

Name: Roshaan Ahmed Khan

ID: BAI-24S-002

Course: Programming Fundamentals

Department: Artificial Intelligence &
Mathematical Sciences

PROGRAMMING LANGUAGE

PYTHON

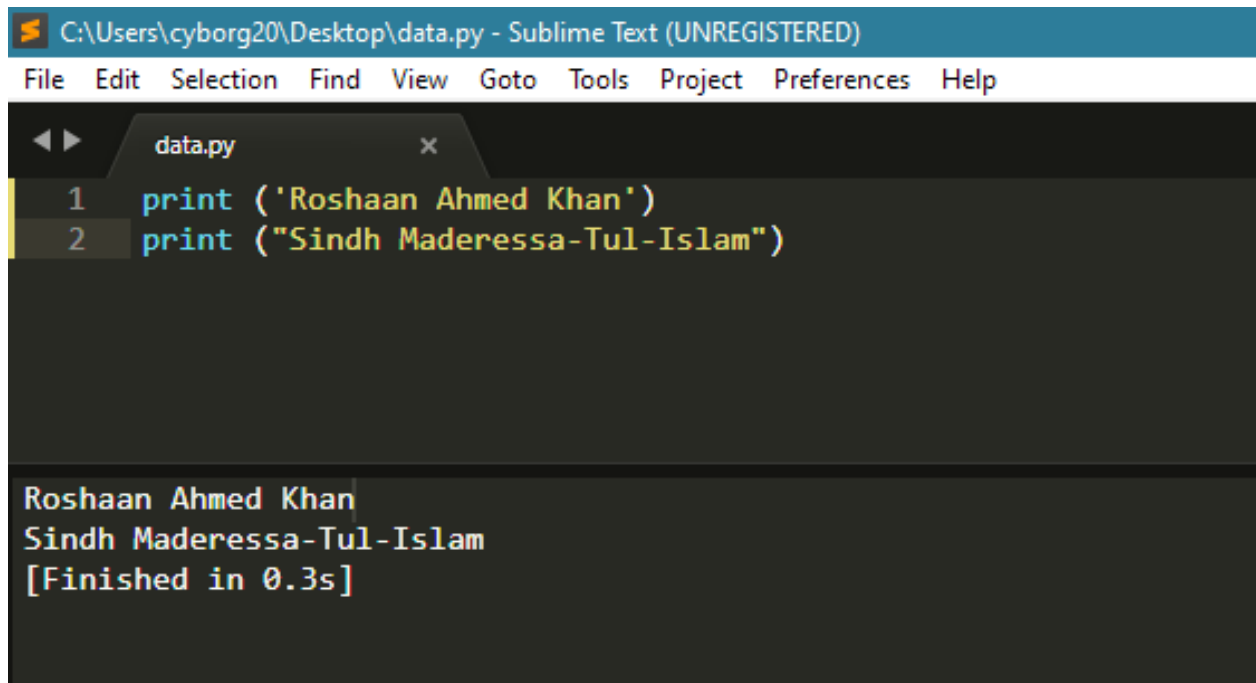
LABORATORY MANUAL



PYTHON LAB WORK 2 | PRINT, VARIABLES

Related to Print Statement:

1. Write your name and university name in print () function with single quote one time and double quote second time in two lines.



The screenshot shows a Sublime Text editor window titled "C:\Users\cyborg20\Desktop\data.py - Sublime Text (UNREGISTERED)". The menu bar includes File, Edit, Selection, Find, View, Goto, Tools, Project, Preferences, and Help. The editor has a tab for "data.py". The code in the file is:

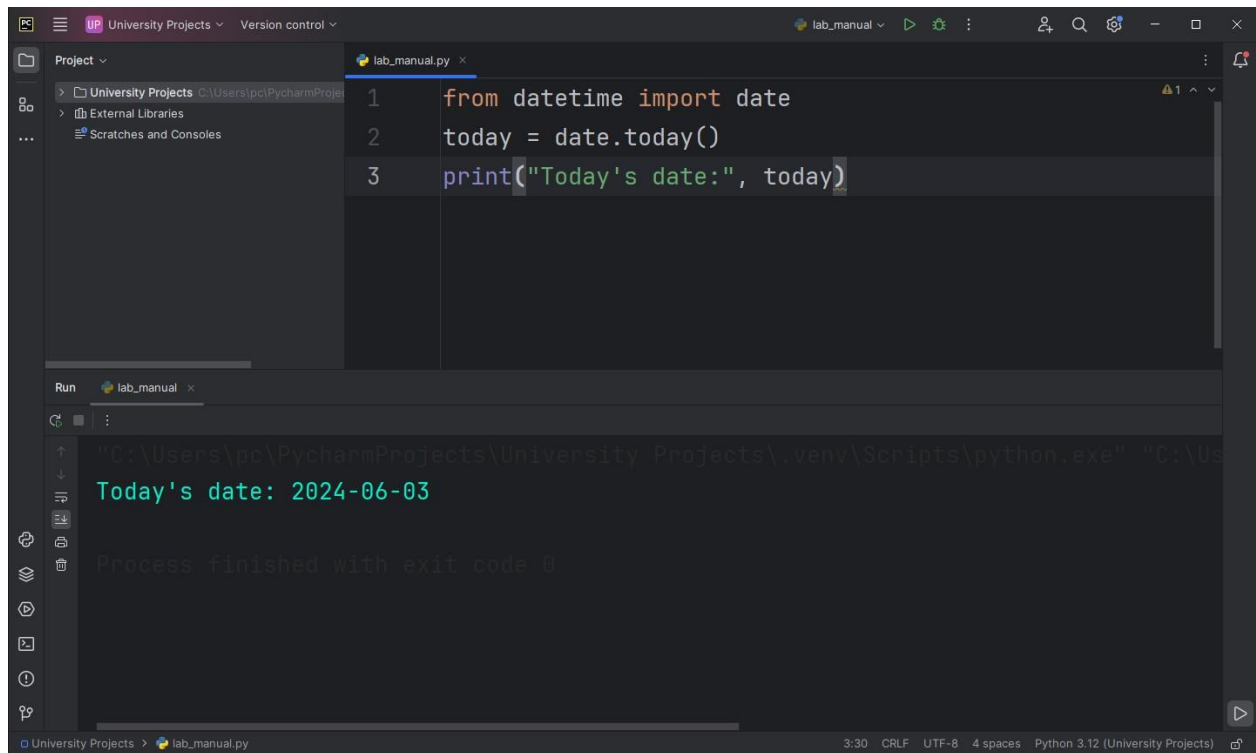
```
1 print ('Roshaan Ahmed Khan')
2 print ("Sindh Maderessa-Tul-Islam")
```

Below the code, the output of the script is displayed:

```
Roshaan Ahmed Khan
Sindh Maderessa-Tul-Islam
[Finished in 0.3s]
```

Related to Print Statement:

2. Execute below code for data and time, paste your screenshot
from datetime import date today = date.today()
print("Today's date:", today)



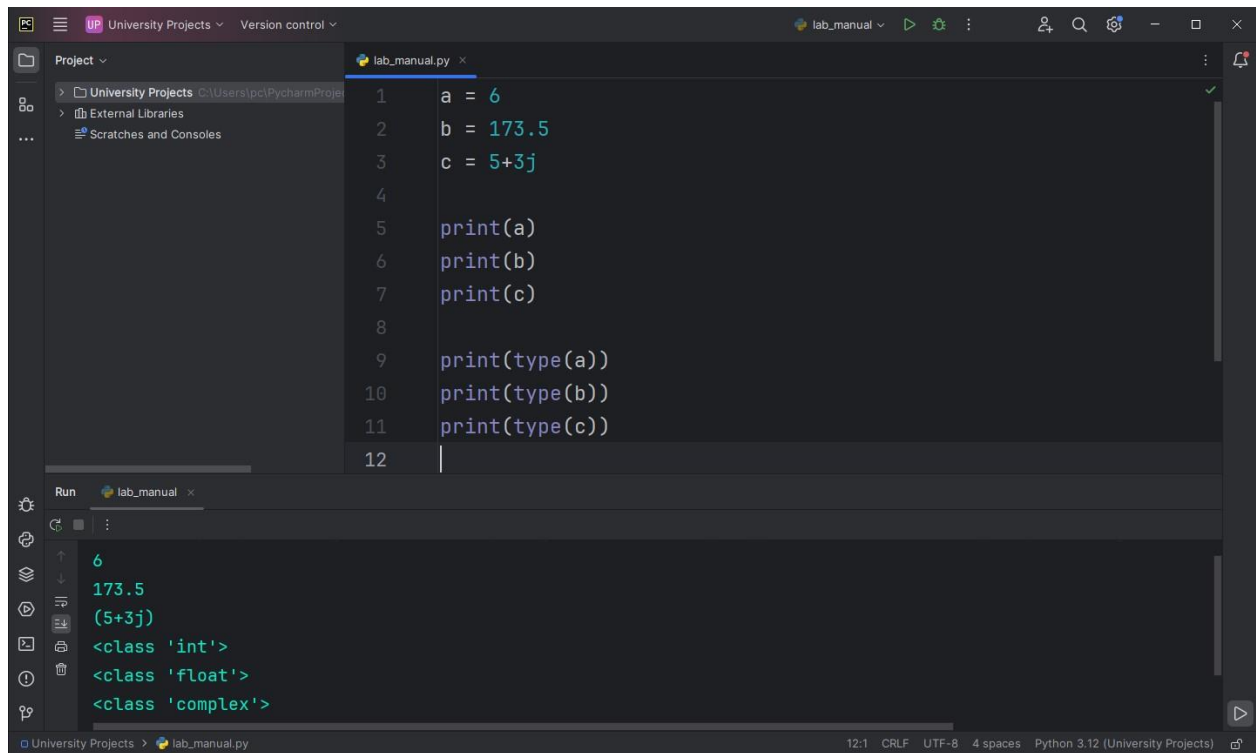
The screenshot shows the PyCharm IDE interface. The top toolbar includes icons for file operations, a search icon, and a play button. The left sidebar shows the 'Project' view with a tree structure containing 'University Projects', 'External Libraries', and 'Scratches and Consoles'. The main editor window displays a file named 'lab_manual.py' with the following code:

```
1 from datetime import date
2 today = date.today()
3 print("Today's date:", today)
```

Below the editor is the 'Run' console. It shows the command prompt path: `"C:\Users\pc\PycharmProjects\University Projects\.venv\Scripts\python.exe" "C:\Users\pc\PycharmProjects\University Projects\lab_manual.py"`. The output of the script is `Today's date: 2024-06-03`. Below the output, it states `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.

