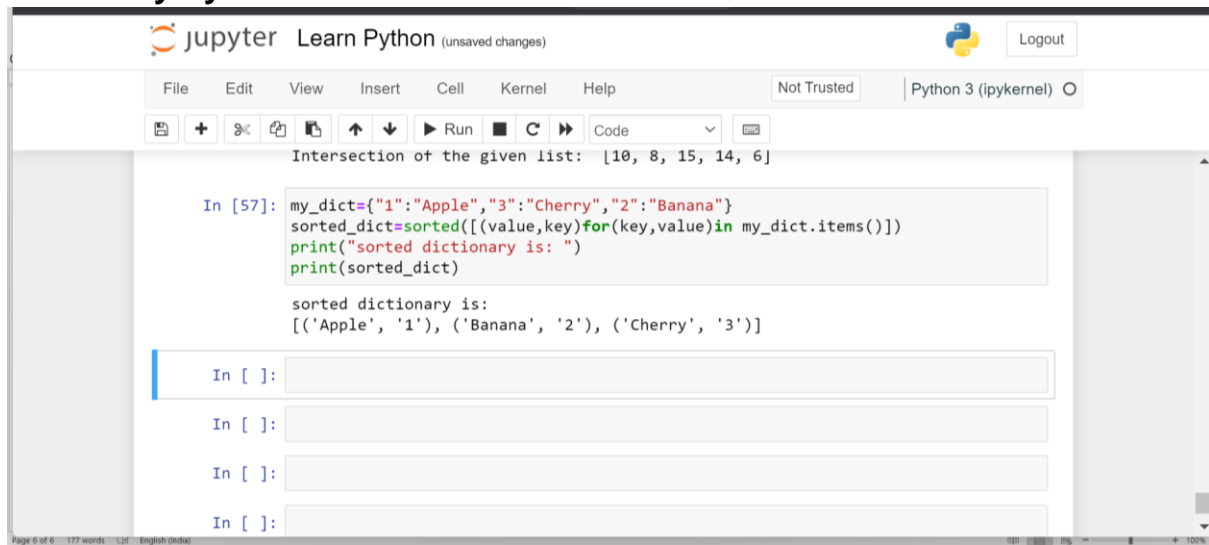
```
The string is a palindrome: True

In [54]: x=[10,12,18,8,15,14,6]
         y=[8,7,11,14,6,15,10,17]
         X=x+y
         Y=[]
         print("Union in the given list is: ",X)
         for i in x:
             for j in y:
                 if i==j:
                     Y.append(i)
         print("Intersection of the given list: ",Y)

Union in the given list is:  [10, 12, 18, 8, 15, 14, 6, 8, 7, 11, 14, 6, 15, 1
0, 17]
Intersection of the given list:  [10, 8, 15, 14, 6]
```

The output of the code is displayed below the cell, showing the union and intersection of the two lists.

## 11. Write a Python script to sort (ascending and descending) a dictionary by value



A screenshot of a Jupyter Notebook interface. The top bar shows 'jupyter Learn Python (unsaved changes)' and a 'Logout' button. The menu bar includes File, Edit, View, Insert, Cell, Kernel, and Help. The status bar indicates 'Not Trusted' and 'Python 3 (ipykernel)'. The notebook contains a code cell with the following Python code:

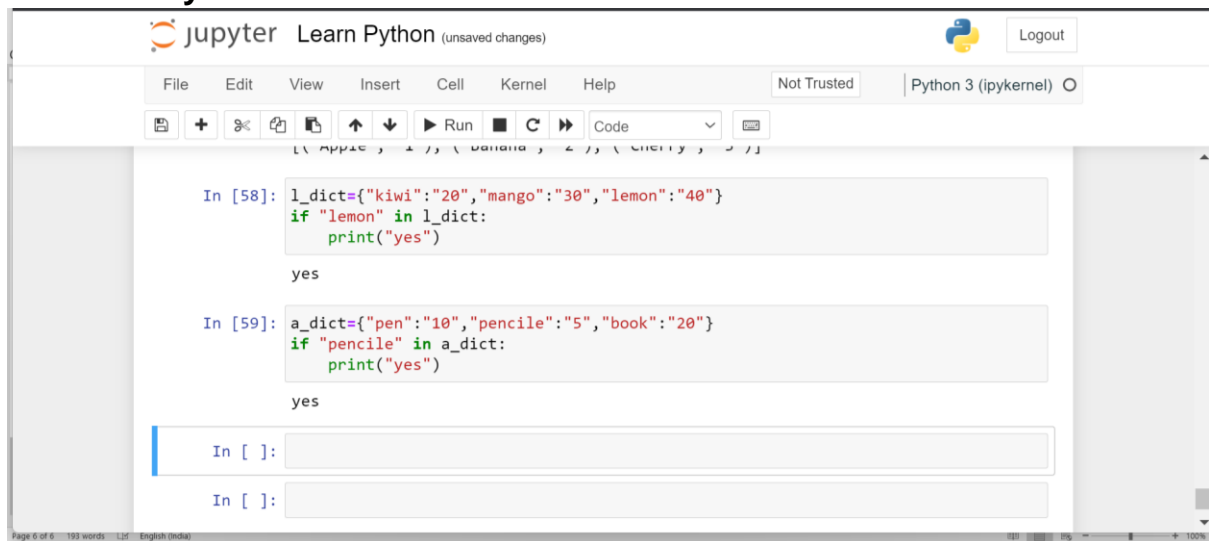
```
Intersection of the given list:  [10, 8, 15, 14, 6]

In [57]: my_dict={"1":"Apple","3":"Cherry","2":"Banana"}
         sorted_dict=sorted([(value,key)for(key,value)in my_dict.items()])
         print("sorted dictionary is: ")
         print(sorted_dict)

sorted dictionary is:
[('Apple', '1'), ('Banana', '2'), ('Cherry', '3')]
```

