Data Visualization with Python (Lab 9)

Data Visualization

Dr. Tam Nguyen

tamnguyen@udayton.edu

Objective

- Using Python for Data Visualization
  - Install Python IDE
  - Use Python to plot charts/diagrams with the given data

Matplotlib

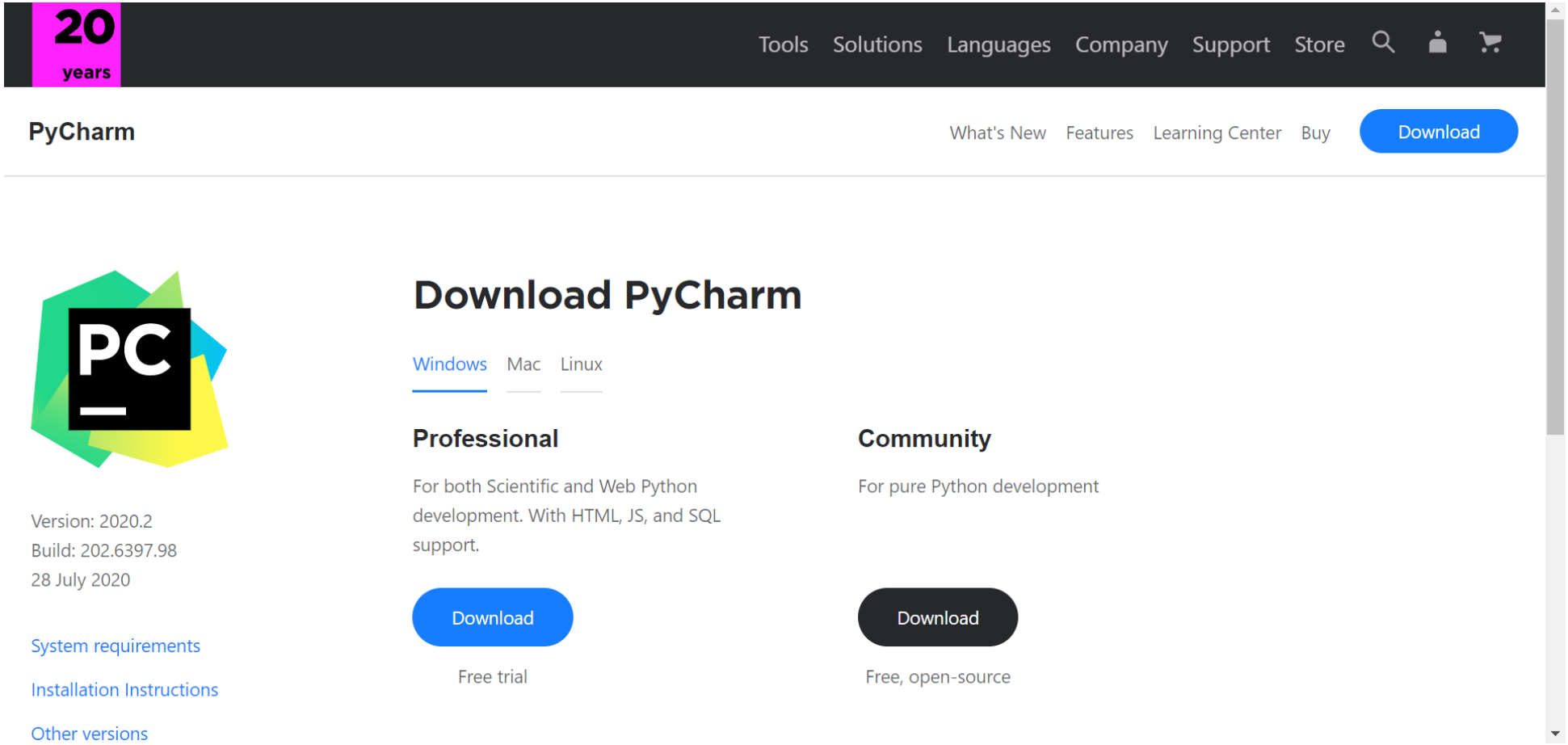
- Matplotlib is a visualization library in Python for 2D plots of arrays.
- In particular, Matplotlib is a multi-platform data visualization library built on NumPy arrays and designed to work with the broader SciPy stack.