Related to Type of Numerical Values:

1. You need to take 3 different variables a, b and c. Assign them numerical values of integer, float and complex. Print all value with their type by using `type ()` function.



The screenshot shows the PyCharm IDE interface. The top toolbar includes icons for file operations, a search icon, and a play button. The left sidebar shows the 'Project' view with 'University Projects' and 'External Libraries'. The main editor window displays a file named 'lab_manual.py' with the following code:

```
1 a = 6
2 b = 173.5
3 c = 5+3j
4
5 print(a)
6 print(b)
7 print(c)
8
9 print(type(a))
10 print(type(b))
11 print(type(c))
12
```

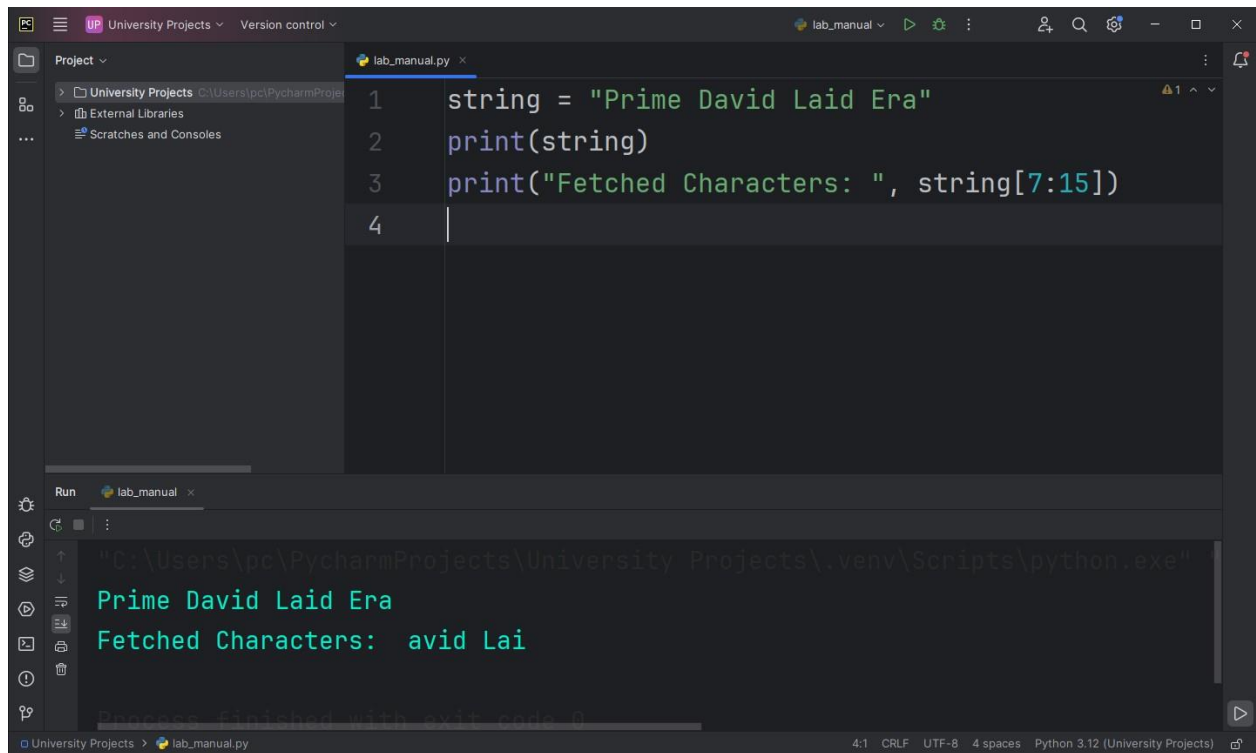
Below the editor is the 'Run' console, which shows the output of the script:

```
6
173.5
(5+3j)
<class 'int'>
<class 'float'>
<class 'complex'>
```

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

Related to String Data Type:

1. Write a long string of words about 20 characters and fetch characters starting from 8th to 15th position.



The screenshot shows the PyCharm IDE interface. The top toolbar includes icons for file operations, a search icon, and a play button. The left sidebar shows the 'Project' view with 'University Projects' and 'External Libraries'. The main editor window displays a file named 'lab_manual.py' with the following code:

```
1 string = "Prime David Laid Era"
2 print(string)
3 print("Fetched Characters: ", string[7:15])
4
```

The bottom panel shows the 'Run' output for 'lab_manual'. It displays the command path: "C:\Users\pc\PycharmProjects\University Projects\.venv\Scripts\python.exe". The output is:

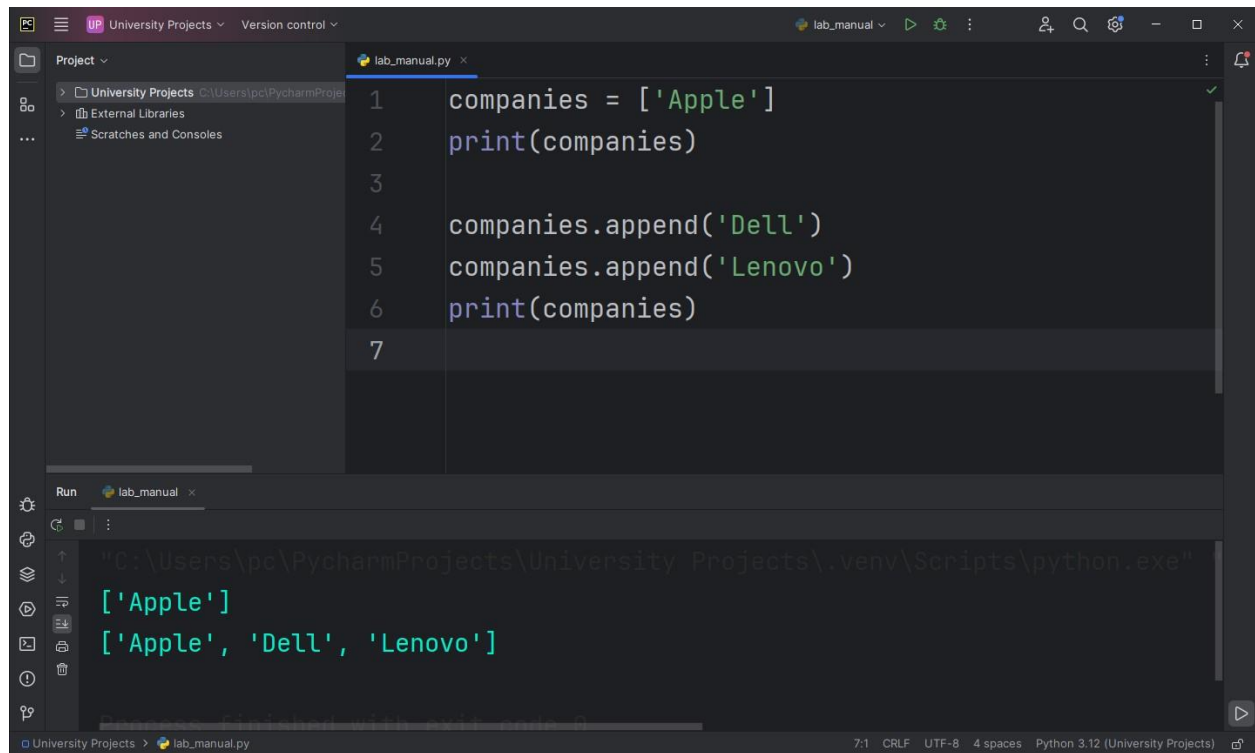
```
Prime David Laid Era
Fetched Characters:  avid Lai
```

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

PYTHON LAB WORK 3 | LISTS

Related to Insert in List:

1. Write a program to insert items in list which are the names of different laptop manufacturing companies.



The screenshot shows the PyCharm IDE interface. The top toolbar includes icons for file operations, a search icon, and a play button. The left sidebar shows the 'Project' view with 'University Projects' and 'lab_manual.py' selected. The main editor window displays the following Python code:

```
1 companies = ['Apple']
2 print(companies)
3
4 companies.append('Dell')
5 companies.append('Lenovo')
6 print(companies)
7
```

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

```
"C:\Users\pc\PycharmProjects\University Projects\.venv\Scripts\python.exe"
['Apple']
['Apple', 'Dell', 'Lenovo']
##### 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.

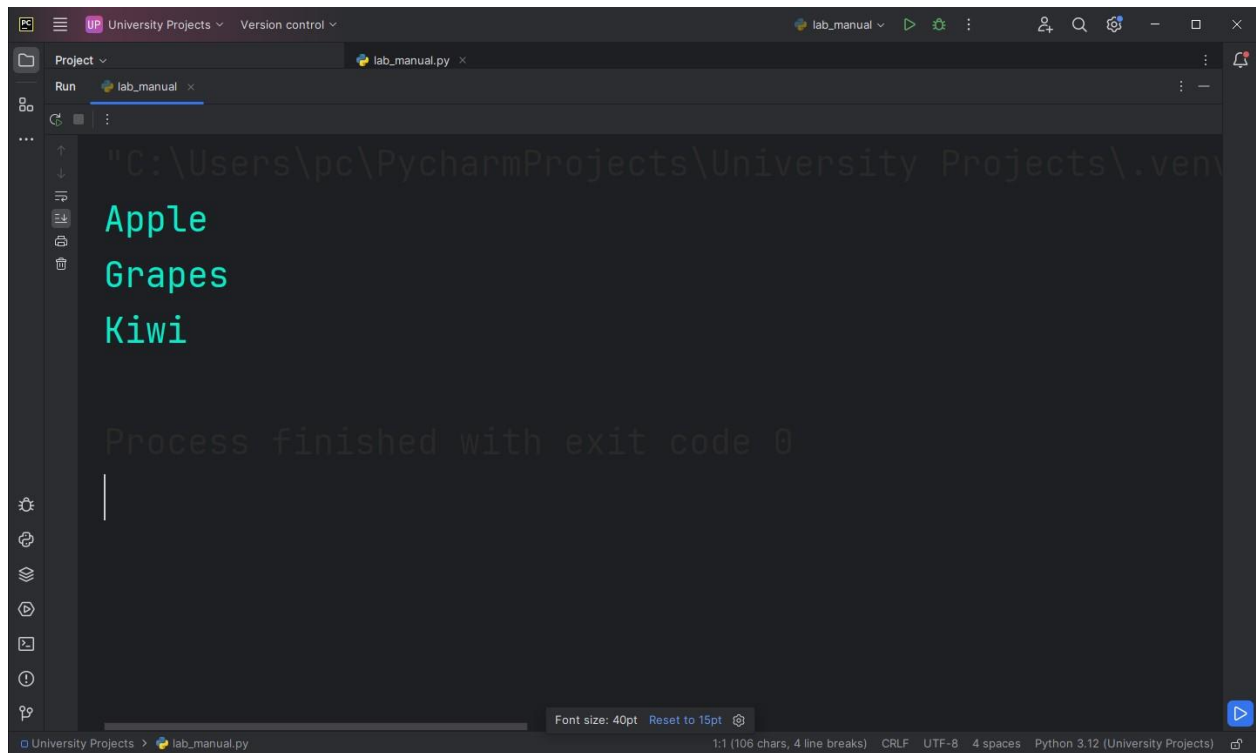
Related to Get Values from List from Given Position:

1. Write a list of fruits and get values of list from any position.

