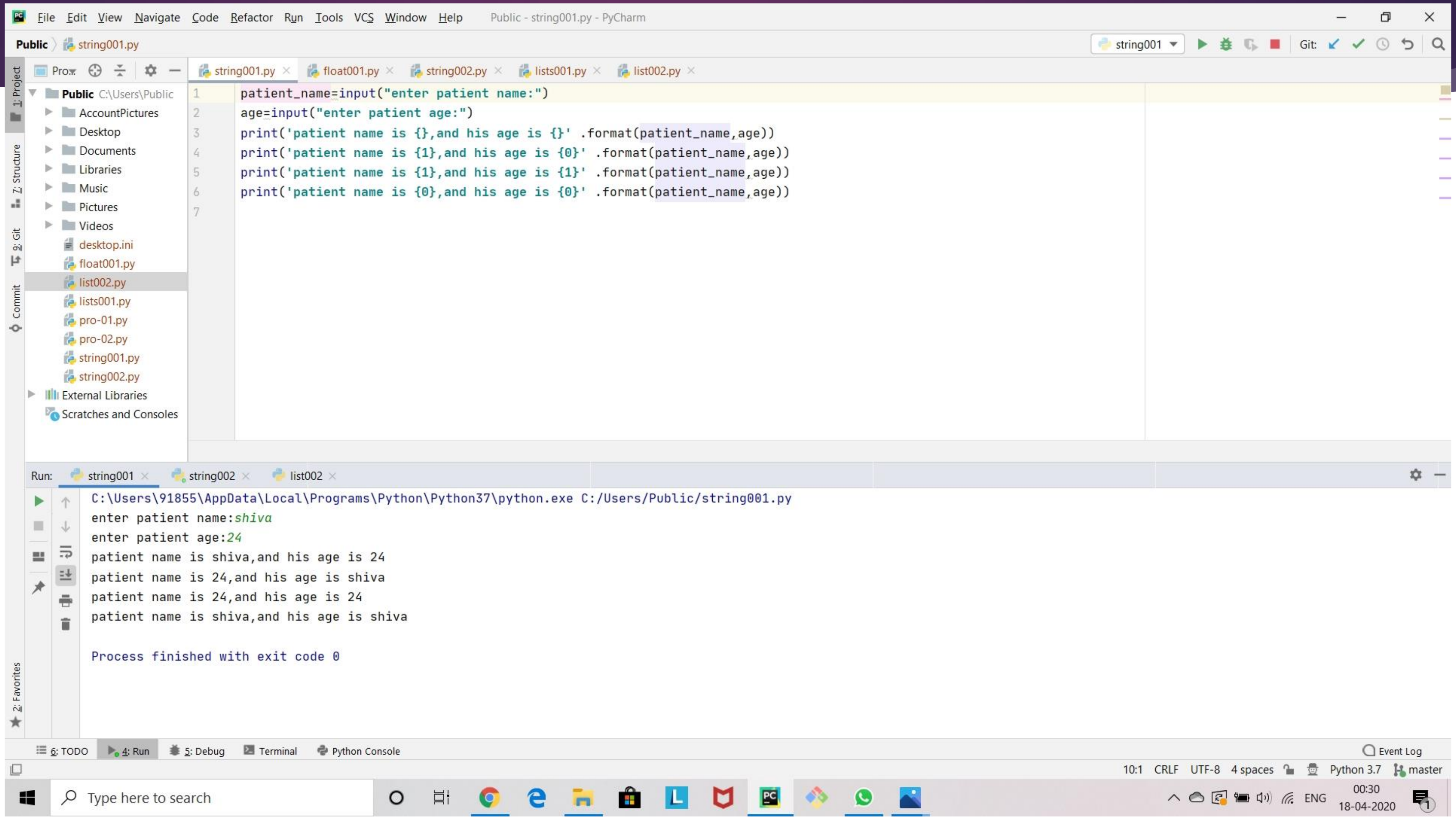
Python_02

String formating using . Format() function

- #string formating using .format()
- #lets take a program that takes the information of the patient in a hospital
- #And prints the result.
- patient_name=input("enter patient name:")
- age=input("enter patient age:")
- print('patient name is {},and his age is {}' .format(patient_name,age))