What can be drawn with Matplotlib?

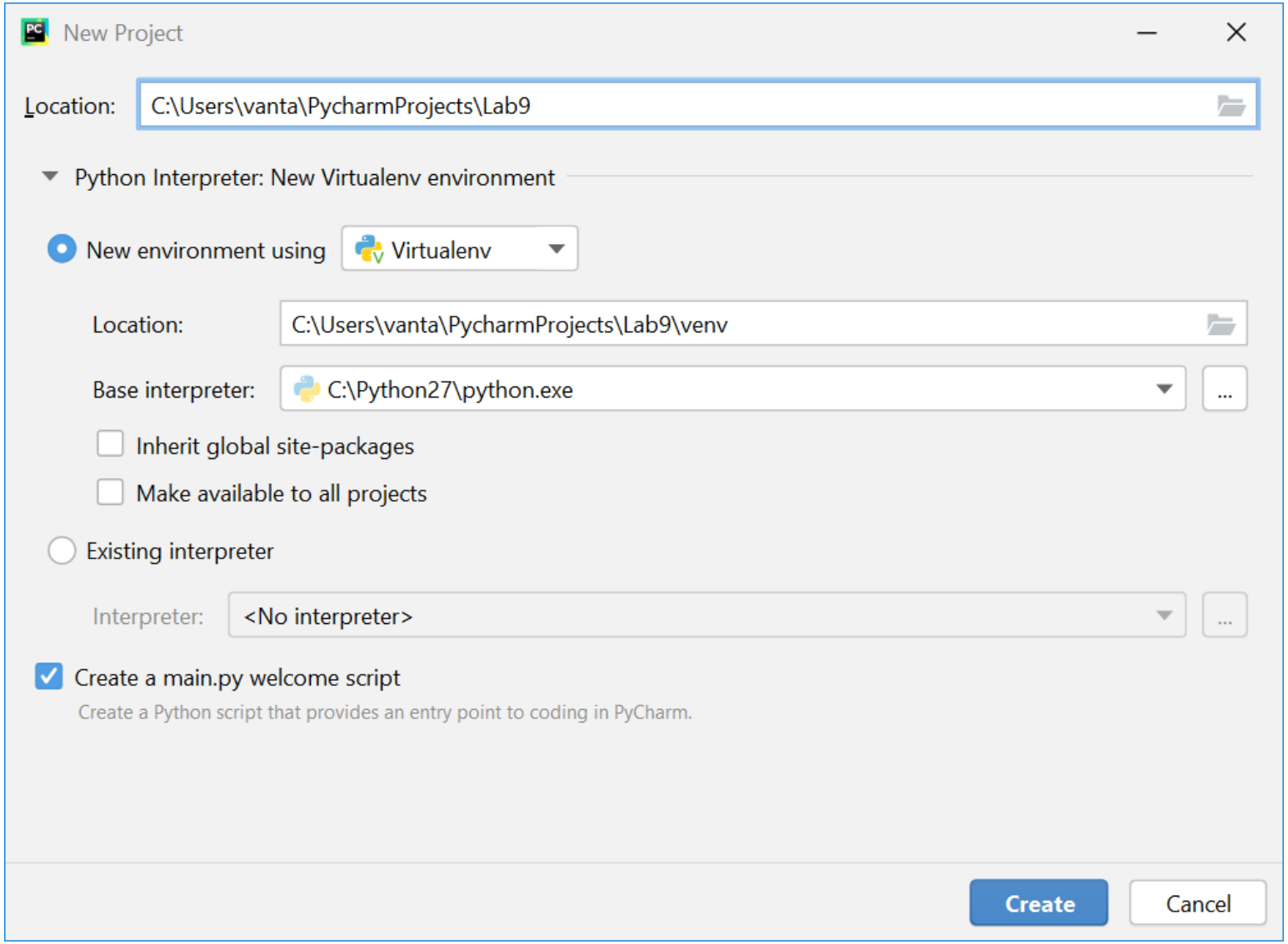
- Matplotlib comes with a wide variety of plots
  - Basic graphs: bar graph, line chart, histogram, scatter plot
  - Advanced graphs: treemap, parallel coordinates.

