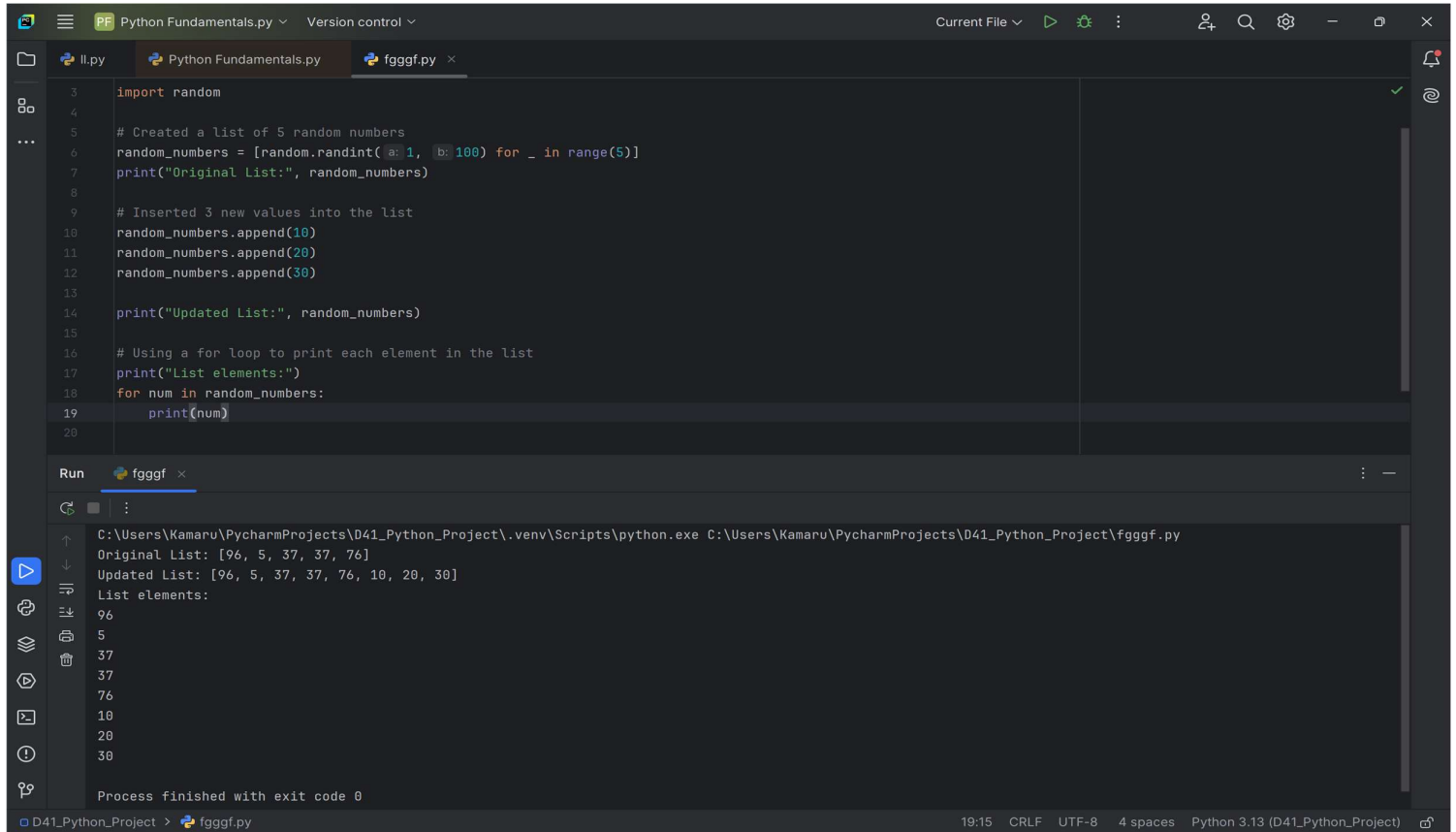


# 1. LIST



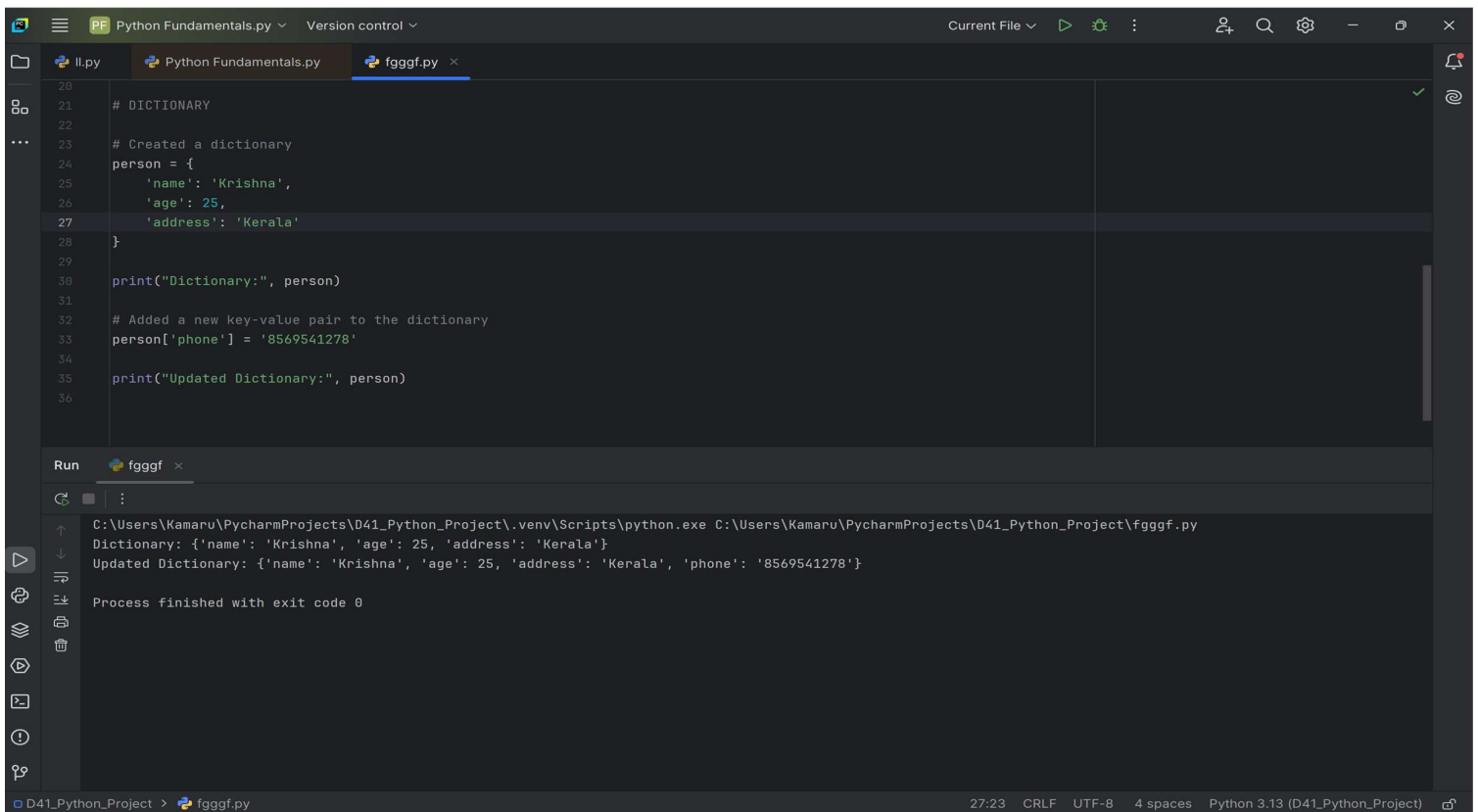
The screenshot shows a PyCharm IDE with a Python file named `fgggf.py`. The code defines a list of 5 random numbers, appends three more values (10, 20, 30), and prints the updated list. It also uses a for loop to print each element of the list.

```
3 import random
4
5 # Created a list of 5 random numbers
6 random_numbers = [random.randint(a=1, b=100) for _ in range(5)]
7 print("Original List:", random_numbers)
8
9 # Inserted 3 new values into the list
10 random_numbers.append(10)
11 random_numbers.append(20)
12 random_numbers.append(30)
13
14 print("Updated List:", random_numbers)
15
16 # Using a for loop to print each element in the list
17 print("List elements:")
18 for num in random_numbers:
19     print(num)
20
```

