

Data structures in python

Topic :List

Exercise

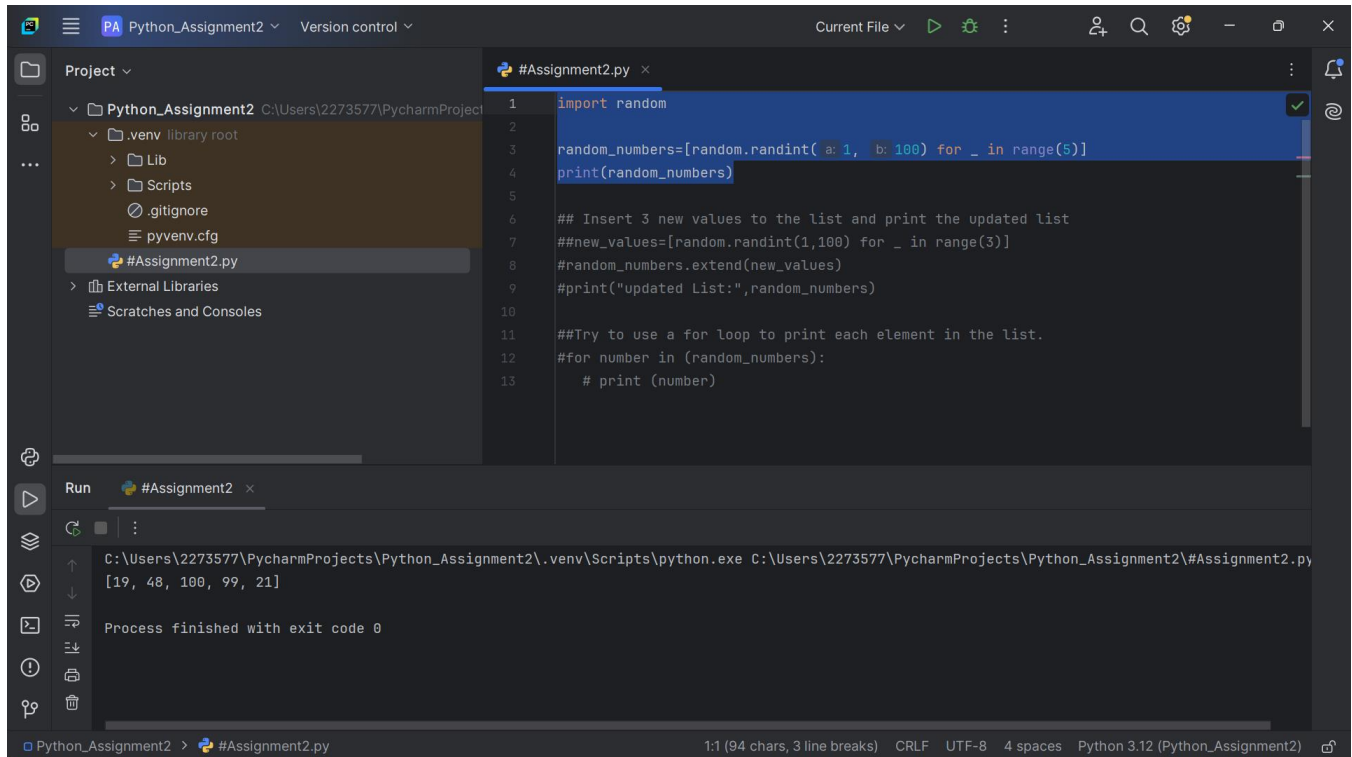
Q1. Create a list of 5 random numbers and print the list.

Solution : The code generates a list of 5 random integers between 1 and 100 using a list comprehension and prints the list.

import random

random_numbers=[random.randint(1, 100) for _ in range(5)]

print(random_numbers)



The screenshot shows the PyCharm IDE interface. The left sidebar displays the project structure for 'Python_Assignment2'. The main editor window shows the file '#Assignment2.py' with the following code:

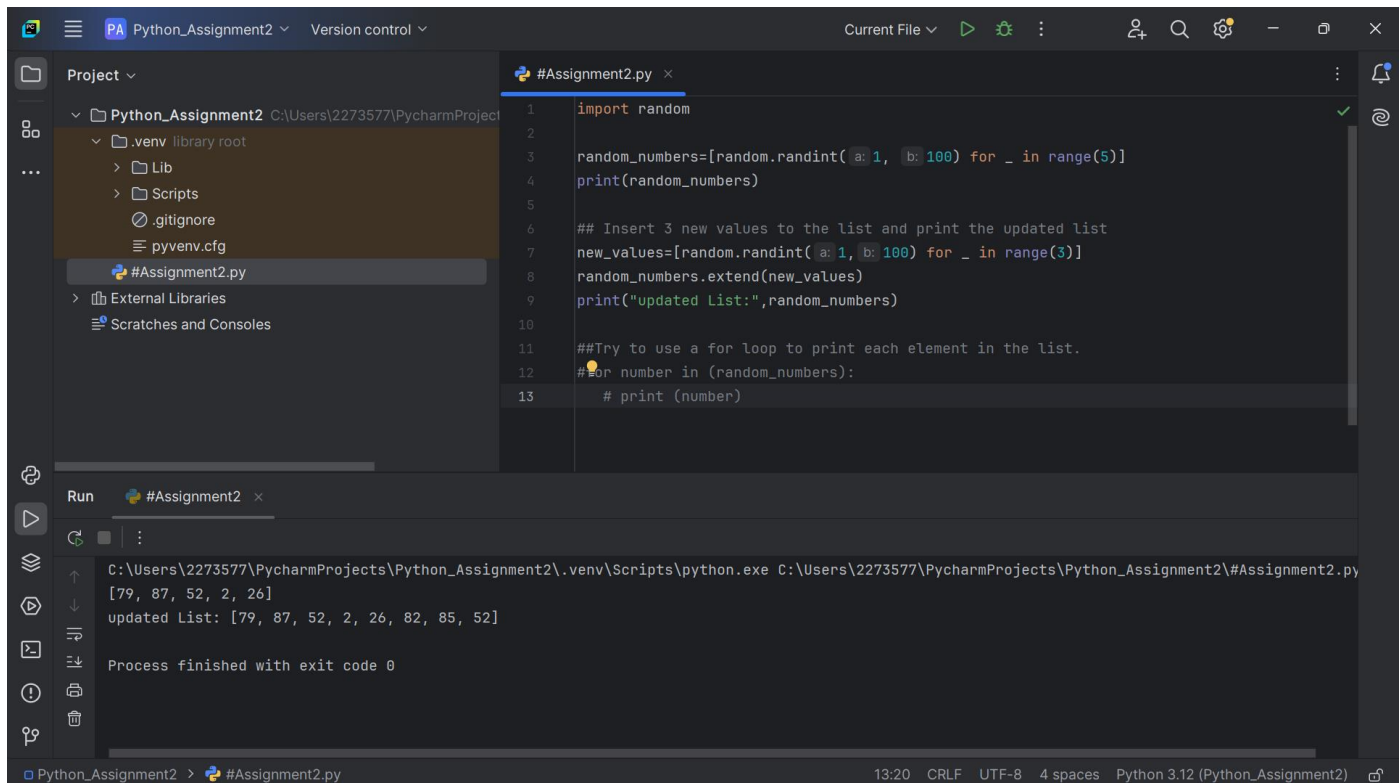
```
1 import random
2
3 random_numbers=[random.randint( a: 1, b: 100) for _ in range(5)]
4 print(random_numbers)
5
6 ## Insert 3 new values to the list and print the updated list
7 ##new_values=[random.randint(1,100) for _ in range(3)]
8 #random_numbers.extend(new_values)
9 #print("updated List:",random_numbers)
10
11 ##Try to use a for loop to print each element in the list.
12 #for number in (random_numbers):
13     # print (number)
```

The bottom panel shows the 'Run' output for '#Assignment2'. The command executed is 'C:\Users\2273577\PycharmProjects\Python_Assignment2\.venv\Scripts\python.exe C:\Users\2273577\PycharmProjects\Python_Assignment2\#Assignment2.py', and the output is '[19, 48, 100, 99, 21]'. Below the output, it states 'Process finished with exit code 0'.

Q2. Insert 3 new values to the list and print the updated list.

Solution : Solution The code extends the original list of 5 random numbers by adding 3 new random values and prints the updated list

```
new_values=[random.randint(1,100) for _ in range(3)]  
random_numbers.extend(new_values)  
print("updated List:",random_numbers)
```



The screenshot displays the PyCharm IDE interface. The top toolbar includes icons for file operations, search, and running code. The left sidebar shows the project structure for 'Python_Assignment2', including a virtual environment (.venv) and various files like .gitignore and pyenv.cfg. The main editor window shows the file '#Assignment2.py' with the following Python code:

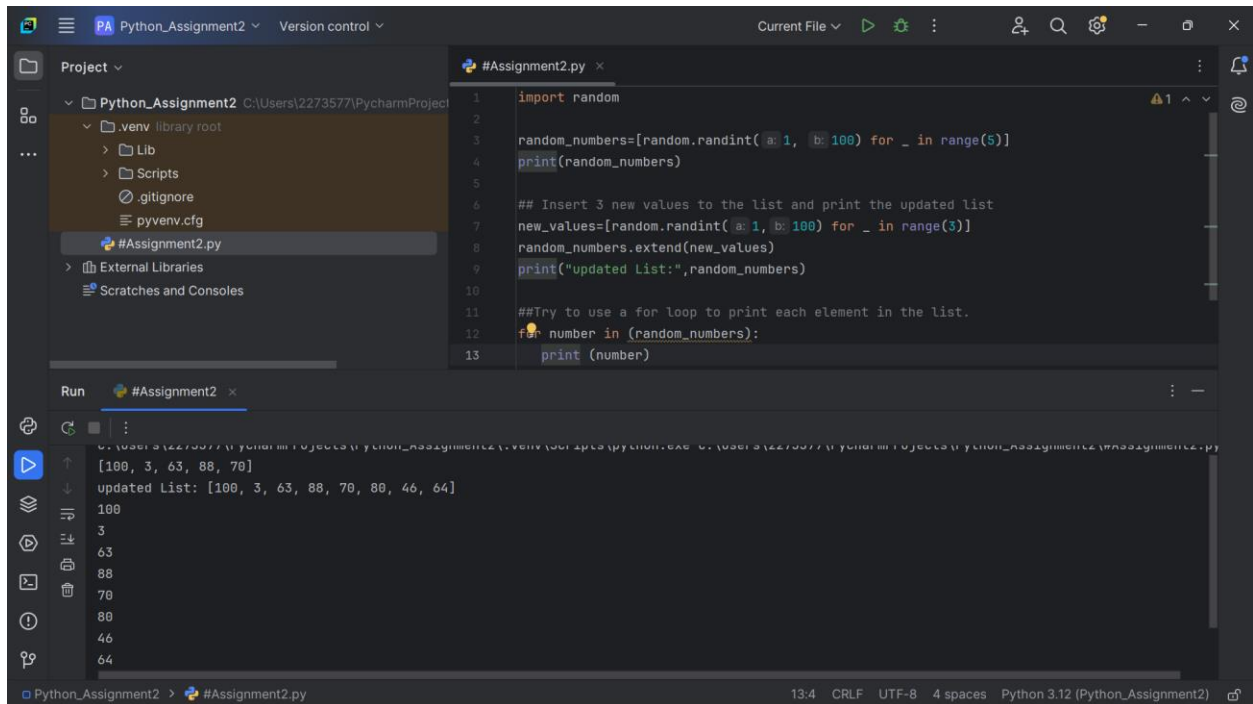
```
1 import random  
2  
3 random_numbers=[random.randint(a: 1, b: 100) for _ in range(5)]  
4 print(random_numbers)  
5  
6 ## Insert 3 new values to the list and print the updated list  
7 new_values=[random.randint(a: 1, b: 100) for _ in range(3)]  
8 random_numbers.extend(new_values)  
9 print("updated List:",random_numbers)  
10  
11 ##Try to use a for loop to print each element in the list.  
12 #for number in (random_numbers):  
13     # print (number)
```