Install PyCharm IDE

- https://www.jetbrains.com/pycharm/download/#section=windows



Create Lab9 project



Create Python file: main.py

File Edit View Navigate Code Refactor Run Tools VCS Window Help Lab9 - main.py

Lab9

main

Project

main.py Lab9d.py tk.tcl init.tcl

New
Cut Ctrl+X
Copy
Paste Ctrl+V
Find Usages Alt+F7
Find in Path... Ctrl+Shift+F
Replace in Path... Ctrl+Shift+R
Inspect Code...
Refactor
Clean Python Compiled Files
Add to Favorites
Reformat Code Ctrl+Alt+L
Optimize Imports Ctrl+Alt+O
Show in Explorer
Directory Path Ctrl+Alt+F12
Open in Terminal
Local History
Reload from Disk
Compare With... Ctrl+D
Mark Directory as
Remove BOM
Create Gist...

File
New Scratch File Ctrl+Alt+Shift+Insert
Directory
Python Package
Python File
HTML File
EditorConfig File
Resource Bundle

```
# Use a breakpoint in the code line below to debug your script.
print("Hi, {0}".format(name))  # Press Ctrl+F8 to toggle the breakpoint.


# Press the green button in the gutter to run the script.
if __name__ == '__main__':
```

Run:

```
cts\Lab9\venv\Scripts\python.exe C:/Users/vanta/PycharmProjects/Lab9/main.py

code 0
```

4: Run TODO 6: Problems 5: Debug Terminal Python Console Event Log

Creates a Python file from the specified template 19:1 CRLF UTF-8 4 spaces Python 2.7 (Lab9)

Hello World

main

main.py

```
1  print('Hello World')
2
3
4
5
6
7
8
9
10
11
12
```

15 3

Run: main

```
C:\Users\vanta\PycharmProjects\Lab9\venv\Scripts\python.exe C:/Users/vanta/PycharmProjects/Lab9/main.py
Hello World


Process finished with exit code 0
```