- #usually .format puts the words in the string as we supplied in the .format.
- #but we can inter change these format as our wish as shown below.
- print('patient name is {1},and his age is {0}'.format(patient_name,age))
- #we can also print the same charater in both place as show below
- print('patient name is {1},and his age is {1}' .format(patient_name,age))
- print('patient name is {0},and his age is {0}' .format(patient_name,age))

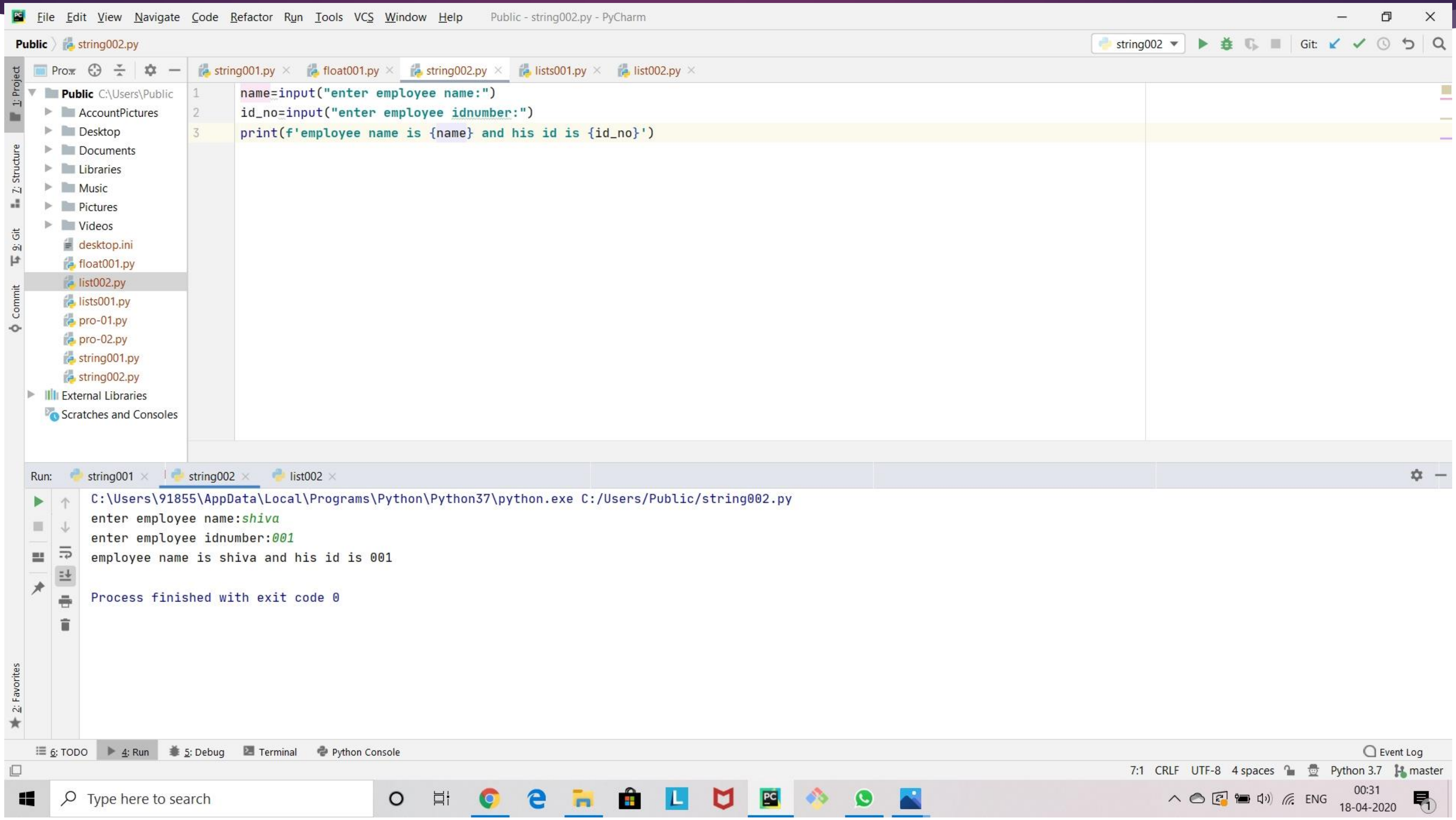


```
patient_name=input("enter patient name:")
age=input("enter patient age:")
print('patient name is {},and his age is {}' .format(patient_name,age))
print('patient name is {1},and his age is {0}' .format(patient_name,age))
print('patient name is {1},and his age is {1}' .format(patient_name,age))
print('patient name is {0},and his age is {0}' .format(patient_name,age))
```

```
C:\Users\91855\AppData\Local\Programs\Python\Python37\python.exe C:/Users/Public/string001.py
enter patient name:shiva
enter patient age:24
patient name is shiva,and his age is 24
patient name is 24,and his age is shiva
patient name is 24,and his age is 24
patient name is shiva,and his age is shiva

Process finished with exit code 0
```