```
fruits = ['Apple', 'Mango', 'Dates', 'Grapes', 'Kiwi']

print(fruits[0])print(fruits[3])
print(fruits[4])
```

Output:



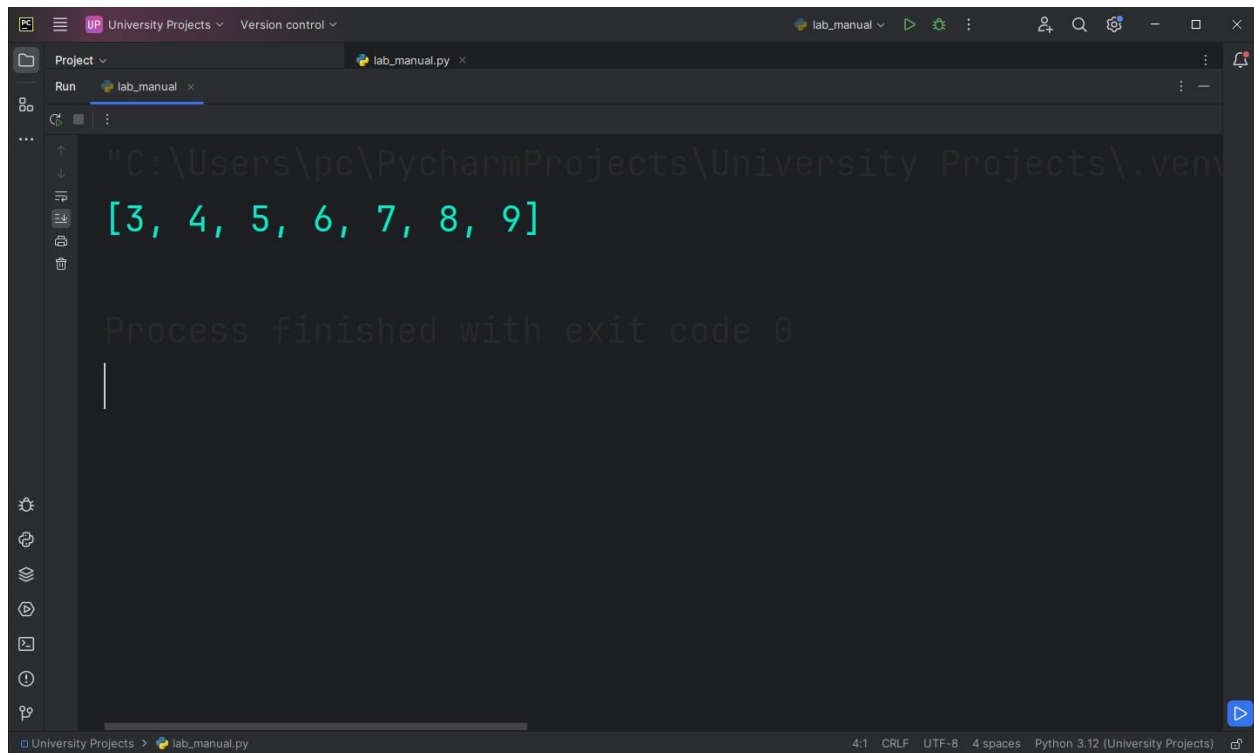
```
"C:\Users\pc\PycharmProjects\University Projects\.venv\n\nApple\nGrapes\nKiwi\n\nProcess finished with exit code 0
```

Related to Get Values from List from Specific Position:

1. Create a list of numbers and get values starting from 3rd position.

```
numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9]\n\nprint(numbers[2:])
```

Output:

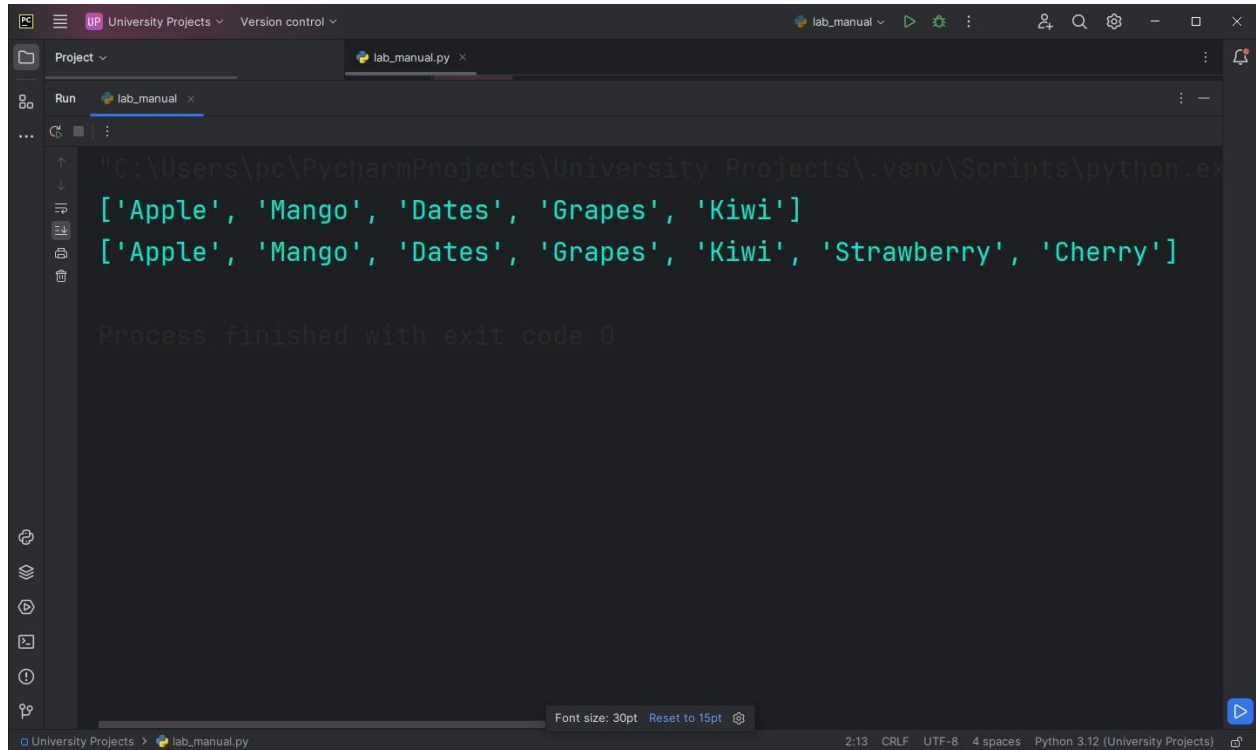


Related to Get Values from List from Specific Position:

1. Update the fruit list you made in the last exercise and add any two new fruits in it.

```
fruits = ['Apple', 'Mango', 'Dates', 'Grapes',  
'Kiwi']  
print(fruits)  
  
fruits.append('Strawberry')  
fruits.append('Cherry')  
print(fruits)
```


Output:



```
"C:\Users\pc\PycharmProjects\University Projects\.venv\Scripts\python.exe  
['Apple', 'Mango', 'Dates', 'Grapes', 'Kiwi']  
['Apple', 'Mango', 'Dates', 'Grapes', 'Kiwi', 'Strawberry', 'Cherry']  
  
Process finished with exit code 0
```

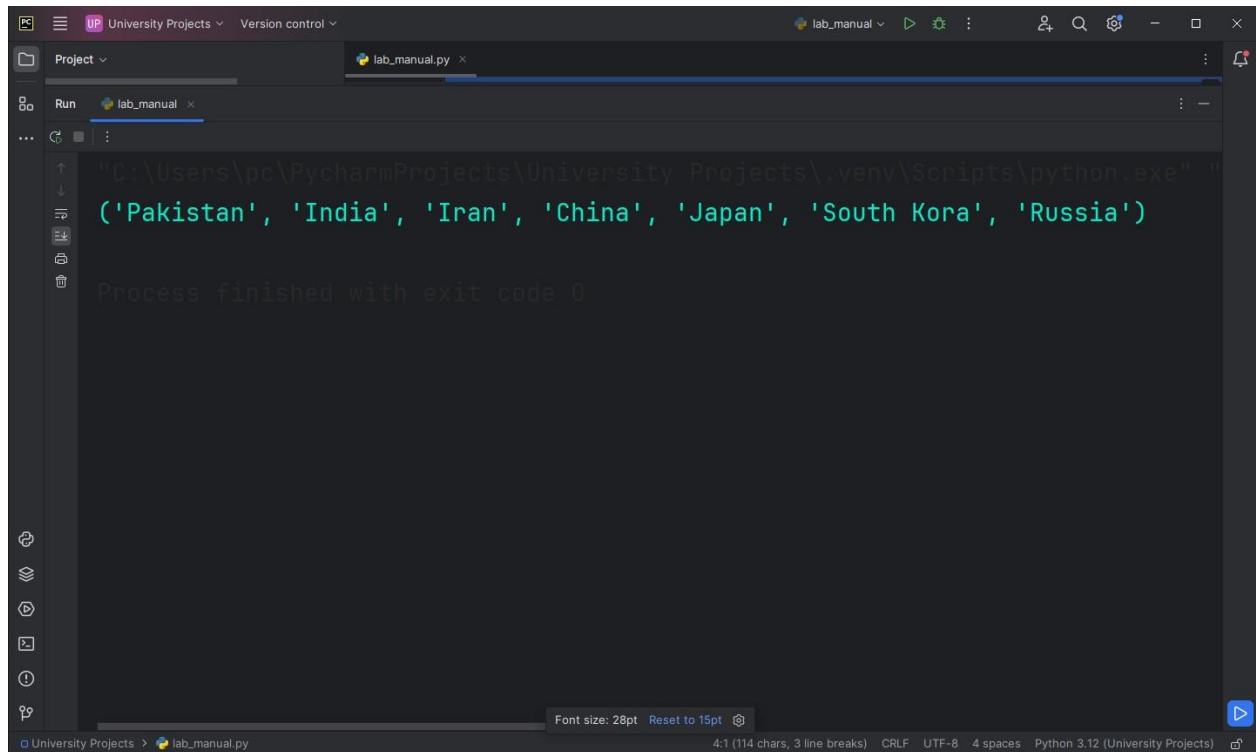
PYTHON LAB WORK 4 | TUPLES

Related to Insert Values in Tuples:

1. Make a tuple to add items of Asian countries names