Create a vector and display it

main.py

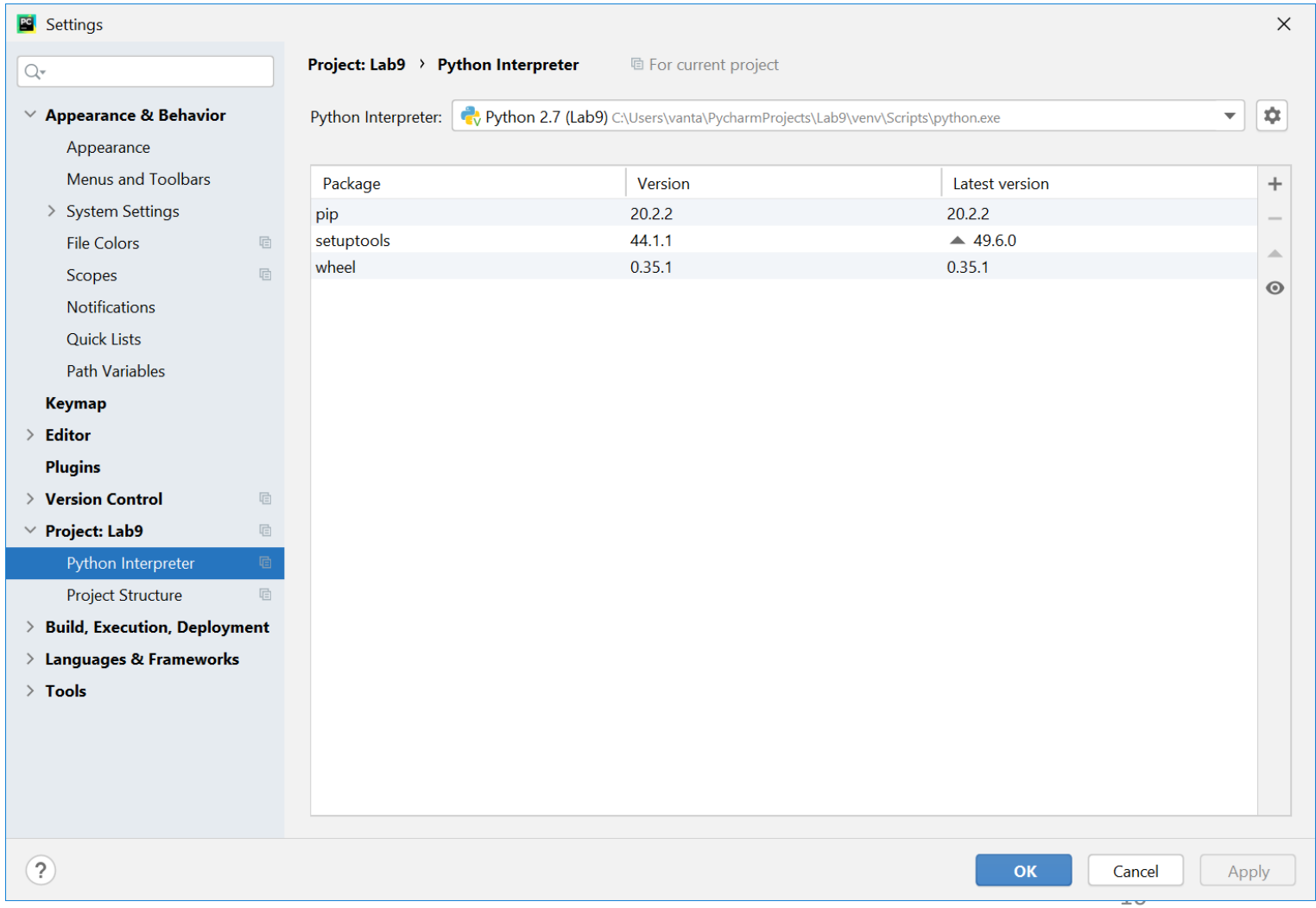
```python
print('Hello World')
a = [1,2,3,4]
print(a)
```