Below the editor, the 'Run' tab is active, showing the execution output for '#Assignment2'. The command executed is 'C:\Users\2273577\PycharmProjects\Python_Assignment2\.venv\Scripts\python.exe C:\Users\2273577\PycharmProjects\Python_Assignment2\#Assignment2.py'. The output shows the initial list '[79, 87, 52, 2, 26]' and the updated list 'updated List: [79, 87, 52, 2, 26, 82, 85, 52]'. The process finished with exit code 0. The bottom status bar indicates the file encoding is UTF-8, 4 spaces, and the Python version is 3.12.

Q3. Try to use a for loop to print each element in the list.

Solution: This code iterates through each element in random_numbers and prints it.

```
for number in (random_numbers):  
    print (number)
```



The screenshot shows the PyCharm IDE interface. The top toolbar includes icons for file operations, running, and debugging. The left sidebar shows the project structure for 'Python_Assignment2', including a virtual environment and various files. The main editor displays the file '#Assignment2.py' with the following Python code:

```
1 import random  
2  
3 random_numbers=[random.randint(a=1, b=100) for _ in range(5)]  
4 print(random_numbers)  
5  
6 ## Insert 3 new values to the list and print the updated list  
7 new_values=[random.randint(a=1, b=100) for _ in range(3)]  
8 random_numbers.extend(new_values)  
9 print("updated List:",random_numbers)  
10  
11 ##Try to use a for loop to print each element in the list.  
12 for number in (random_numbers):  
13     print (number)
```

Below the editor, the 'Run' console shows the output of the script:

```
[100, 3, 63, 88, 70]  
updated List: [100, 3, 63, 88, 70, 80, 46, 64]  
100  
3  
63  
88  
70  
80  
46  
64
```

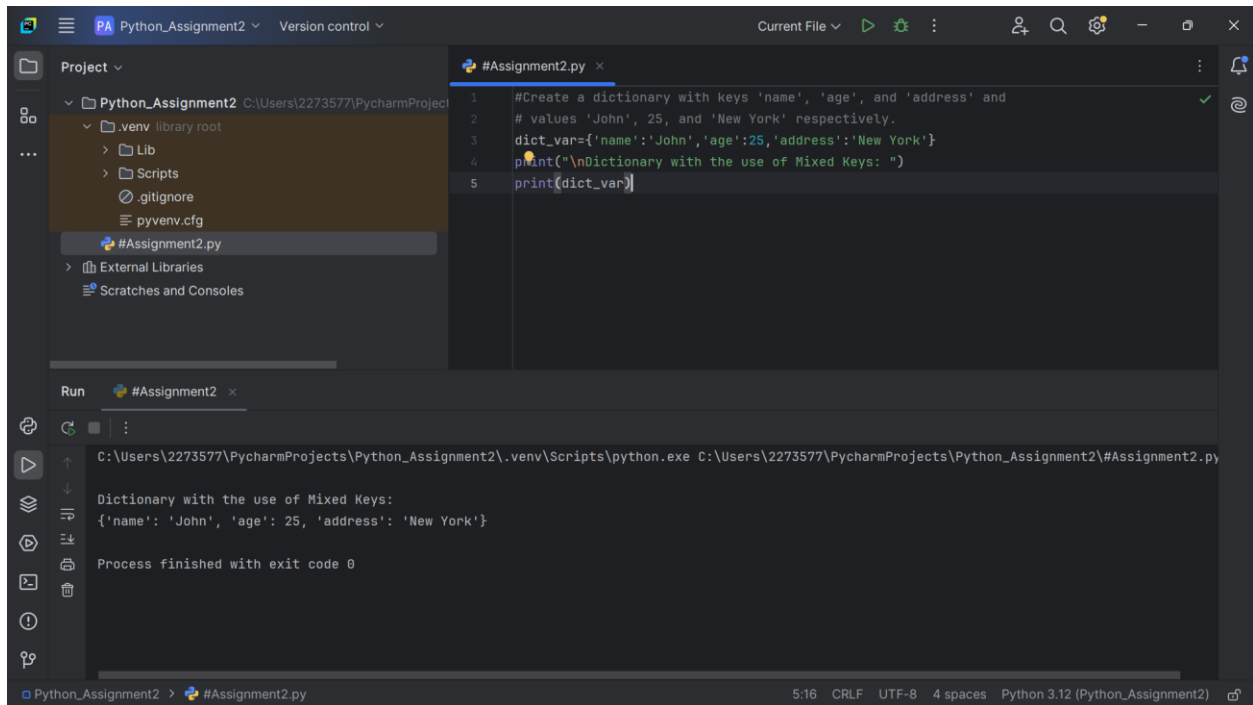
The status bar at the bottom indicates the file encoding is UTF-8, uses 4 spaces for indentation, and is running Python 3.12.