```
asian_countries = ('Pakistan', 'India',  
'Iran', 'China', 'Japan', 'South Kora',  
'Russia')  
  
print(asian_countries)
```

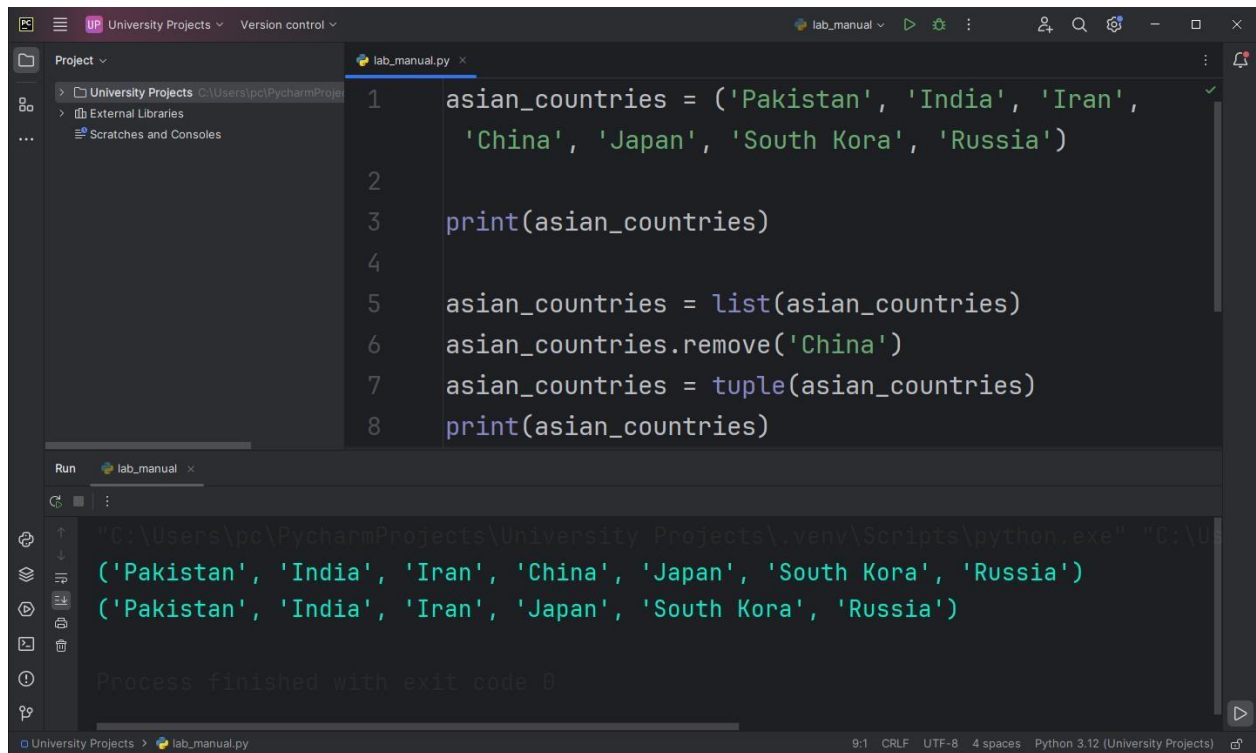
Output:



```
"D:\Users\pc\PycharmProjects\University Projects\.venv\Scripts\python.exe" "  
(('Pakistan', 'India', 'Iran', 'China', 'Japan', 'South Kora', 'Russia'))  
Process finished with exit code 0
```

Related to Delete Items in Tuples:

1. Delete the name of China from previous lab 1 from Asian countries tuple



The screenshot shows the PyCharm IDE interface. The top toolbar includes icons for file operations, a search icon, and a play button. The left sidebar shows the 'Project' view with 'University Projects' and 'External Libraries'. The main editor window displays a file named 'lab_manual.py' with the following Python code:

```
1 asian_countries = ('Pakistan', 'India', 'Iran',  
2 'China', 'Japan', 'South Kora', 'Russia')  
3  
4 print(asian_countries)  
5  
6 asian_countries = list(asian_countries)  
7 asian_countries.remove('China')  
8 asian_countries = tuple(asian_countries)  
9 print(asian_countries)
```

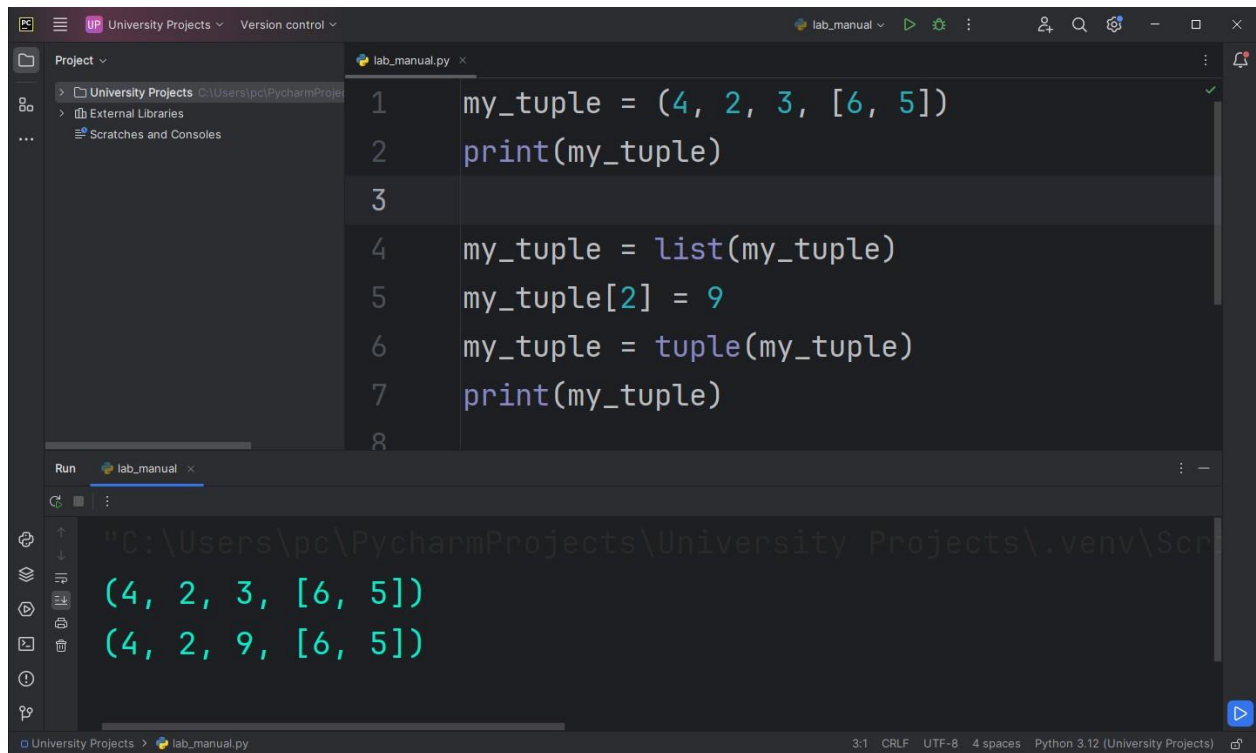
Below the editor is the 'Run' console. It shows the command: `"C:\Users\pc\PycharmProjects\University Projects\.venv\Scripts\python.exe" "C:\Users\pc\PycharmProjects\University Projects\lab_manual.py"`. The output is:

```
('Pakistan', 'India', 'Iran', 'China', 'Japan', 'South Kora', 'Russia')  
('Pakistan', 'India', 'Iran', 'Japan', 'South Kora', 'Russia')
```

The console also indicates 'Process finished with exit code 0'. The bottom status bar shows '9:1 CRLF UTF-8 4 spaces Python 3.12 (University Projects)'.

Related to Changes in Tuples:

1. `my_tuple = (4, 2, 3, [6, 5])` in the above given tuple, change the value of 3 to 9



The screenshot shows the PyCharm IDE interface. The main editor window displays a Python script in `lab_manual.py` with the following code:

```
1 my_tuple = (4, 2, 3, [6, 5])
2 print(my_tuple)
3
4 my_tuple = list(my_tuple)
5 my_tuple[2] = 9
6 my_tuple = tuple(my_tuple)
7 print(my_tuple)
8
```

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

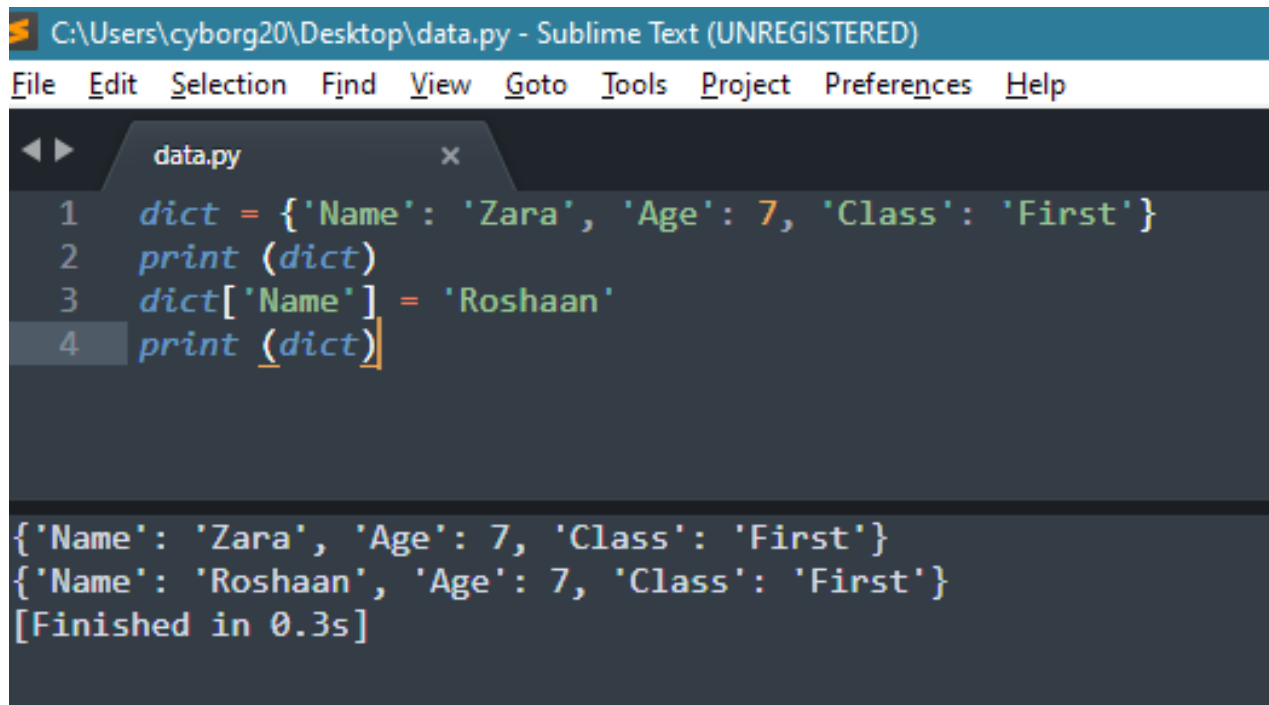
```
"C:\Users\pc\PycharmProjects\University Projects\.venv\Scr
(4, 2, 3, [6, 5])
(4, 2, 9, [6, 5])
```

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.

PYTHON LAB WORK 5 | DICTIONARY

Related to Updating a Dictionary:

1. Write a dictionary and update any value in it `dict = {'Name': 'Zara', 'Age': 7, 'Class': 'First'}` change the name in the above dictionary to something else