The Run window shows the following output:

```
C:\Users\Kamaru\PycharmProjects\D41_Python_Project\.venv\Scripts\python.exe C:\Users\Kamaru\PycharmProjects\D41_Python_Project\fgggf.py
Original List: [96, 5, 37, 37, 76]
Updated List: [96, 5, 37, 37, 76, 10, 20, 30]
List elements:
96
5
37
37
76
10
20
30
Process finished with exit code 0
```

# 2. DICTIONARY



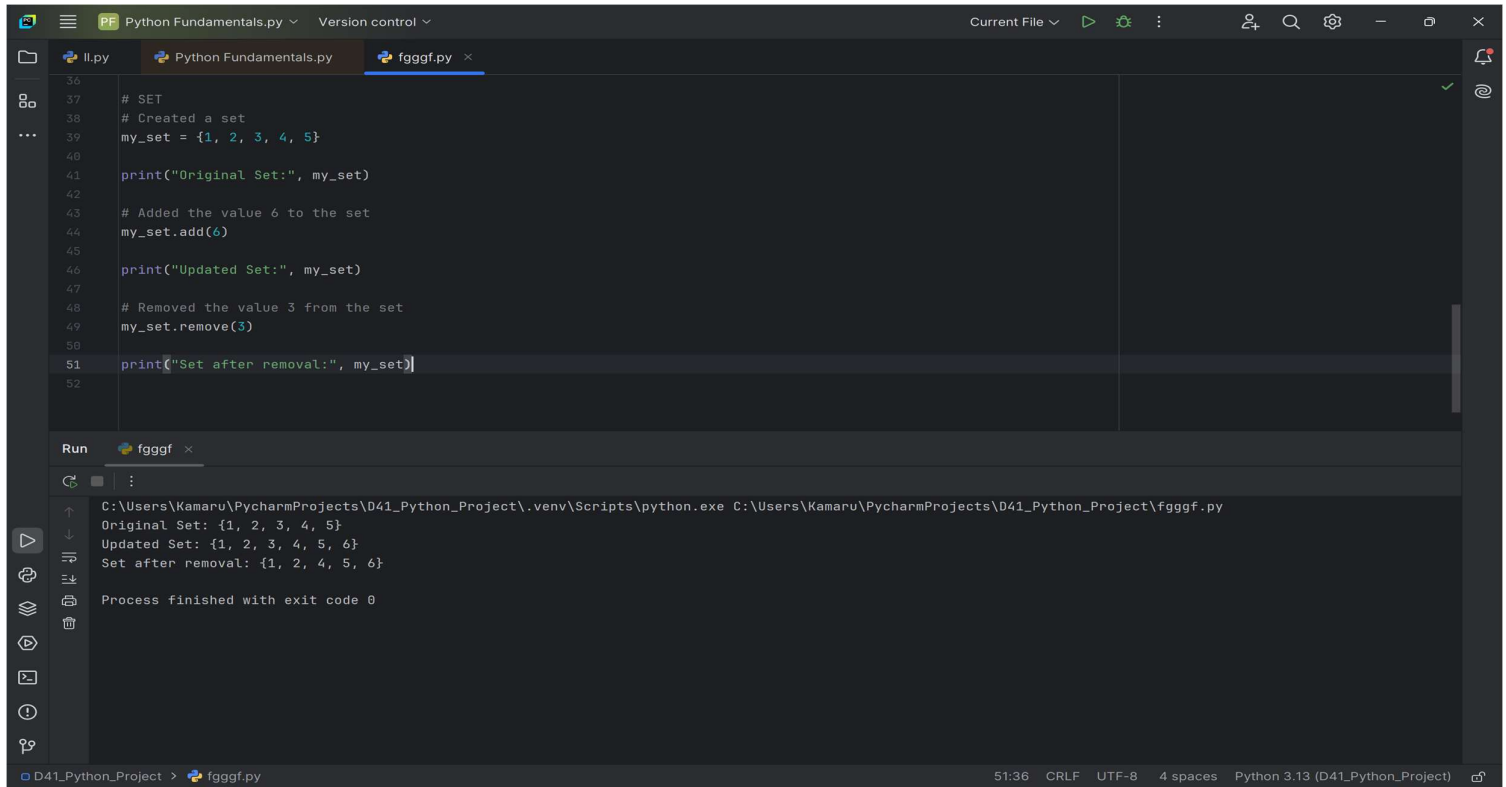
The screenshot shows a PyCharm IDE with a Python file named `fgggf.py`. The code creates a dictionary for a person, adds a new key-value pair ('phone'), and prints the updated dictionary.

```
20
21 # DICTIONARY
22
23 # Created a dictionary
24 person = {
25     'name': 'Krishna',
26     'age': 25,
27     'address': 'Kerala'
28 }
29
30 print("Dictionary:", person)
31
32 # Added a new key-value pair to the dictionary
33 person['phone'] = '8569541278'
34
35 print("Updated Dictionary:", person)
36
```