Topic: Dictionary

Exercise

Q1. Create a dictionary with keys 'name', 'age', and 'address' and values 'John', 25, and 'New York' respectively.

Solution: `dict_var={'name':'John','age':25,'address':'New York'}`
`print("\nDictionary with the use of Mixed Keys: ")`
`print(dict_var)`



The screenshot displays the PyCharm IDE interface. The top toolbar includes icons for file operations, running, and debugging. The left sidebar shows the project structure for 'Python_Assignment2', with the file '#Assignment2.py' selected. The main editor window shows the following code:

```
1 #Create a dictionary with keys 'name', 'age', and 'address' and
2 # values 'John', 25, and 'New York' respectively.
3 dict_var={'name':'John','age':25,'address':'New York'}
4 print("\nDictionary with the use of Mixed Keys: ")
5 print(dict_var)
```

Below the editor, the 'Run' console shows the execution output:

```
C:\Users\2273577\PycharmProjects\Python_Assignment2\.venv\Scripts\python.exe C:\Users\2273577\PycharmProjects\Python_Assignment2\#Assignment2.py

Dictionary with the use of Mixed Keys:
{'name': 'John', 'age': 25, 'address': 'New York'}

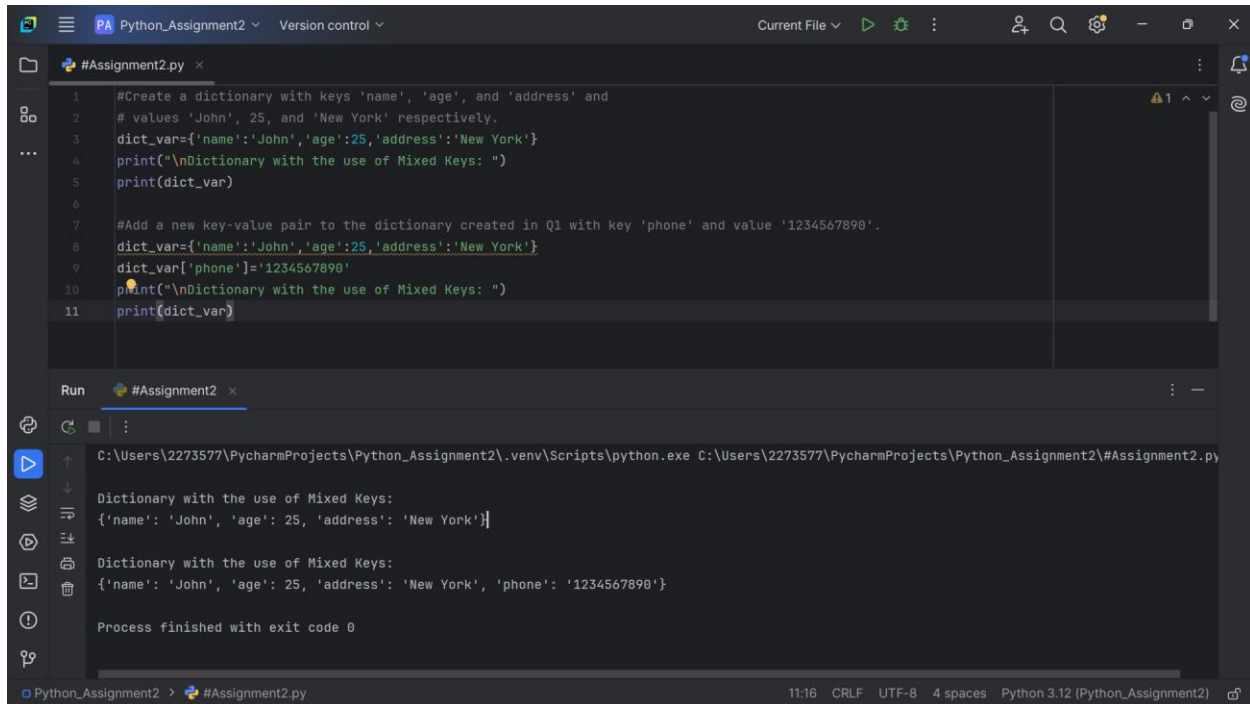
Process finished with exit code 0
```

The status bar at the bottom indicates the file encoding is UTF-8, uses 4 spaces for indentation, and is running Python 3.12.

Q2. Add a new key-value pair to the dictionary created in Q1 with key 'phone' and value '1234567890'.