String formating using f-strings

- #String formating using f-strings
- #f-strings is the new feature in the python 3.6
- #A program to print the basic information of the employee by entering details
- name=input("enter employee name:")
- id_no=input("enter employee idnumber:")
- print(f'employee name is {name} and his id is {id_no}')



```
name=input("enter employee name:")
id_no=input("enter employee idnumber:")
print(f'employee name is {name} and his id is {id_no}')
```

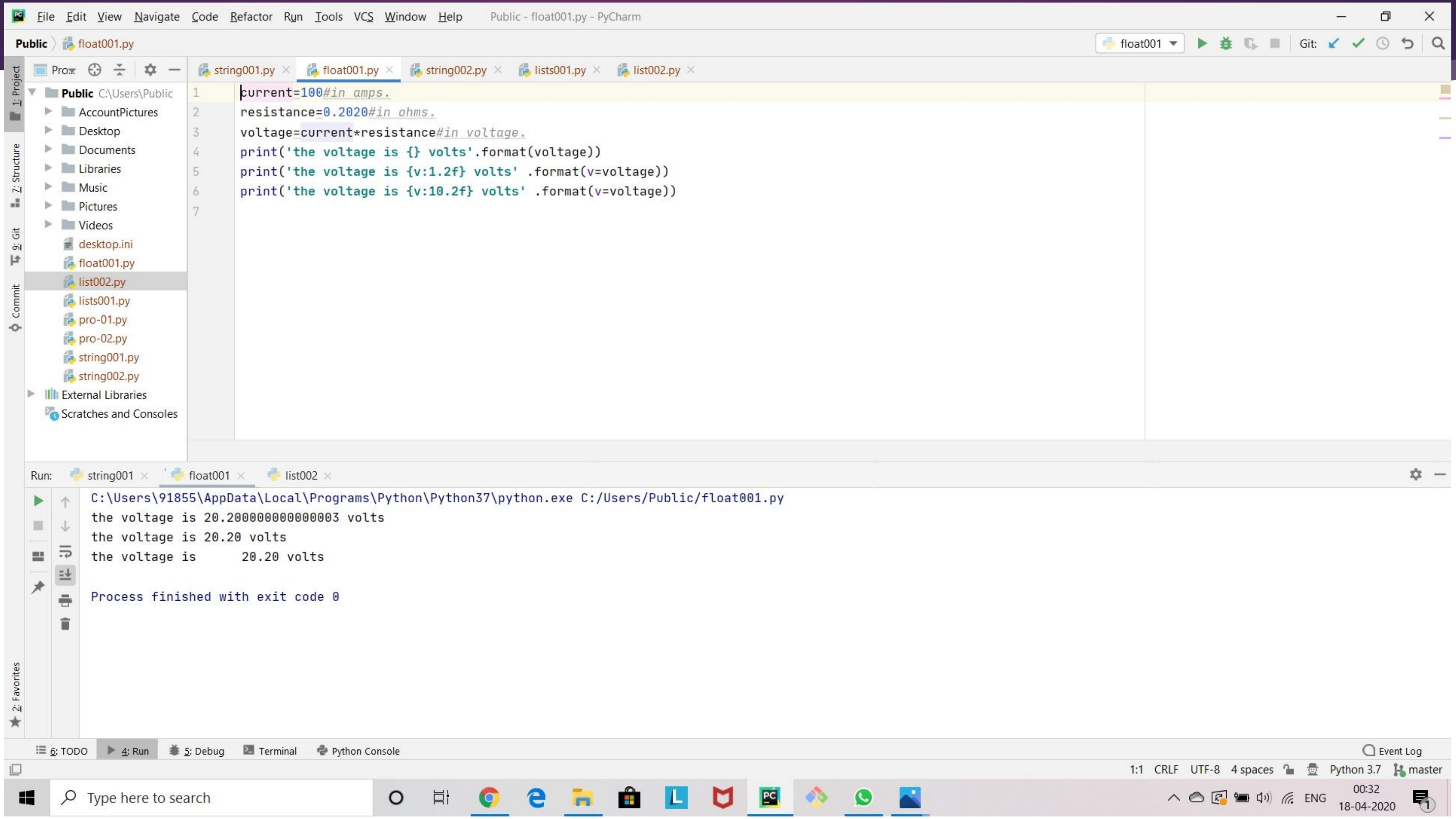
```
C:\Users\91855\AppData\Local\Programs\Python\Python37\python.exe C:/Users/Public/string002.py
enter employee name:shiva
enter employee idnumber:001
employee name is shiva and his id is 001

Process finished with exit code 0
```