The Run window shows the following output:

```
C:\Users\Kamaru\PycharmProjects\D41_Python_Project\.venv\Scripts\python.exe C:\Users\Kamaru\PycharmProjects\D41_Python_Project\fgggf.py
Dictionary: {'name': 'Krishna', 'age': 25, 'address': 'Kerala'}
Updated Dictionary: {'name': 'Krishna', 'age': 25, 'address': 'Kerala', 'phone': '8569541278'}
Process finished with exit code 0
```

## 3.SET



The screenshot shows the PyCharm IDE with a Python file named `fgggf.py`. The code defines a set `my_set` with values {1, 2, 3, 4, 5}, prints it, adds the value 6, prints the updated set, removes the value 3, and prints the set after removal. The Run window shows the output of the code execution.

```
36
37 # SET
38 # Created a set
39 my_set = {1, 2, 3, 4, 5}
40
41 print("Original Set:", my_set)
42
43 # Added the value 6 to the set
44 my_set.add(6)
45
46 print("Updated Set:", my_set)
47
48 # Removed the value 3 from the set
49 my_set.remove(3)
50
51 print("Set after removal:", my_set)
52
```

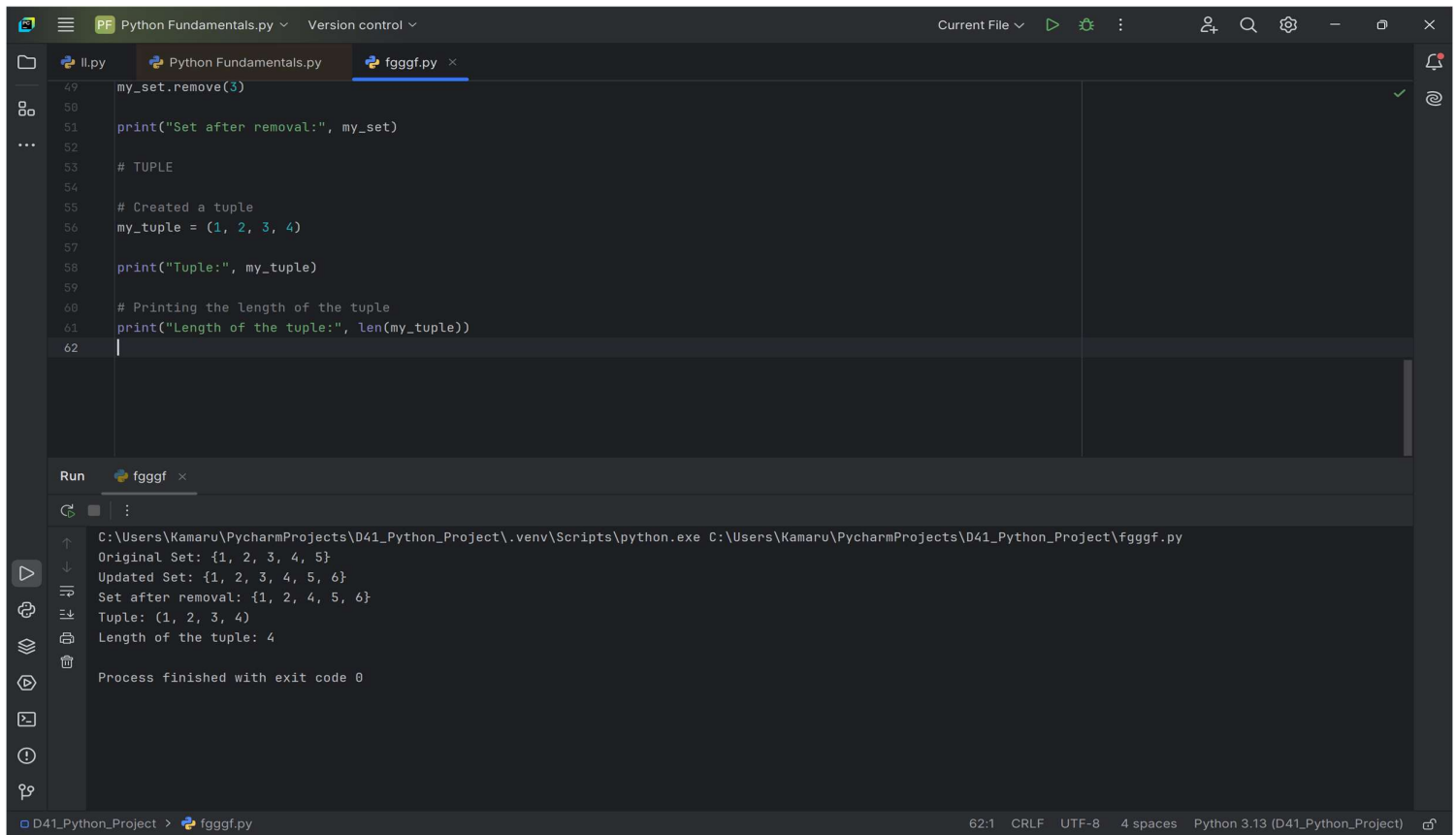
Run fgggf x