Solution: This code adds the key 'phone' with the value '1234567890' to the existing dictionary

```
dict_var={'name':'John','age':25,'address':'New York'}
dict_var['phone']='1234567890'
print("\nDictionary with the use of Mixed Keys: ")
print(dict_var)
```



The screenshot displays the PyCharm IDE interface. The top toolbar includes icons for file operations, running, and debugging. The main editor window shows a file named `#Assignment2.py` with the following Python code:

```
1 #Create a dictionary with keys 'name', 'age', and 'address' and
2 # values 'John', 25, and 'New York' respectively.
3 dict_var={'name':'John','age':25,'address':'New York'}
4 print("\nDictionary with the use of Mixed Keys: ")
5 print(dict_var)
6
7 #Add a new key-value pair to the dictionary created in Q1 with key 'phone' and value '1234567890'.
8 dict_var={'name':'John','age':25,'address':'New York'}
9 dict_var['phone']='1234567890'
10 print("\nDictionary with the use of Mixed Keys: ")
11 print(dict_var)
```

Below the editor, the 'Run' console shows the output of the script:

```
Run #Assignment2
C:\Users\2273577\PycharmProjects\Python_Assignment2\.venv\Scripts\python.exe C:\Users\2273577\PycharmProjects\Python_Assignment2\#Assignment2.py

Dictionary with the use of Mixed Keys:
{'name': 'John', 'age': 25, 'address': 'New York'}

Dictionary with the use of Mixed Keys:
{'name': 'John', 'age': 25, 'address': 'New York', 'phone': '1234567890'}

Process finished with exit code 0
```

The status bar at the bottom indicates the file encoding is UTF-8, the line ending is CRLF, and the Python interpreter is Python 3.12.

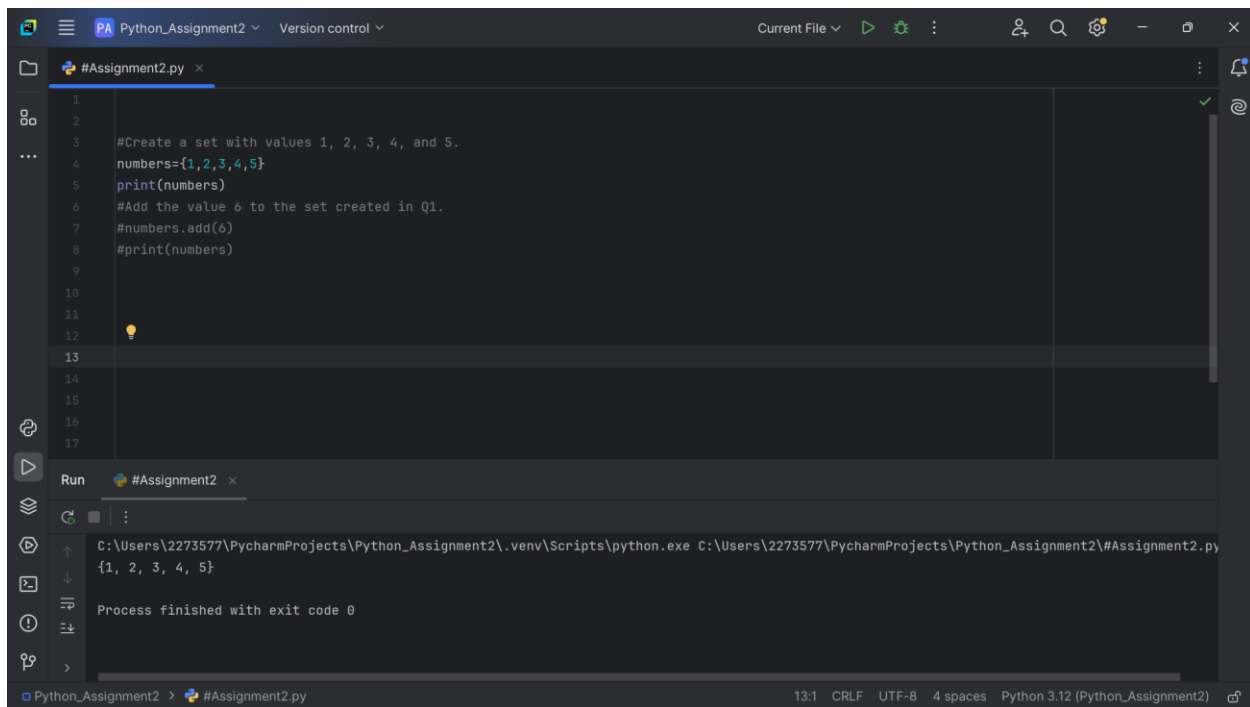
Topic: Set

Exercise

Q1.Create a set with values 1, 2, 3, 4, and 5.