Float formating

- #float formatting"{value:width.precisionf}
- #program to explain how float formatting works.
- current=100#in amps.
- Resistance=0.2020#in ohms.
- voltage=current*resistance#in voltage.
- print('the voltage is {} volts'.format(voltage))

- #if u want to windup the value of the voltage to limited decimal values.
- #for that we can use float formatting.
- print('the voltage is {v:1.2f} volts' .format(v=voltage))
- #The value '1' in the flower brackets is the width in which the result should fit.
- #for example if we increase the width of the result there is change in the format.

- #it is printedprint('the voltage is {v:10.2f} volts' .format(v=voltage))
- #there is a space in between is and the result which is cauSed by increasing width.



```
current=100#in amps.
resistance=0.2020#in ohms.
voltage=current*resistance#in voltage.
print('the voltage is {} volts'.format(voltage))
print('the voltage is {v:1.2f} volts' .format(v=voltage))
print('the voltage is {v:10.2f} volts' .format(v=voltage))
```

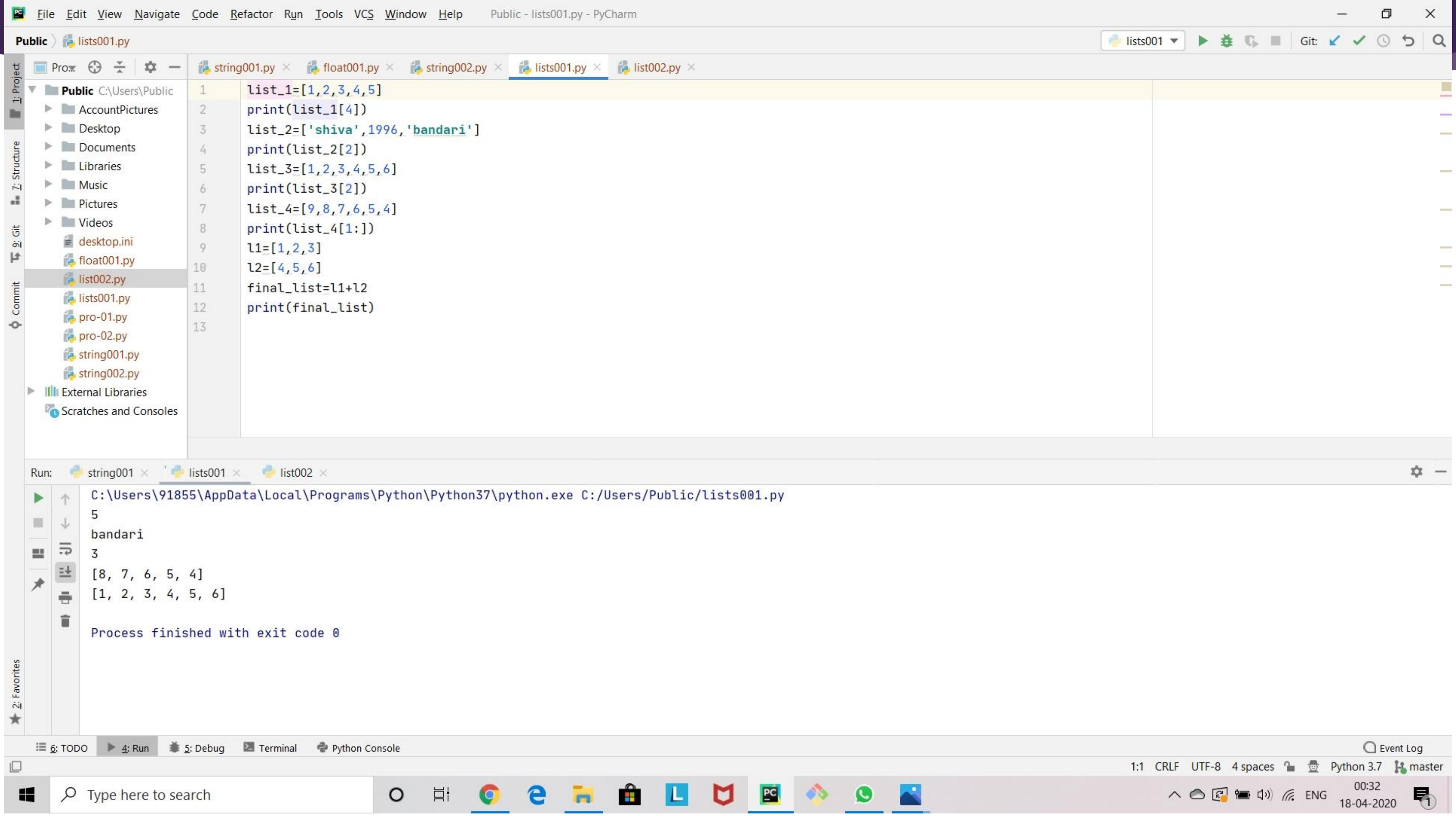
```
C:\Users\91855\AppData\Local\Programs\Python\Python37\python.exe C:/Users/Public/float001.py
the voltage is 20.200000000000003 volts
the voltage is 20.20 volts
the voltage is      20.20 volts

Process finished with exit code 0
```

Lists-01

- #lists is a ordered sequence that can hold a variety of object types.
- #they use [] brackets and commas to separate objects in the lists.
- #lists supports both indexing and slicing.
- #Here is a program to show how lists works.
- list_1=[1,2,3,4,5]
- print(list_1[4])

- #The placing in lists starts from 0.
- #we can also place variety of datatypes in a lists.
- list_2=['shiva',1996,'bandari’]
- print(list_2[2])
- #program to show how indexing works in lists.
- list_3=[1,2,3,4,5,6]
- print(list_3[2])

- #program to show how slicing works in slicing works in a lists.
- list_4=[9,8,7,6,5,4]
- print(list_4[1:])
- #program to show how concatination works in lists.
- l1=[1,2,3]
- l2=[4,5,6]
- final_list=l1+l2
- print(final_list)