C:\Users\Kamaru\PycharmProjects\D41\_Python\_Project\.venv\Scripts\python.exe C:\Users\Kamaru\PycharmProjects\D41\_Python\_Project\fgggf.py

Original Set: {1, 2, 3, 4, 5}  
Updated Set: {1, 2, 3, 4, 5, 6}  
Set after removal: {1, 2, 4, 5, 6}

Process finished with exit code 0

## 4.TUPLE



The screenshot shows the PyCharm IDE with a Python file named `fgggf.py`. The code removes the value 3 from the set, prints the updated set, creates a tuple `my_tuple` with values (1, 2, 3, 4), prints it, and prints its length. The Run window shows the output of the code execution.

```
49 my_set.remove(3)
50
51 print("Set after removal:", my_set)
52
53 # TUPLE
54
55 # Created a tuple
56 my_tuple = (1, 2, 3, 4)
57
58 print("Tuple:", my_tuple)
59
60 # Printing the length of the tuple
61 print("Length of the tuple:", len(my_tuple))
62
```

Run fgggf x

C:\Users\Kamaru\PycharmProjects\D41\_Python\_Project\.venv\Scripts\python.exe C:\Users\Kamaru\PycharmProjects\D41\_Python\_Project\fgggf.py

Original Set: {1, 2, 3, 4, 5}  
Updated Set: {1, 2, 3, 4, 5, 6}  
Set after removal: {1, 2, 4, 5, 6}  
Tuple: (1, 2, 3, 4)  
Length of the tuple: 4

Process finished with exit code 0