Solution: This code initializes a set containing the specified values.

```
numbers={1,2,3,4,5}  
print(numbers)
```



The screenshot shows the PyCharm IDE interface. The main editor window displays a Python file named `#Assignment2.py` with the following code:

```
1  
2  
3 #Create a set with values 1, 2, 3, 4, and 5.  
4 numbers={1,2,3,4,5}  
5 print(numbers)  
6 #Add the value 6 to the set created in Q1.  
7 #numbers.add(6)  
8 #print(numbers)  
9  
10  
11  
12  
13  
14  
15  
16  
17
```

The Run tool window at the bottom shows the execution output:

```
Run #Assignment2  
C:\Users\2273577\PycharmProjects\Python_Assignment2\.venv\Scripts\python.exe C:\Users\2273577\PycharmProjects\Python_Assignment2\#Assignment2.py  
{1, 2, 3, 4, 5}  
Process finished with exit code 0
```

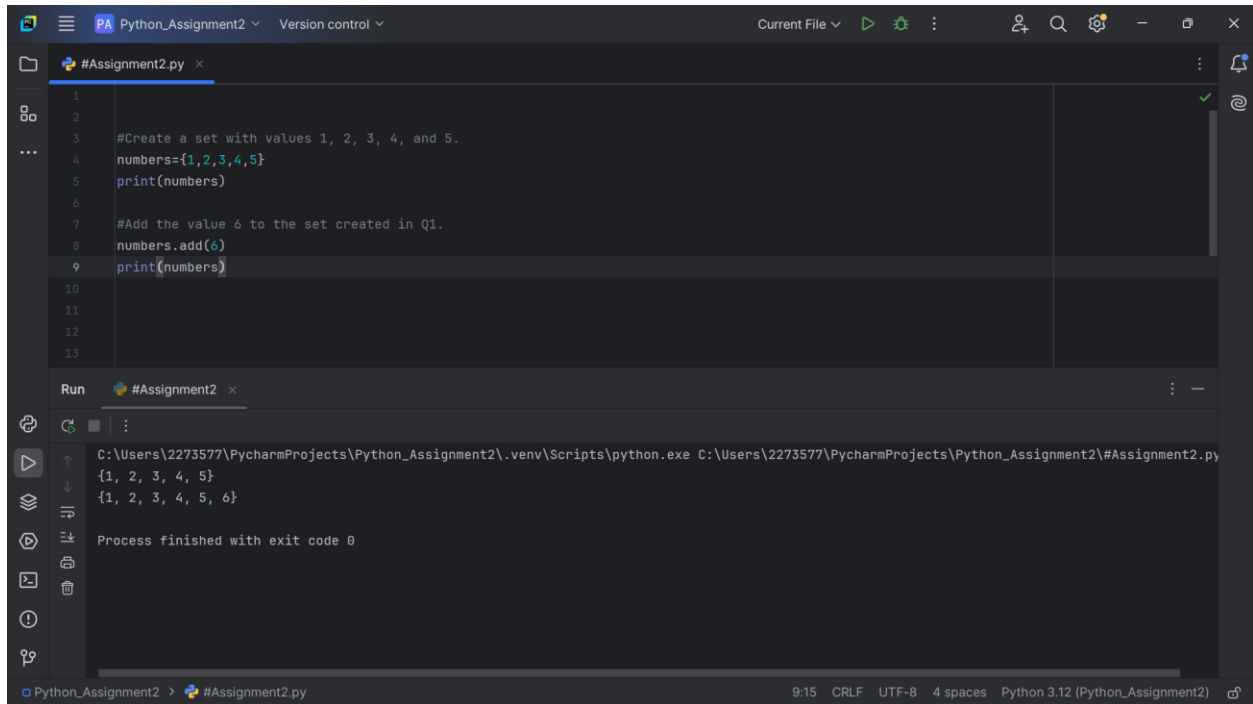
The status bar at the bottom indicates the file encoding is UTF-8, the line ending is CRLF, and the indentation is 4 spaces. The Python version is 3.12.

Q2. Add the value 6 to the set created in Q1.

Solution: This code adds the value 6 to the existing set.

numbers.add(6)

print(numbers)



The screenshot shows the PyCharm IDE interface. The top toolbar includes icons for file operations, running, and debugging. The main editor window displays a Python file named `#Assignment2.py` with the following code:

```
1
2
3 #Create a set with values 1, 2, 3, 4, and 5.
4 numbers={1,2,3,4,5}
5 print(numbers)
6
7 #Add the value 6 to the set created in Q1.
8 numbers.add(6)
9 print(numbers)
10
11
12
13
```

Below the editor is the 'Run' console, which shows the output of the script:

```
Run #Assignment2
C:\Users\2273577\PycharmProjects\Python_Assignment2\.venv\Scripts\python.exe C:\Users\2273577\PycharmProjects\Python_Assignment2\#Assignment2.py
{1, 2, 3, 4, 5}
{1, 2, 3, 4, 5, 6}
Process finished with exit code 0
```