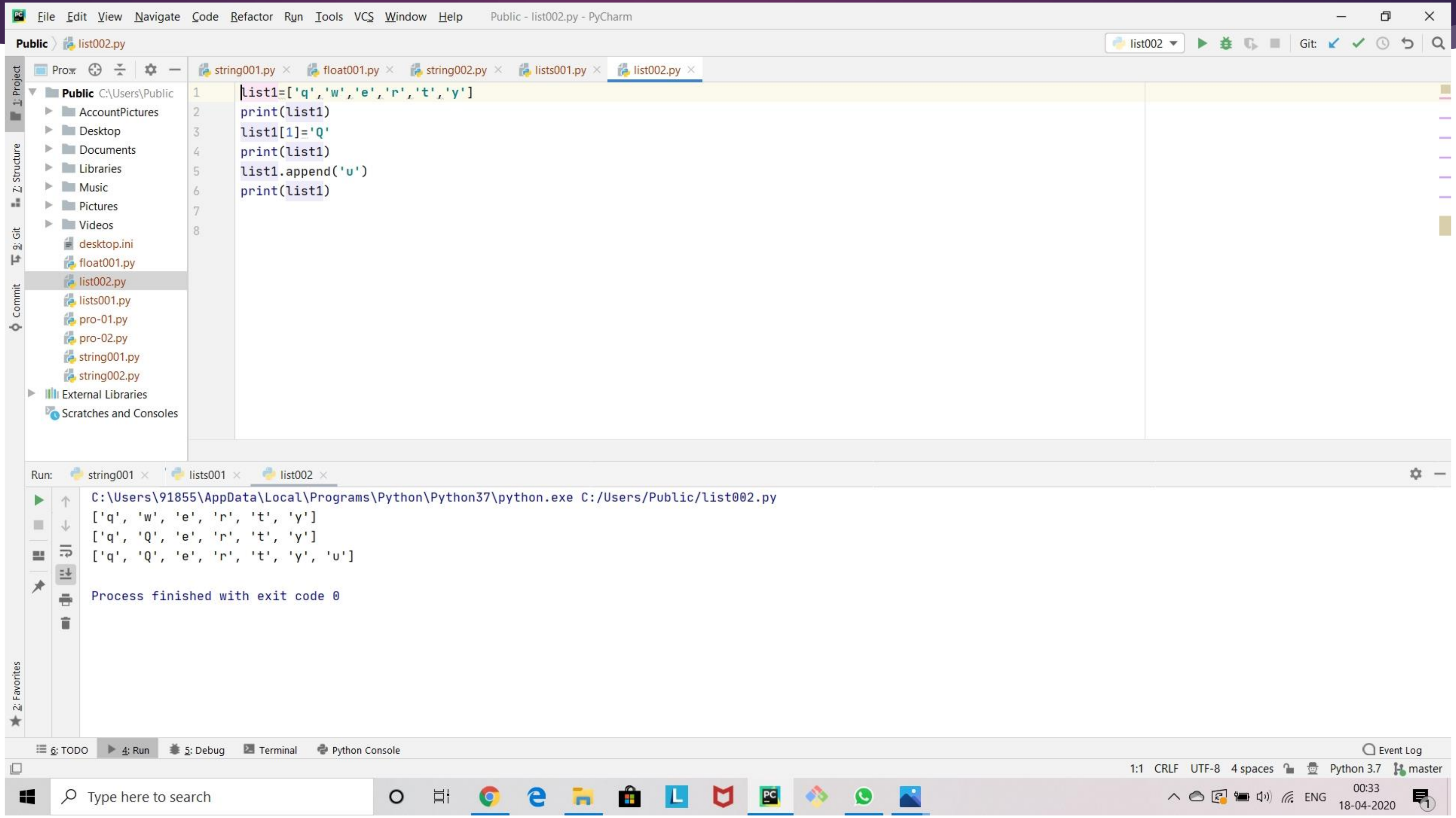
```
list_1=[1,2,3,4,5]
print(list_1[4])
list_2=['shiva',1996,'bandari']
print(list_2[2])
list_3=[1,2,3,4,5,6]
print(list_3[2])
list_4=[9,8,7,6,5,4]
print(list_4[1:])
l1=[1,2,3]
l2=[4,5,6]
final_list=l1+l2
print(final_list)
```

```
C:\Users\91855\AppData\Local\Programs\Python\Python37\python.exe C:/Users/Public/lists001.py
5
bandari
3
[8, 7, 6, 5, 4]
[1, 2, 3, 4, 5, 6]

Process finished with exit code 0
```

List-02

- #lists are mutatable(changable)
- #here a program to show how a list can be changed.
- list1=['q','w','e','r','t','y’]
- print(list1)
- list1[1]='Q’
- print(list1)

- #here u can view that 'w' in the first list is replaced by 'Q’.
- #we can add a new object to the above list by using 'append' function.
- list1.append('u’)
- print(list1)
- #there you can see in the output in the picture,in which a new letter 'u' is added.



```
list1=['q','w','e','r','t','y']
print(list1)
list1[1]='Q'
print(list1)
list1.append('u')
print(list1)
```

```
C:\Users\91855\AppData\Local\Programs\Python\Python37\python.exe C:/Users/Public/list002.py
['q', 'w', 'e', 'r', 't', 'y']
['q', 'Q', 'e', 'r', 't', 'y']
['q', 'Q', 'e', 'r', 't', 'y', 'u']

Process finished with exit code 0
```

Thank you