Run: main

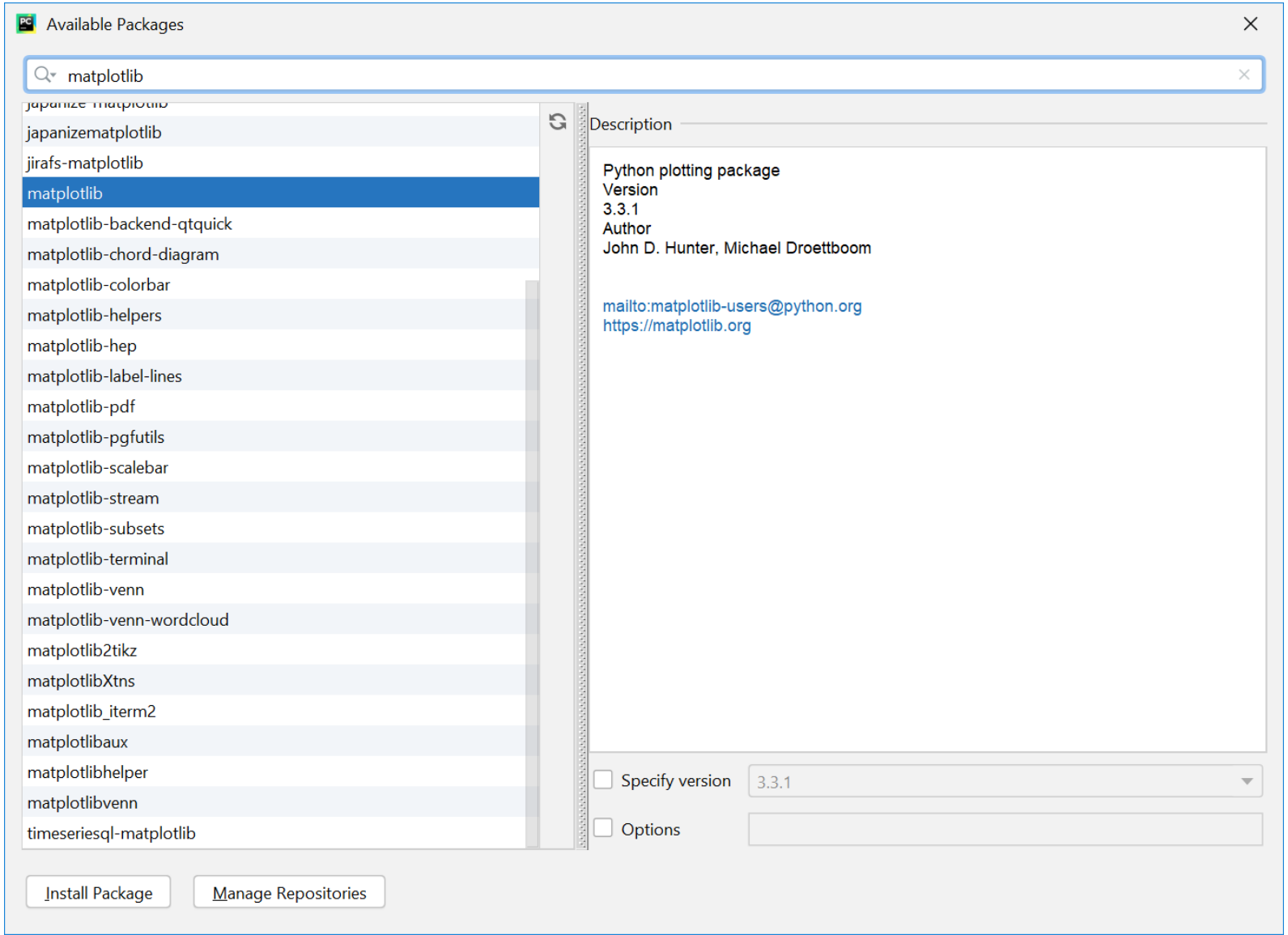
```
C:\Users\vanta\PycharmProjects\Lab9\venv\Scripts\python.exe C:/Users/vanta/PycharmProjects/Lab9/main.py
Hello World
[1, 2, 3, 4]

Process finished with exit code 0
```