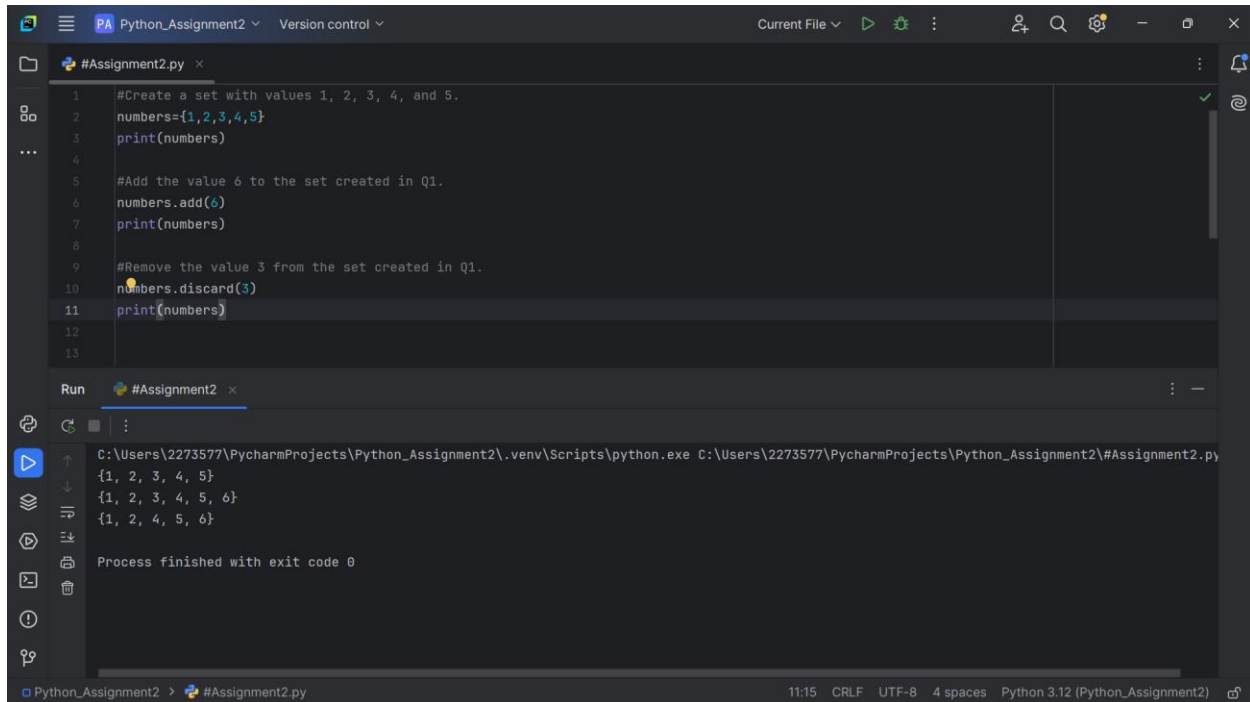
The status bar at the bottom indicates the file encoding is UTF-8, the line ending is CRLF, and the Python version is 3.12.

Q3. Remove the value 3 from the set created in Q1.

Solution: This code removes the value 3 from the existing set

```
numbers.discard(3)
```

```
print(numbers)
```



The screenshot shows the PyCharm IDE interface. The top toolbar includes icons for file operations, running, and debugging. The main editor window displays a Python file named `#Assignment2.py` with the following code:

```
1 #Create a set with values 1, 2, 3, 4, and 5.  
2 numbers={1,2,3,4,5}  
3 print(numbers)  
4  
5 #Add the value 6 to the set created in Q1.  
6 numbers.add(6)  
7 print(numbers)  
8  
9 #Remove the value 3 from the set created in Q1.  
10 numbers.discard(3)  
11 print(numbers)  
12  
13
```

Below the editor, the 'Run' console shows the execution output for `#Assignment2`:

```
C:\Users\2273577\PycharmProjects\Python_Assignment2\.venv\Scripts\python.exe C:\Users\2273577\PycharmProjects\Python_Assignment2\#Assignment2.py  
{1, 2, 3, 4, 5}  
{1, 2, 3, 4, 5, 6}  
{1, 2, 4, 5, 6}  
  
Process finished with exit code 0
```

The status bar at the bottom indicates the current file is `#Assignment2.py`, the encoding is UTF-8, and the Python version is 3.12.