Below the code cell, there are four empty input boxes for additional code or output.

## 12. Write a Python script to check whether a given key already exists in a dictionary



A screenshot of a Jupyter Notebook interface. The top bar shows the Jupyter logo, 'Learn Python (unsaved changes)', a Python logo, and a 'Logout' button. Below the top bar is a menu bar with 'File', 'Edit', 'View', 'Insert', 'Cell', 'Kernel', and 'Help'. To the right of the menu bar are 'Not Trusted' and 'Python 3 (ipykernel)' buttons. Below the menu bar is a toolbar with icons for file operations, running, and code execution. The main area contains two code cells. The first cell, labeled 'In [58]:', contains the following code: 

```
l_dict={"kiwi":"20","mango":"30","lemon":"40"}
if "lemon" in l_dict:
    print("yes")
```

 The output of this cell is 'yes'. The second cell, labeled 'In [59]:', contains the following code: 

```
a_dict={"pen":"10","pencil":"5","book":"20"}
if "pencil" in a_dict:
    print("yes")
```

 The output of this cell is 'yes'. Below these cells are two empty input fields labeled 'In [ ]:'. The bottom status bar shows 'Page 6 of 6', '193 words', and 'English (India)'.

```
In [58]: l_dict={"kiwi":"20","mango":"30","lemon":"40"}
if "lemon" in l_dict:
    print("yes")

yes

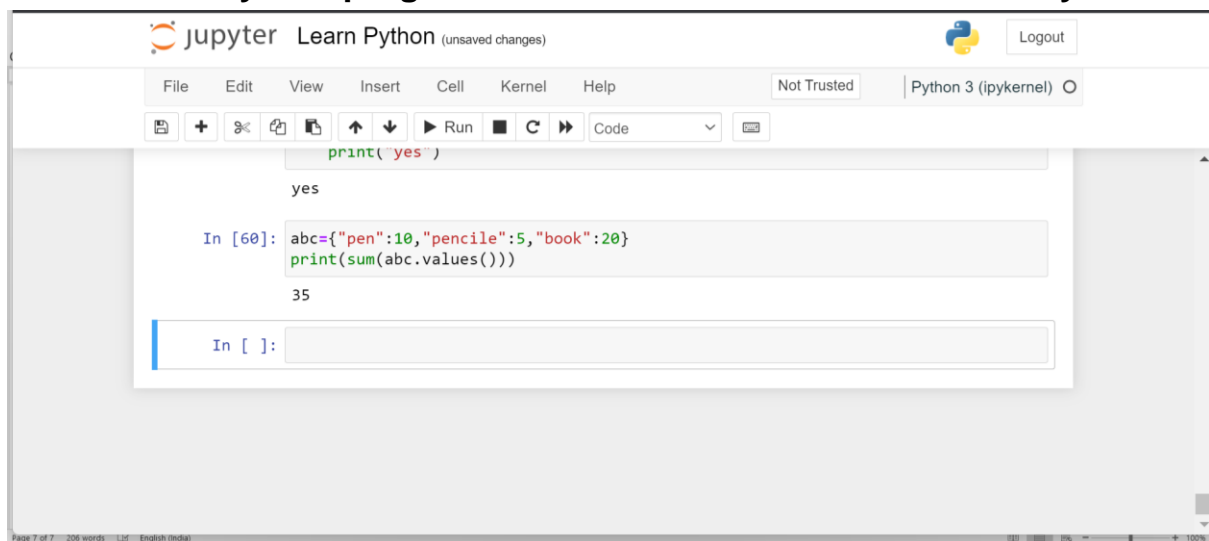
In [59]: a_dict={"pen":"10","pencil":"5","book":"20"}
if "pencil" in a_dict:
    print("yes")

yes

In [ ]:

In [ ]:
```