Update Settings

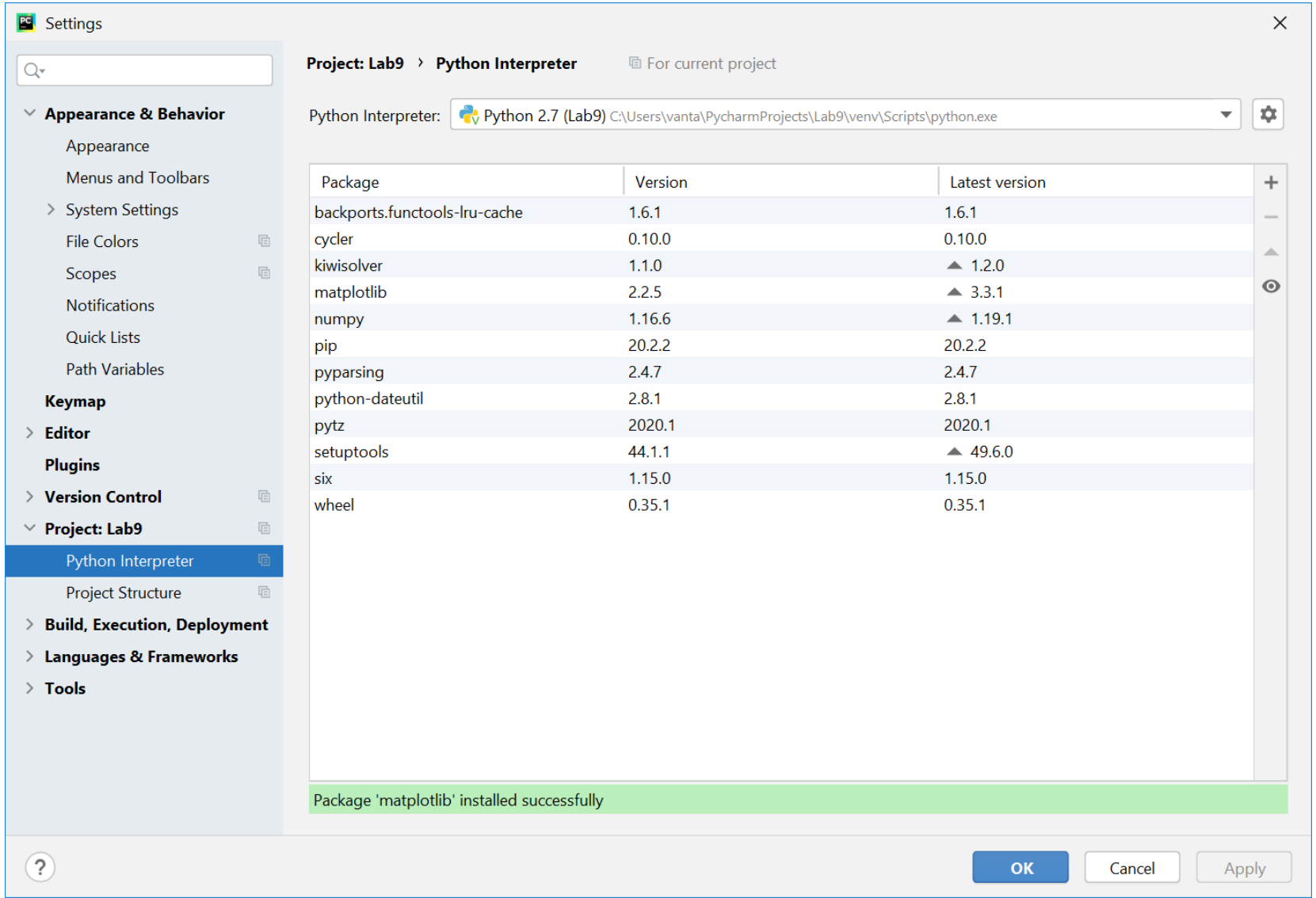
- File\Settings\Project:Lab9



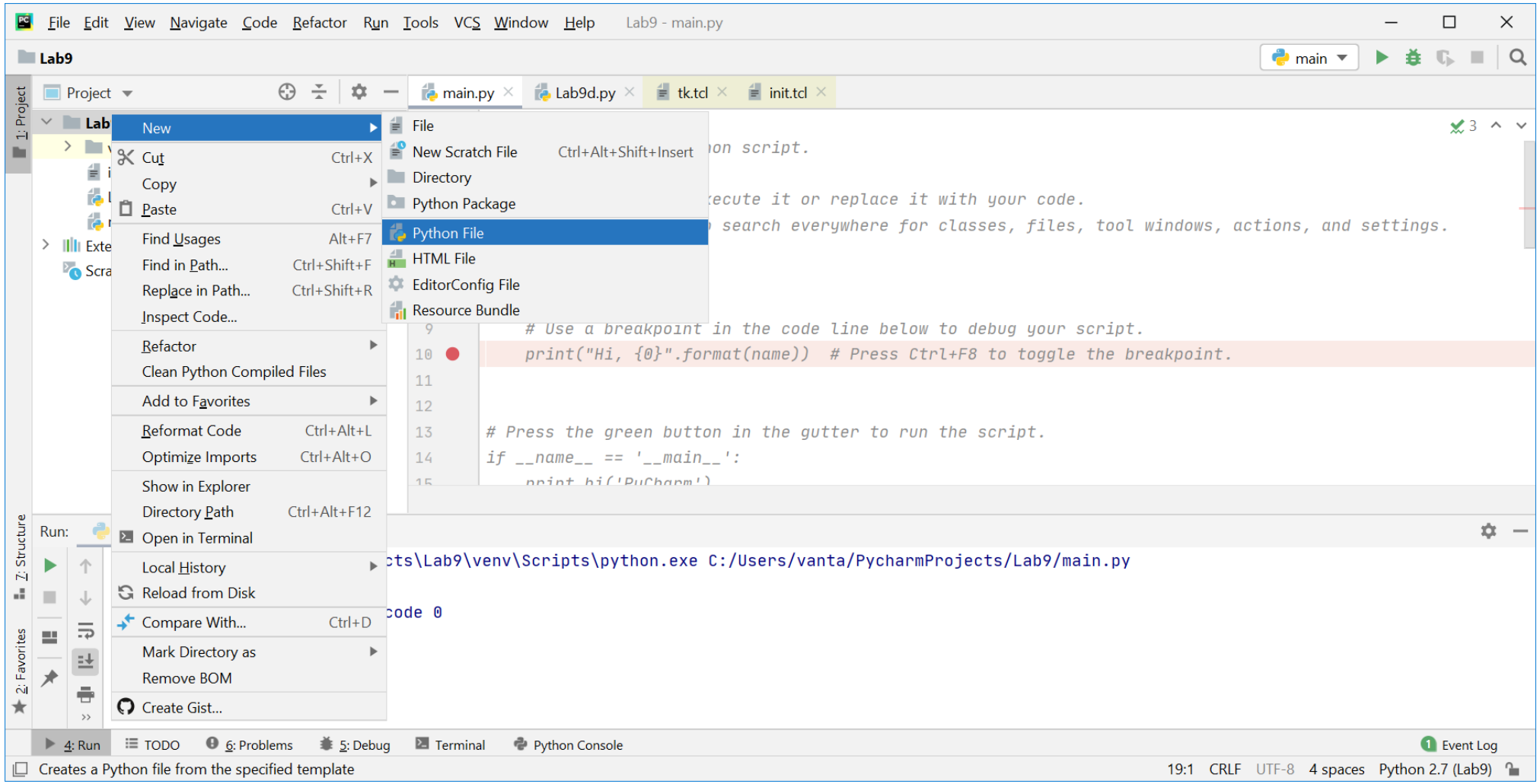
Add matplotlib



Installation finished



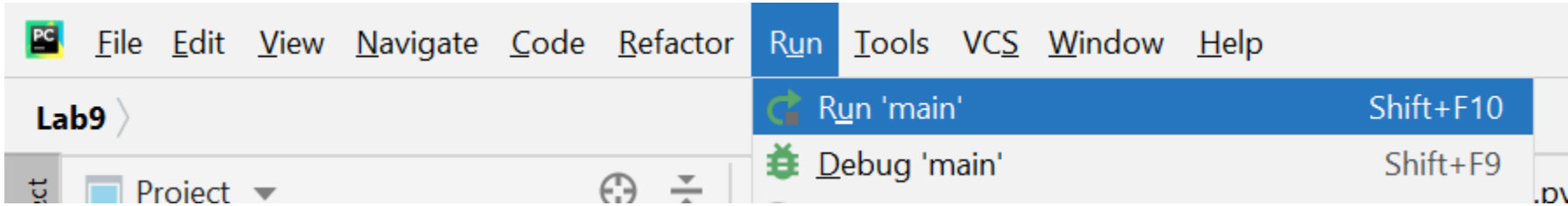