```
C:\Users\cyborg20\Desktop\data.py - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

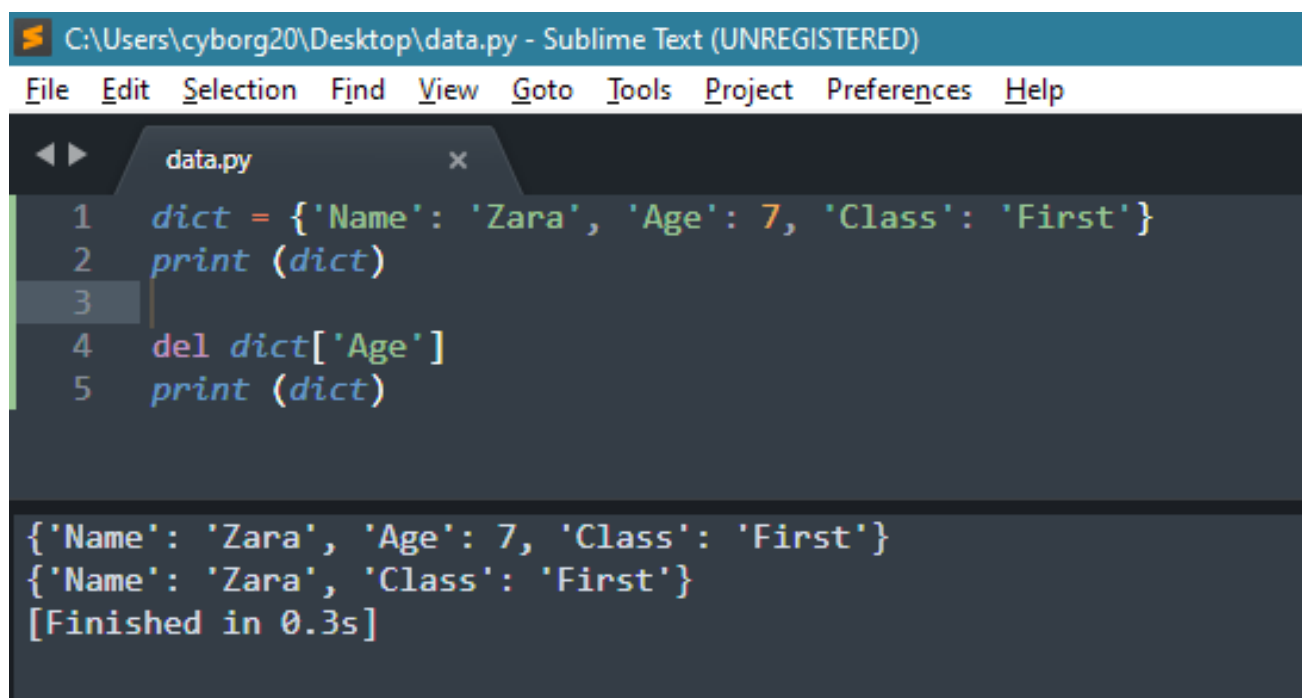
data.py x
1 dict = {'Name': 'Zara', 'Age': 7, 'Class': 'First'}
2 print (dict)
3 dict['Name'] = 'Roshaan'
4 print (dict)

{'Name': 'Zara', 'Age': 7, 'Class': 'First'}
{'Name': 'Roshaan', 'Age': 7, 'Class': 'First'}
[Finished in 0.3s]
```

Related to Updating a Dictionary:

2. dict = {'Name': 'Zara', 'Age': 7, 'Class': 'First'}

Remove Age from the above dictionary



```
C:\Users\cyborg20\Desktop\data.py - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

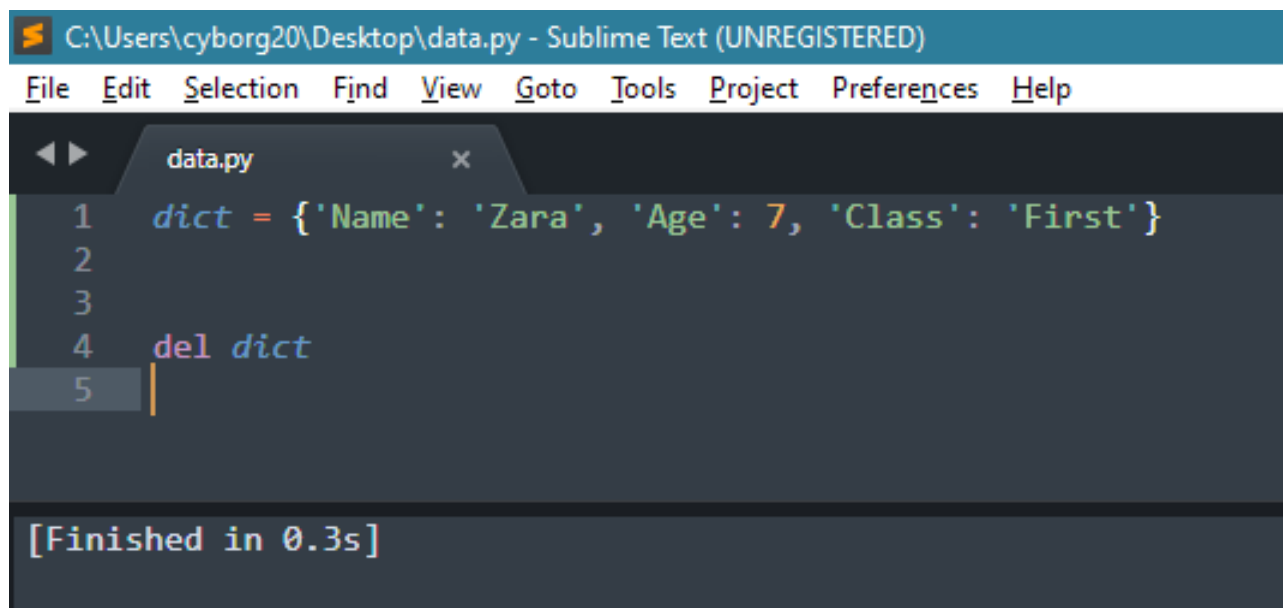
data.py x
1 dict = {'Name': 'Zara', 'Age': 7, 'Class': 'First'}
2 print (dict)
3
4 del dict['Age']
5 print (dict)

{'Name': 'Zara', 'Age': 7, 'Class': 'First'}
{'Name': 'Zara', 'Class': 'First'}
[Finished in 0.3s]
```

Related to Deleting a Dictionary:

3. `py_dict = {1:'a',2:'b',3:'c'}`

How do you delete the whole dictionary of `py_dict`?



The screenshot shows a Sublime Text editor window titled "C:\Users\cyborg20\Desktop\data.py - Sublime Text (UNREGISTERED)". The menu bar includes File, Edit, Selection, Find, View, Goto, Tools, Project, Preferences, and Help. The editor has a tab for "data.py". The code in the file is as follows:

```
1 dict = {'Name': 'Zara', 'Age': 7, 'Class': 'First'}
2
3
4 del dict
5
```

At the bottom of the window, a status bar indicates "[Finished in 0.3s]".

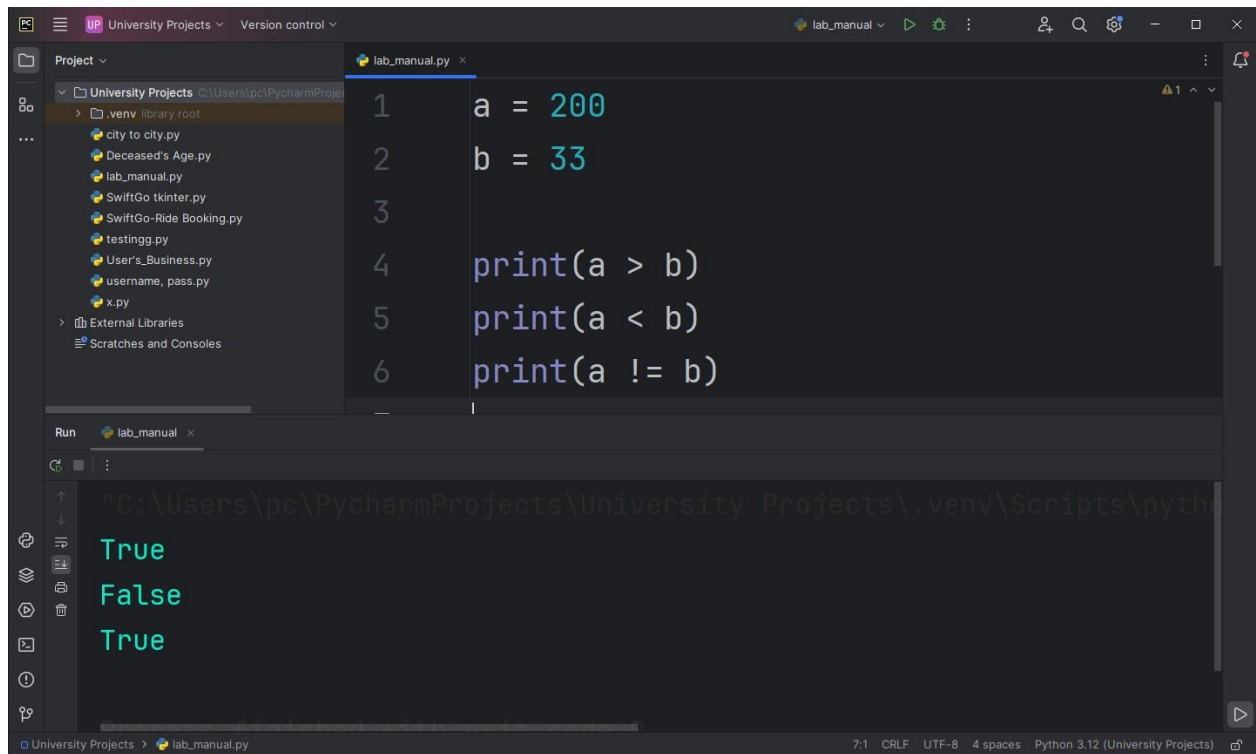
PYTHON LAB WORK 6 | CONDITIONAL STATEMENTS

Related to Comparison Operators of greater than > and less than <:

1. `a = 200`

`b = 33`

compare these two variables with less than and greater than operators and print the output? Also write another program for NOT EQUAL TO operator also



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