## 13. Write a Python program to sum all the values in a dictionary.



A screenshot of a Jupyter Notebook interface. The top bar shows the Jupyter logo, 'Learn Python (unsaved changes)', a Python logo, and a 'Logout' button. Below the top bar is a menu bar with 'File', 'Edit', 'View', 'Insert', 'Cell', 'Kernel', and 'Help'. To the right of the menu bar are 'Not Trusted' and 'Python 3 (ipykernel)' buttons. Below the menu bar is a toolbar with icons for file operations, running, and code execution. The main area contains two code cells. The first cell, labeled 'In [60]:', contains the following code: 

```
abc={"pen":10,"pencil":5,"book":20}
print(sum(abc.values()))
```

 The output of this cell is '35'. Below this cell is an empty input field labeled 'In [ ]:'. The bottom status bar shows 'Page 7 of 7', '206 words', and 'English (India)'.

```
print('yes')

yes

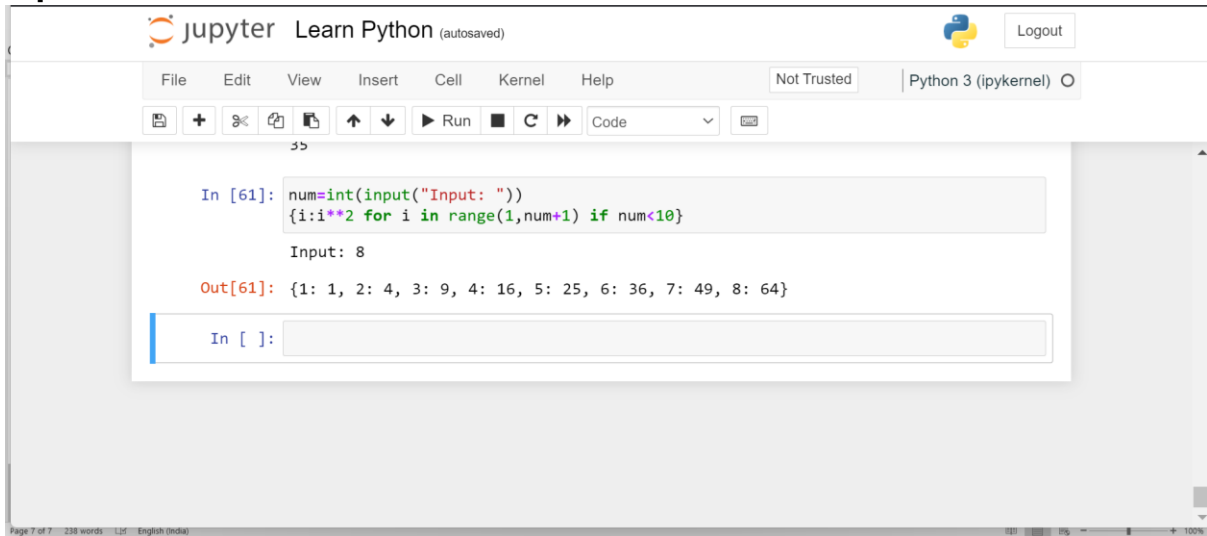
In [60]: abc={"pen":10,"pencil":5,"book":20}
print(sum(abc.values()))

35

In [ ]:
```

## 14. Write a Python program to create a dictionary with a number and its corresponding square from 1 to input number. And also check if the

## input number is less than 10



The screenshot shows a Jupyter Notebook interface with the title "Learn Python (autosaved)". The menu bar includes File, Edit, View, Insert, Cell, Kernel, and Help. The status bar indicates "Not Trusted" and "Python 3 (ipykernel)". The code cell contains the following Python code:

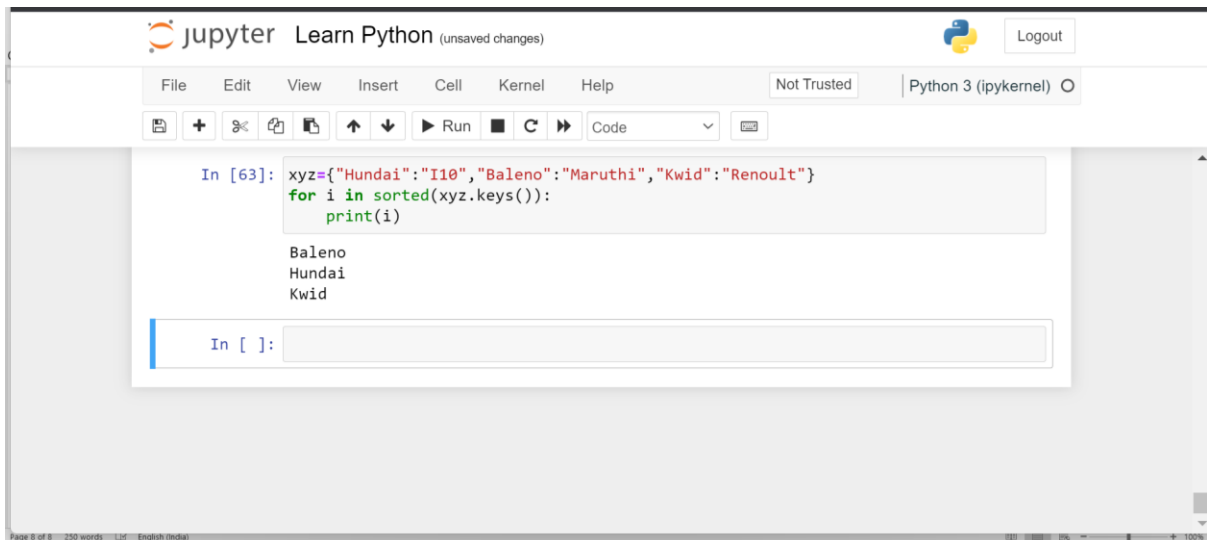
```
In [61]: num=int(input("Input: "))
         {i:i**2 for i in range(1,num+1) if num<10}
```

The output of the code is:

```
Input: 8
Out[61]: {1: 1, 2: 4, 3: 9, 4: 16, 5: 25, 6: 36, 7: 49, 8: 64}
```

The input field shows "Input: 8". The output field shows "{1: 1, 2: 4, 3: 9, 4: 16, 5: 25, 6: 36, 7: 49, 8: 64}". The input field is currently empty and labeled "In [ ]:".

## 15. Write a Python program to sort a given dictionary by key.



The screenshot shows a Jupyter Notebook interface with the title "Learn Python (unsaved changes)". The menu bar includes File, Edit, View, Insert, Cell, Kernel, and Help. The status bar indicates "Not Trusted" and "Python 3 (ipykernel)". The code cell contains the following Python code:

```
In [63]: xyz={"Hundai":"I10", "Baleno":"Maruthi", "Kwid":"Renoult"}
         for i in sorted(xyz.keys()):
             print(i)
```

The output of the code is:

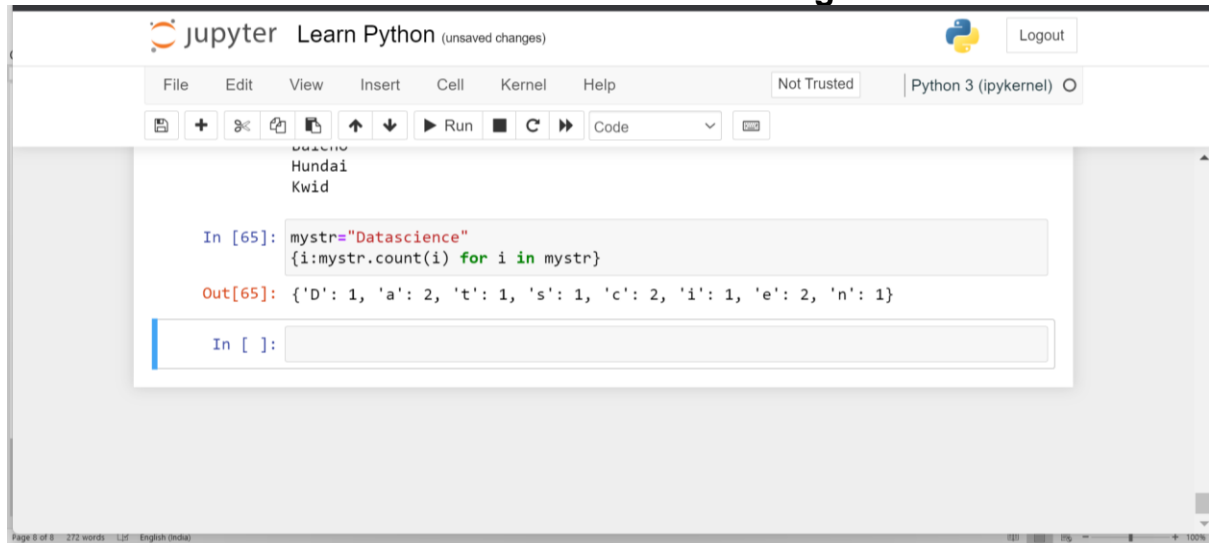
```
Baleno
Hundai
Kwid
```

The input field shows "In [ ]:".