Topic: Tuple

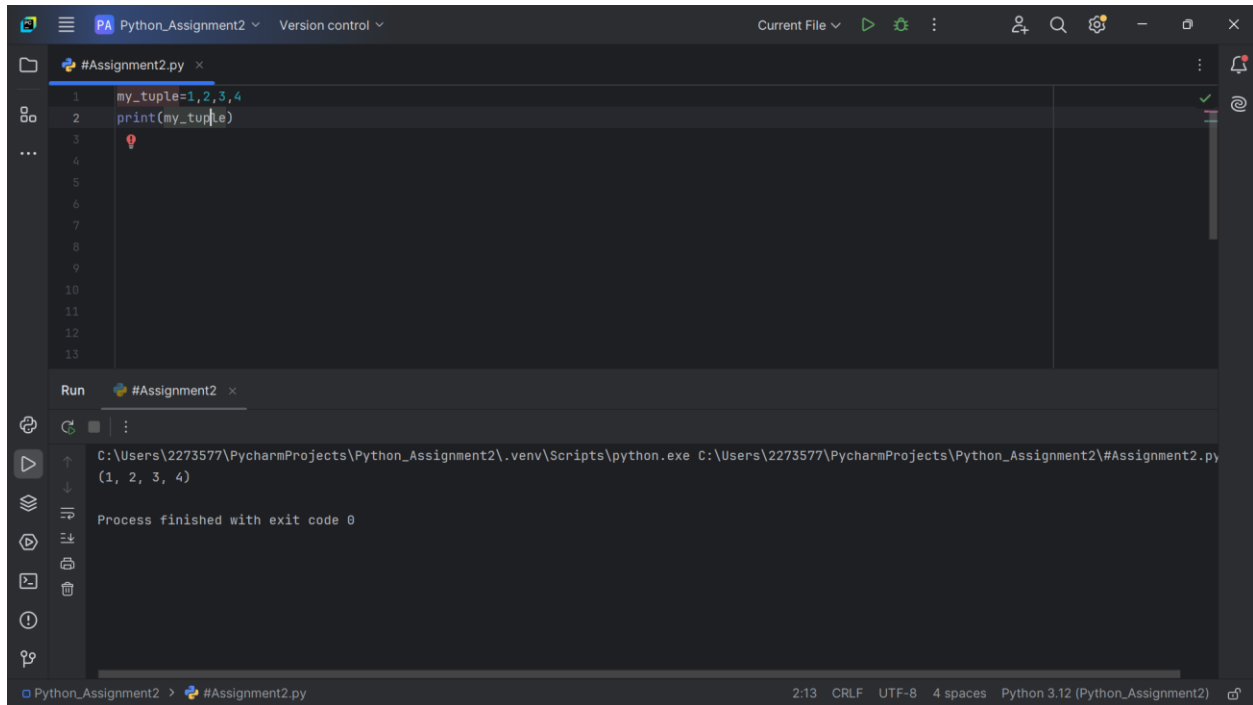
Exercise

Q1. Create a tuple with values 1, 2, 3, and 4

Solution: This code initializes a tuple containing the specified values.

```
my_tuple=1,2,3,4
```

```
print(my_tuple)
```



The screenshot displays the PyCharm IDE interface. The main editor window shows a file named `#Assignment2.py` with the following code:

```
1 my_tuple=1,2,3,4
2 print(my_tuple)
```

The code is executed, and the output is shown in the Run console at the bottom. The console output is:

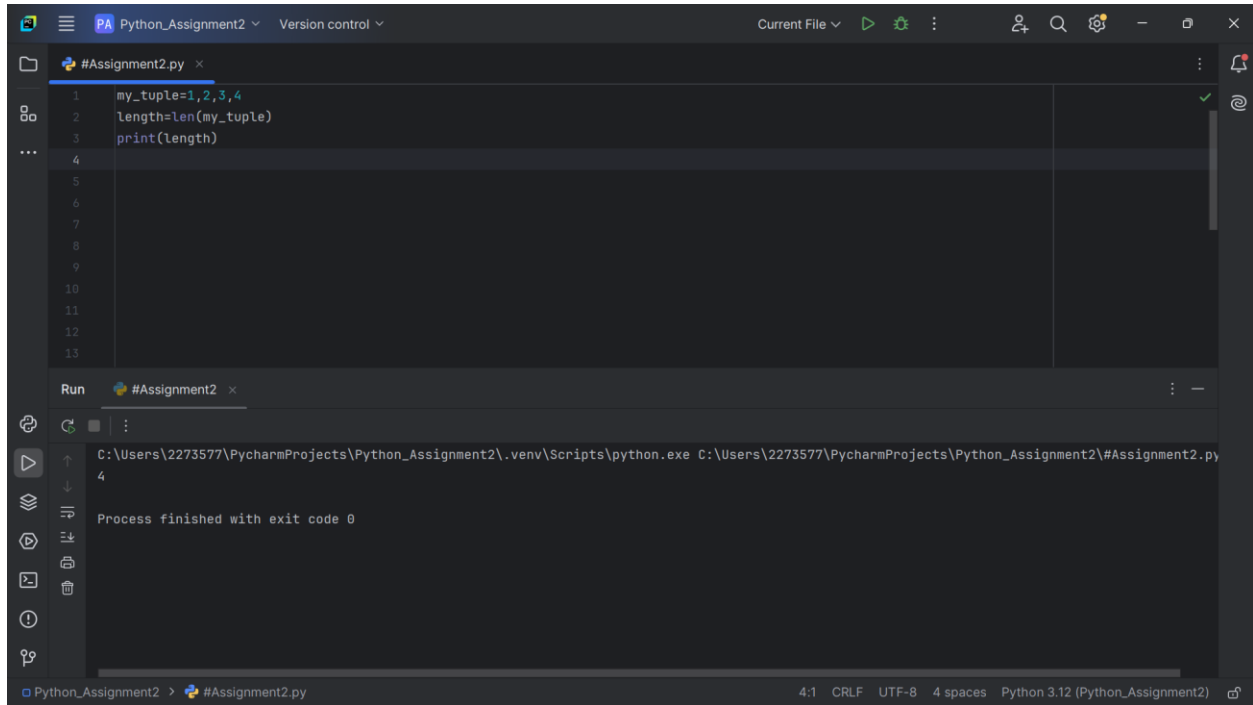
```
C:\Users\2273577\PycharmProjects\Python_Assignment2\.venv\Scripts\python.exe C:\Users\2273577\PycharmProjects\Python_Assignment2\#Assignment2.py
(1, 2, 3, 4)
Process finished with exit code 0
```

The status bar at the bottom indicates the file encoding is UTF-8, the line ending is CRLF, and the Python version is 3.12.

Q2. Print the length of the tuple created in Q1.

Solution: This code prints the length of the tuple `my_tuple`, which is 4.

```
my_tuple=1,2,3,4  
length=len(my_tuple)  
print(length)
```



The screenshot shows the PyCharm IDE interface. The top toolbar includes icons for file operations, search, and running code. The main editor window displays a file named `#Assignment2.py` with the following code:

```
1 my_tuple=1,2,3,4  
2 length=len(my_tuple)  
3 print(length)  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13
```

Below the editor is the 'Run' console. It shows the command executed: `C:\Users\2273577\PycharmProjects\Python_Assignment2\.venv\Scripts\python.exe C:\Users\2273577\PycharmProjects\Python_Assignment2\#Assignment2.py`. The output is `4`, and the status message is 'Process finished with exit code 0'. The bottom status bar indicates the file encoding is UTF-8, uses 4 spaces for indentation, and is running Python 3.12.