```
1 a = 200
2 b = 33
3
4 print(a > b)
5 print(a < b)
6 print(a != b)
```

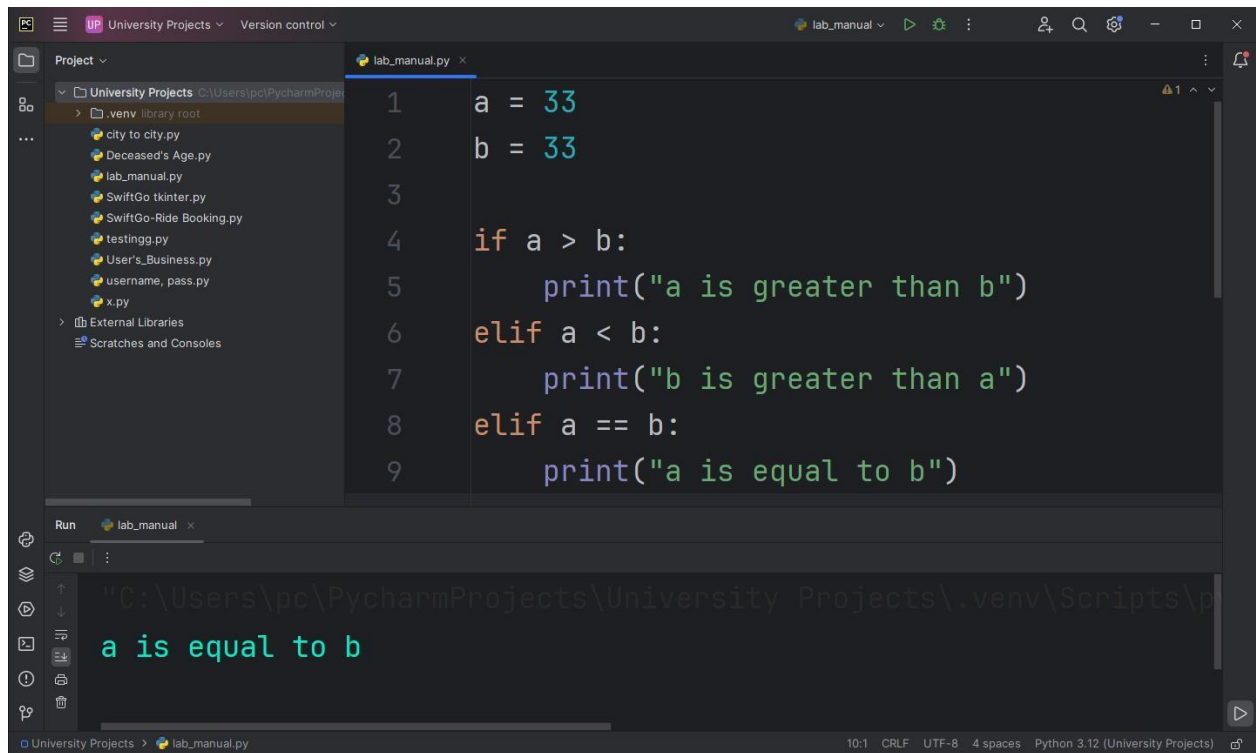
The Run console at the bottom shows the output of the script:

```
"C:\Users\pc\PycharmProjects\University Projects\.venv\Scripts\python.exe"
True
False
True
```

Related to Comparison Operators of greater than > and less than <:

2. a = 33 b
= 33

Write a single program to write ELIF and ELSE condition.
Also, write a shorthand version also.



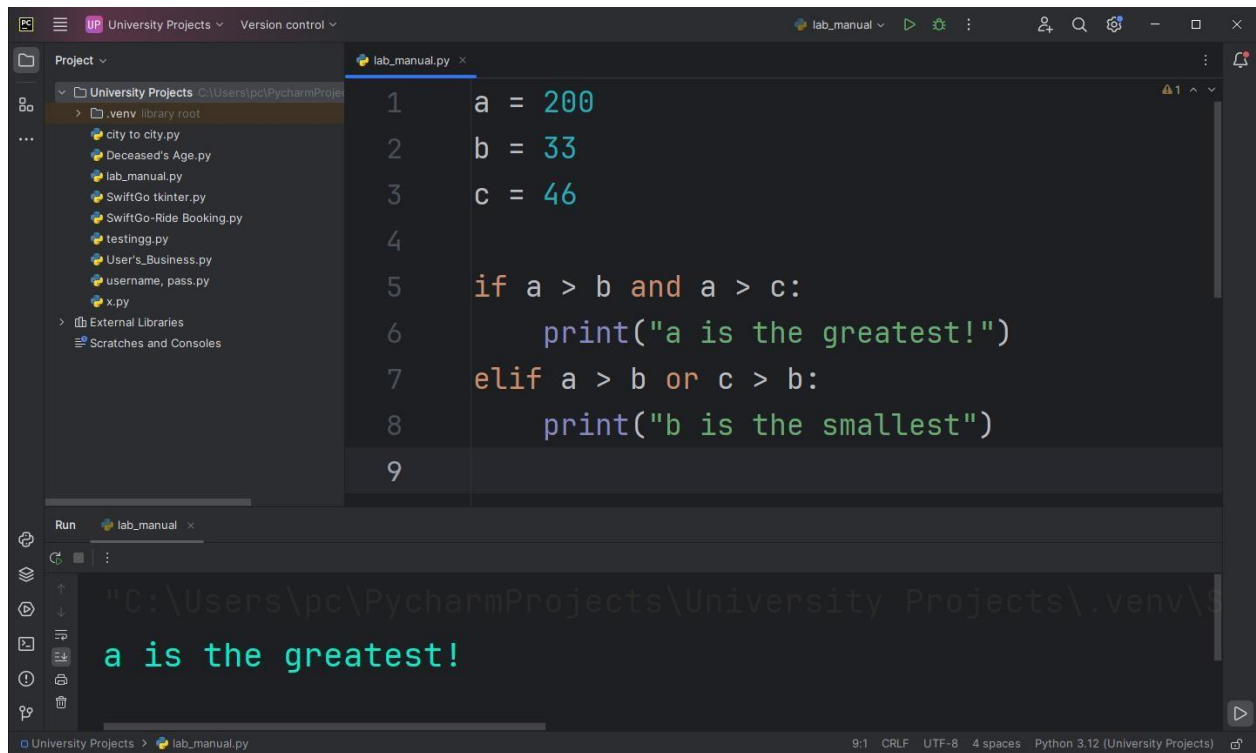
The screenshot shows the PyCharm IDE interface. The top toolbar includes icons for file operations, a search icon, and a play button. The left sidebar displays the 'Project' view with a tree structure of files and folders under 'University Projects'. The main editor window shows a Python file named 'lab_manual.py' with the following code:

```
1 a = 33
2 b = 33
3
4 if a > b:
5     print("a is greater than b")
6 elif a < b:
7     print("b is greater than a")
8 elif a == b:
9     print("a is equal to b")
```

Below the editor is the 'Run' console, which shows the output of the script: 'a is equal to b'. The status bar at the bottom indicates the file encoding (UTF-8), indentation (4 spaces), and the Python interpreter (Python 3.12).

Related to AND & OR:

1. Make three variables and compare their values with AND, OR



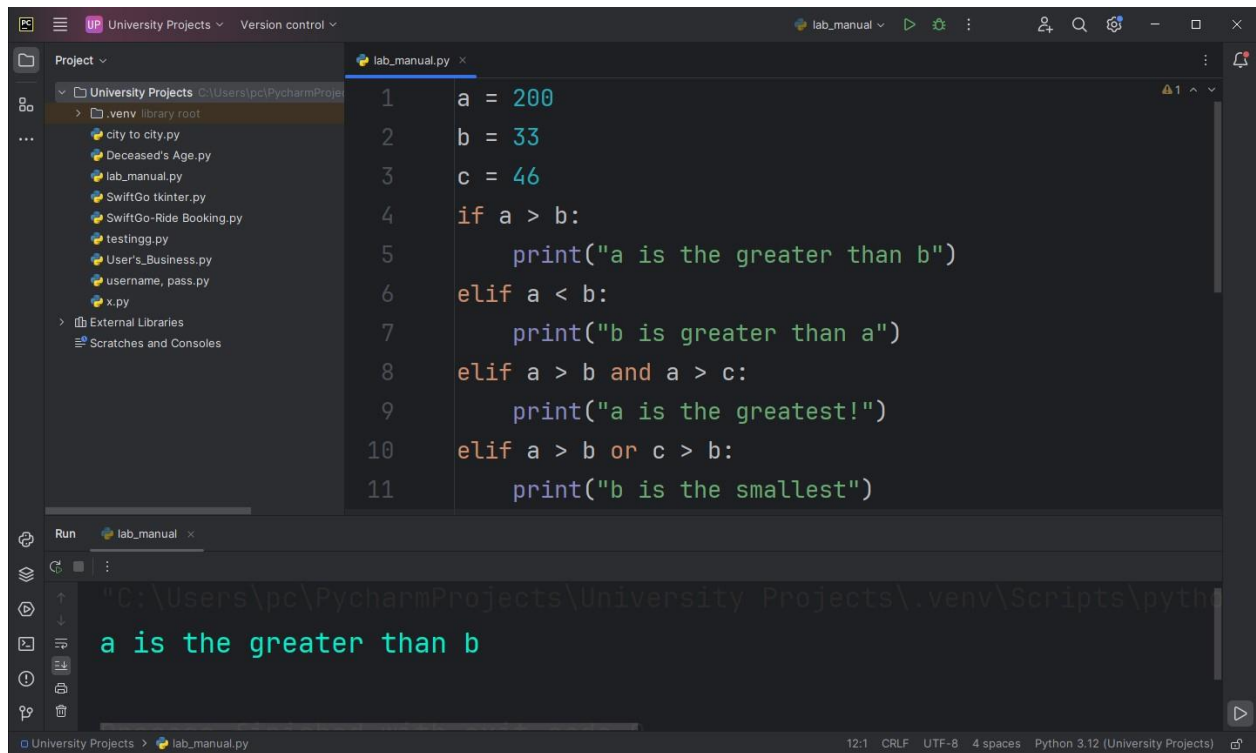
The screenshot shows the PyCharm IDE interface. The top toolbar includes icons for file operations, a search icon, and a play button. The left sidebar displays the 'Project' view with a tree structure of files and folders under 'University Projects'. The main editor window shows a Python file named 'lab_manual.py' with the following code:

```
1 a = 200
2 b = 33
3 c = 46
4
5 if a > b and a > c:
6     print("a is the greatest!")
7 elif a > b or c > b:
8     print("b is the smallest")
9
```

Below the editor is the 'Run' console, which shows the output of the program: 'a is the greatest!'. The status bar at the bottom indicates the file encoding as UTF-8, 4 spaces, and Python 3.12 (University Projects).

Related to Nested if:

1. Write a NESTED IF program with different conditional statements like <, >, AND, OR etc.



The screenshot shows the PyCharm IDE interface. On the left, the 'Project' view displays a file tree for 'University Projects' with files like 'city to city.py', 'Deceased's Age.py', 'lab_manual.py', 'SwiftGo tkinter.py', 'SwiftGo-Ride Booking.py', 'testingg.py', 'User's_Business.py', 'username, pass.py', and 'x.py'. The main editor window shows the code in 'lab_manual.py' with line numbers 1 through 11. The code defines variables a, b, and c, and uses conditional statements to print the greatest or smallest value. The 'Run' window at the bottom shows the output: 'a is the greater than b'. 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.

```
1 a = 200
2 b = 33
3 c = 46
4 if a > b:
5     print("a is the greater than b")
6 elif a < b:
7     print("b is greater than a")
8 elif a > b and a > c:
9     print("a is the greatest!")
10 elif a > b or c > b:
11     print("b is the smallest")
```

Run lab_manual

"C:\Users\pc\PycharmProjects\University Projects\.venv\Scripts\python.exe"

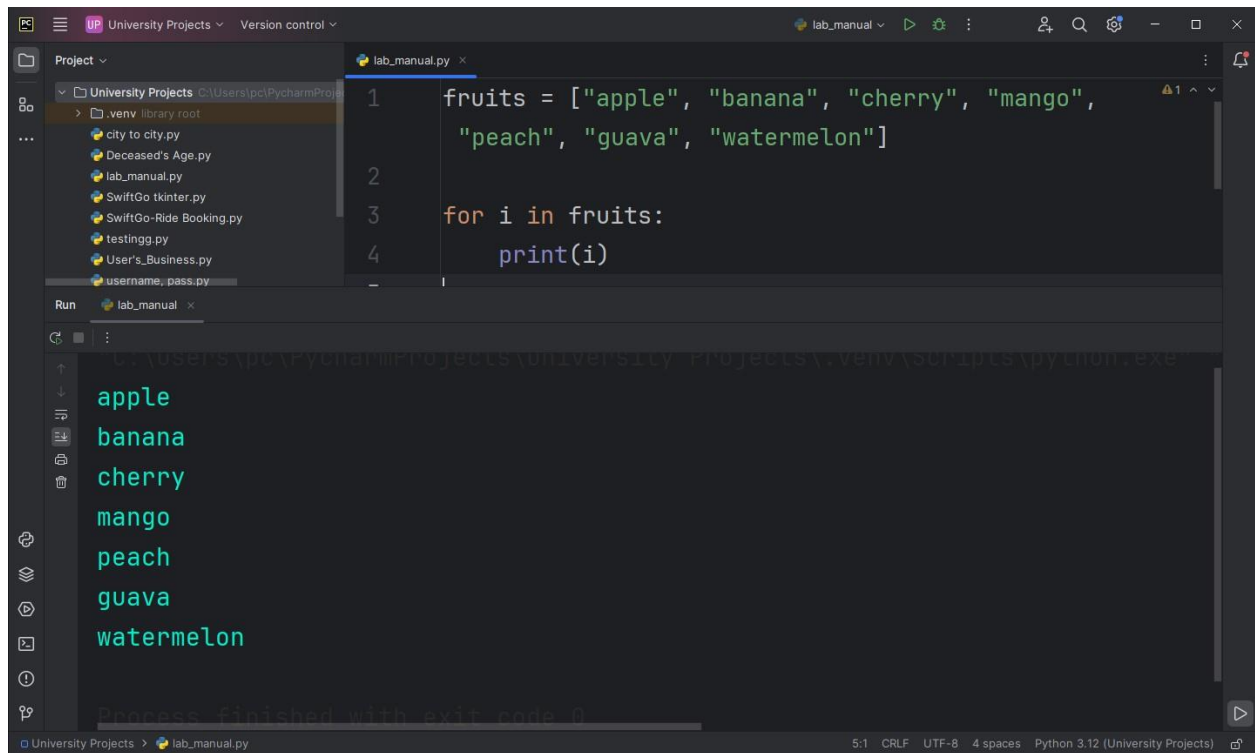
a is the greater than b

University Projects > lab_manual.py 12:1 CRLF UTF-8 4 spaces Python 3.12 (University Projects)

PYTHON LAB WORK 7 | ITERATIONS

Related to For Loops:

1. Loop through this array of fruits `fruits = ["apple", "banana", "cherry", "mango", "peach", "guava", "watermelon"]`



The screenshot shows the PyCharm IDE with a project named 'University Projects'. The file explorer on the left shows a directory structure with files like 'city to city.py', 'Deceased's Age.py', 'lab_manual.py', 'SwiftGo tkinter.py', 'SwiftGo-Ride Booking.py', 'testingg.py', 'User's_Business.py', and 'username, pass.py'. The main editor window displays a Python script named 'lab_manual.py' with the following code:

```
1 fruits = ["apple", "banana", "cherry", "mango",  
2         "peach", "guava", "watermelon"]  
3  
4 for i in fruits:  
5     print(i)
```

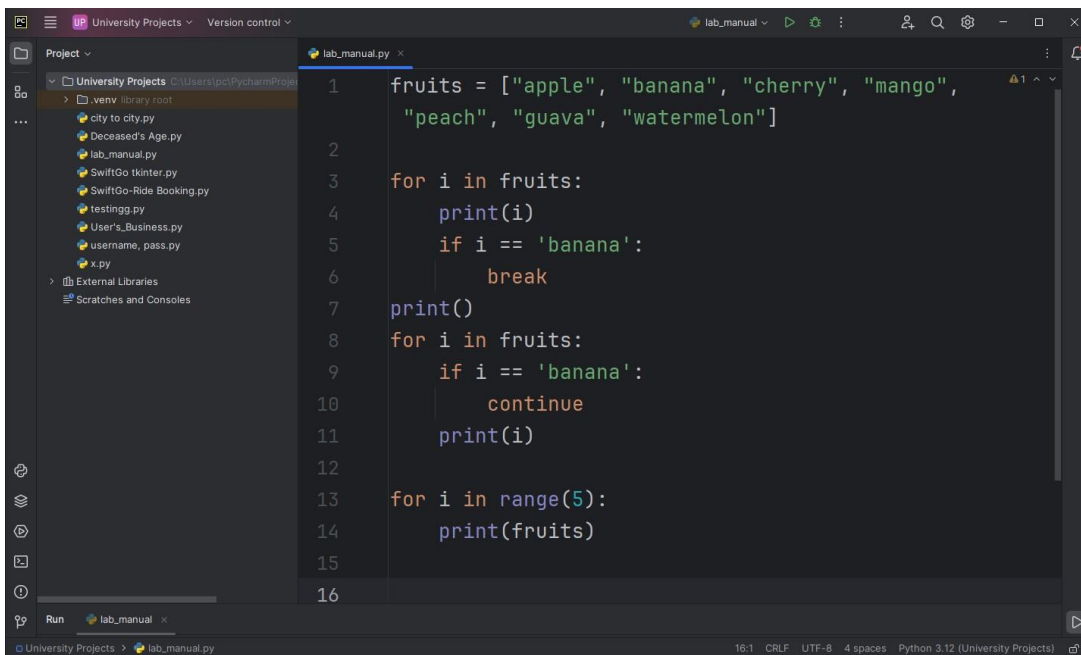
The Run window at the bottom shows the output of the script:

```
apple  
banana  
cherry  
mango  
peach  
guava  
watermelon
```

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.

Related to For Break, Continue and Range:

1. Write another program to add break and continue statements Write the RANGE of the fruits array

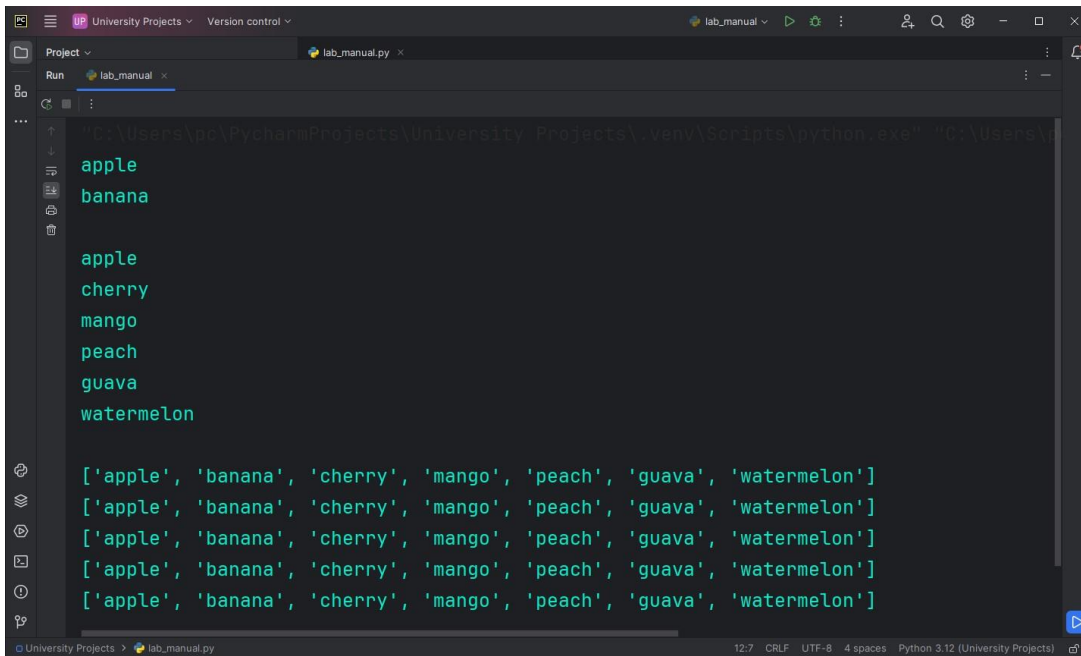


The screenshot shows the PyCharm IDE with the same project and file explorer as the first image. The main editor window displays a Python script named 'lab_manual.py' with the following code:

```
1 fruits = ["apple", "banana", "cherry", "mango",  
2         "peach", "guava", "watermelon"]  
3  
4 for i in fruits:  
5     print(i)  
6     if i == 'banana':  
7         break  
8  
9 print()  
10 for i in fruits:  
11     if i == 'banana':  
12         continue  
13     print(i)  
14  
15 for i in range(5):  
16     print(fruits)
```

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.

Output:



```
"C:\Users\pc\PycharmProjects\University Projects\.venv\Scripts\python.exe" "C:\Users\pc\PycharmProjects\University Projects\lab_manual.py"

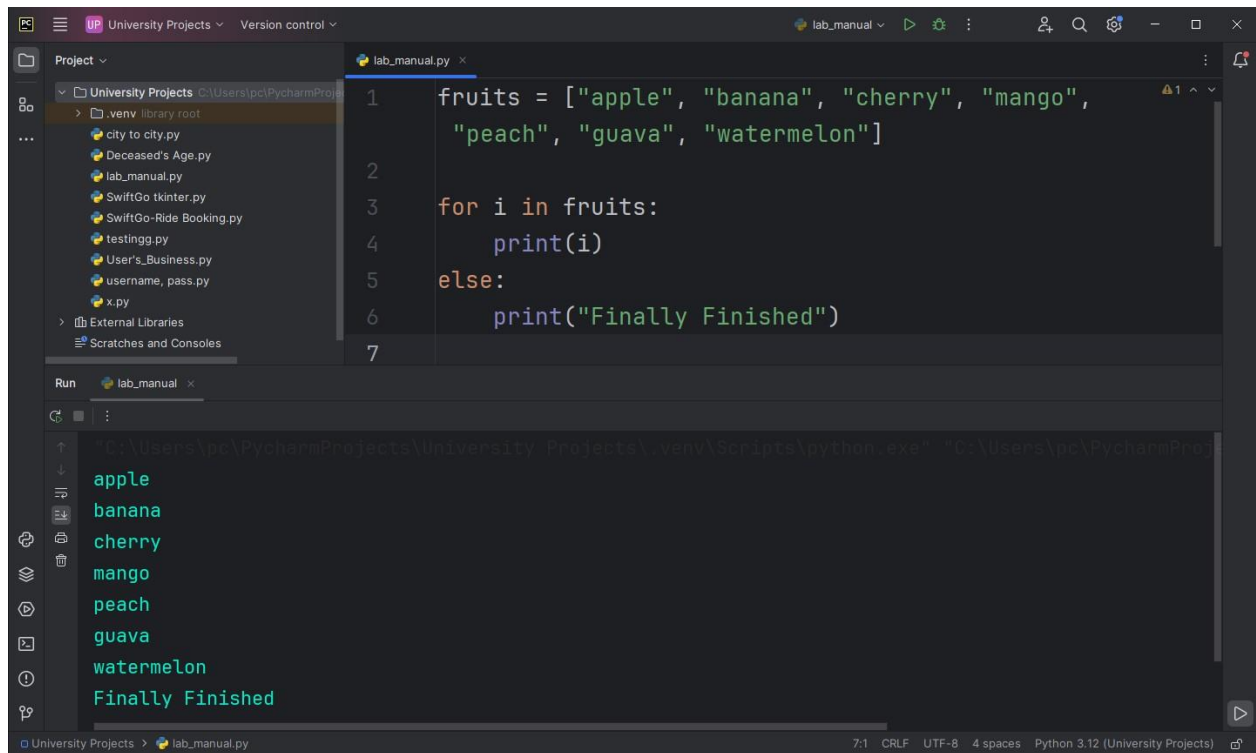
apple
banana

apple
cherry
mango
peach
guava
watermelon

['apple', 'banana', 'cherry', 'mango', 'peach', 'guava', 'watermelon']
['apple', 'banana', 'cherry', 'mango', 'peach', 'guava', 'watermelon']
['apple', 'banana', 'cherry', 'mango', 'peach', 'guava', 'watermelon']
['apple', 'banana', 'cherry', 'mango', 'peach', 'guava', 'watermelon']
['apple', 'banana', 'cherry', 'mango', 'peach', 'guava', 'watermelon']
```

Related to for Else:

1. fruits = ["apple", "banana", "cherry", "mango", "peach", "guava", "watermelon"]
write a program that uses ELSE with the above array



The screenshot shows the PyCharm IDE interface. The top toolbar includes icons for file operations, a search icon, and a play button. The 'Project' sidebar on the left shows a tree view of the 'University Projects' directory, with files like 'city to city.py', 'Deceased's Age.py', 'lab_manual.py', 'SwiftGo tkinter.py', 'SwiftGo-Ride Booking.py', 'testingg.py', 'User's_Business.py', 'username, pass.py', and 'x.py'. The main editor window displays the file 'lab_manual.py' with the following Python code:

```
1 fruits = ["apple", "banana", "cherry", "mango",  
2         "peach", "guava", "watermelon"]  
3  
4 for i in fruits:  
5     print(i)  
6 else:  
7     print("Finally Finished")
```

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

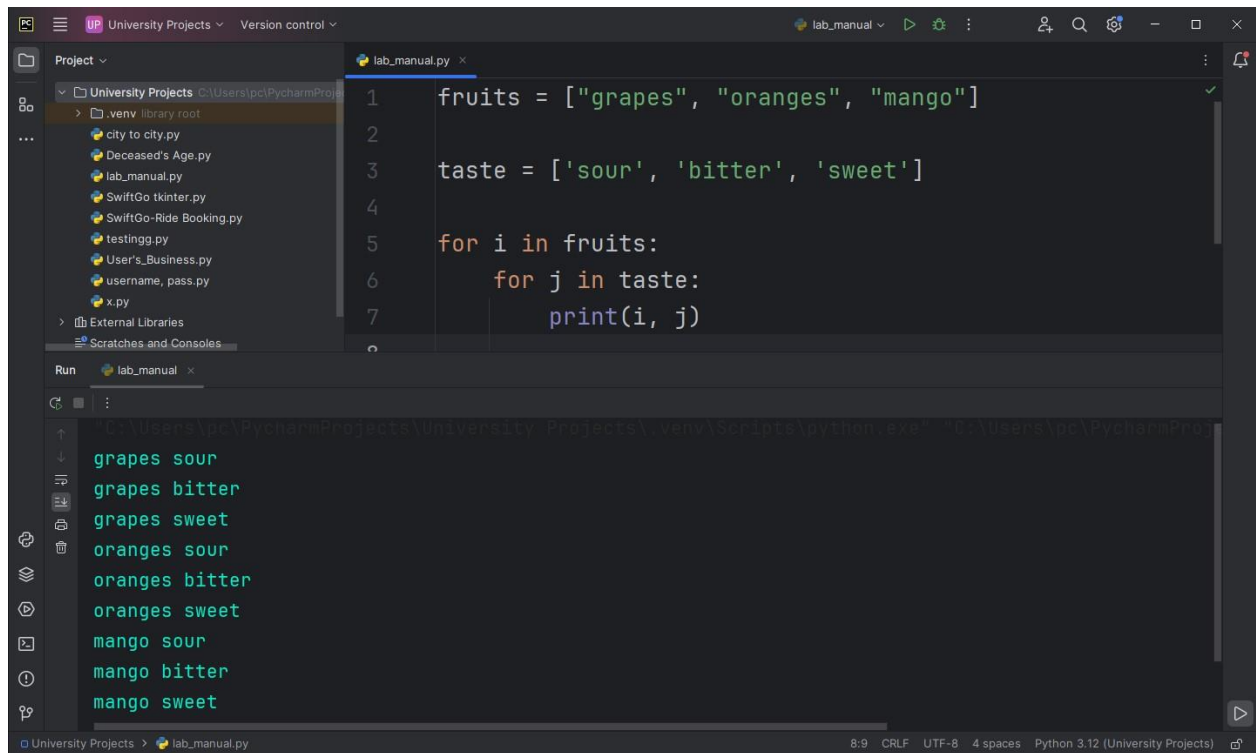
```
apple  
banana  
cherry  
mango  
peach  
guava  
watermelon  
Finally Finished
```

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.

Related to Nested For Loop:

1. taste = ["sour", "bitter", "sweet"]

Write a program that uses a NESTED loop with above array of taste with fruits array



The screenshot shows the PyCharm IDE interface. The top toolbar includes icons for file operations, a search icon, and a play button. The 'Project' sidebar on the left lists files under 'University Projects', including 'city to city.py', 'Deceased's Age.py', 'lab_manual.py', 'SwiftGo tkinter.py', 'SwiftGo-Ride Booking.py', 'testingg.py', 'User's_Business.py', 'username, pass.py', and 'x.py'. The main editor window displays the file 'lab_manual.py' with the following Python code:

```
1 fruits = ["grapes", "oranges", "mango"]
2
3 taste = ['sour', 'bitter', 'sweet']
4
5 for i in fruits:
6     for j in taste:
7         print(i, j)
```

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

```
grapes sour
grapes bitter
grapes sweet
oranges sour
oranges bitter
oranges sweet
mango sour
mango bitter
mango sweet
```

The status bar at the bottom indicates the file is 'lab_manual.py', the encoding is 'UTF-8', and the interpreter is 'Python 3.12 (University Projects)'.

PYTHON LAB WORK 8 | FUNCTIONS

Related to Function creating and Adding Arguments:

1. Write a function that print your name and write a function with one argument and print three names

```
C:\Users\cyborg20\Desktop\data.py - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

data.py x
1 def my_name():
2     print ("My Name is Roshaan")
3 my_name()
4
5 def diff_names(name):
6     print ("My Name is", name)
7
8 diff_names("Sohaib")
9 diff_names("Mushaf")
10 diff_names("Shamoon")

My Name is Roshaan
My Name is Sohaib
My Name is Mushaf
My Name is Shamoon
[Finished in 0.3s]
```

Related to *args & **kwargs:

1. Write a function that takes 5 arguments, but those arguments must not be defined, so you must decide what to use *args or **kwargs