## 16. Write a Python program to create a dictionary from a string.



**Note: Track the count of the letters from the string.**



A screenshot of a Jupyter Notebook interface. The top bar shows the Jupyter logo, the text "Learn Python (unsaved changes)", a Python logo, and a "Logout" button. Below this is a menu bar with "File", "Edit", "View", "Insert", "Cell", "Kernel", and "Help". To the right of the menu bar are "Not Trusted" and "Python 3 (ipykernel)" with a refresh icon. A toolbar contains icons for saving, adding, deleting, and running cells, along with a "Code" dropdown and a "Run" button. The notebook area shows a code cell with the following content:

```
dicts
Hundai
Kwid

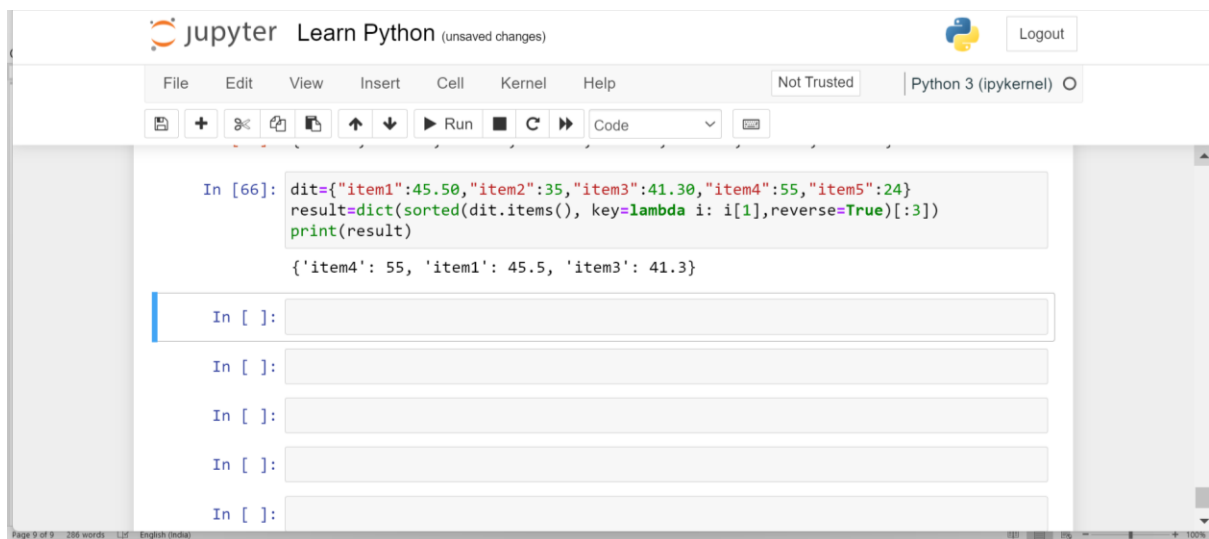
In [65]: mystr="DataScience"
         {i:mystr.count(i) for i in mystr}

Out[65]: {'D': 1, 'a': 2, 't': 1, 's': 1, 'c': 2, 'i': 1, 'e': 2, 'n': 1}

In [ ]:
```

The bottom status bar indicates "Page 8 of 8", "272 words", "1.0f", and "English (India)".

**17. Write a Python program to get the top three items in a shop.**



A screenshot of a Jupyter Notebook interface. The top bar shows the Jupyter logo, the text "Learn Python (unsaved changes)", a Python logo, and a "Logout" button. Below this is a menu bar with "File", "Edit", "View", "Insert", "Cell", "Kernel", and "Help". To the right of the menu bar are "Not Trusted" and "Python 3 (ipykernel)" with a refresh icon. A toolbar contains icons for saving, adding, deleting, and running cells, along with a "Code" dropdown and a "Run" button. The notebook area shows a code cell with the following content:

```
In [66]: dit={"item1":45.50,"item2":35,"item3":41.30,"item4":55,"item5":24}
         result=dict(sorted(dit.items(), key=lambda i: i[1],reverse=True)[:3])
         print(result)

         {'item4': 55, 'item1': 45.5, 'item3': 41.3}

In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
```

The bottom status bar indicates "Page 9 of 9", "285 words", "1.0f", and "English (India)".