Create Python file: Lab9a.py



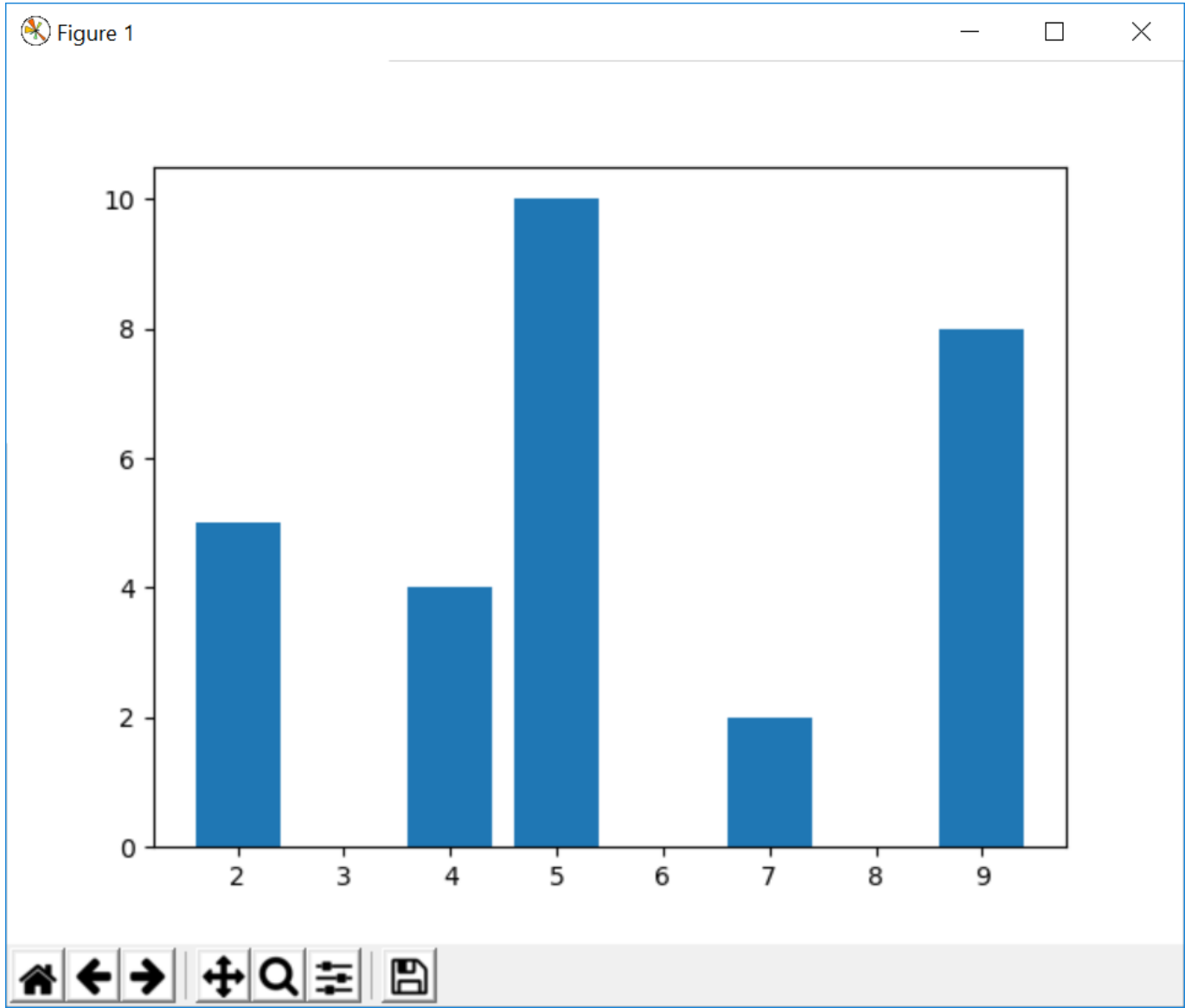
Lab9a.py

```
# importing matplotlib module
from matplotlib import pyplot as plt
def Lab9a():
    # x-axis values
    x = [5, 2, 9, 4, 7]

    # Y-axis values
    y = [10, 5, 8, 4, 2]

    # Create a new figure
    plt.figure(1)

    # Function to plot the bar
    plt.bar(x,y)
    plt.show()
```

main.py




```
from Lab9a import Lab9a
Lab9a()
```

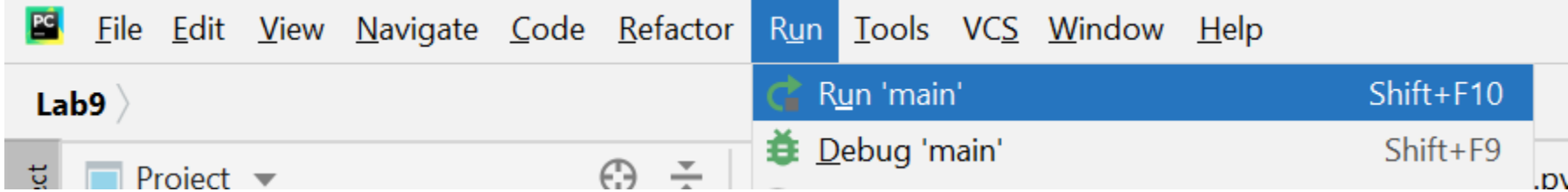


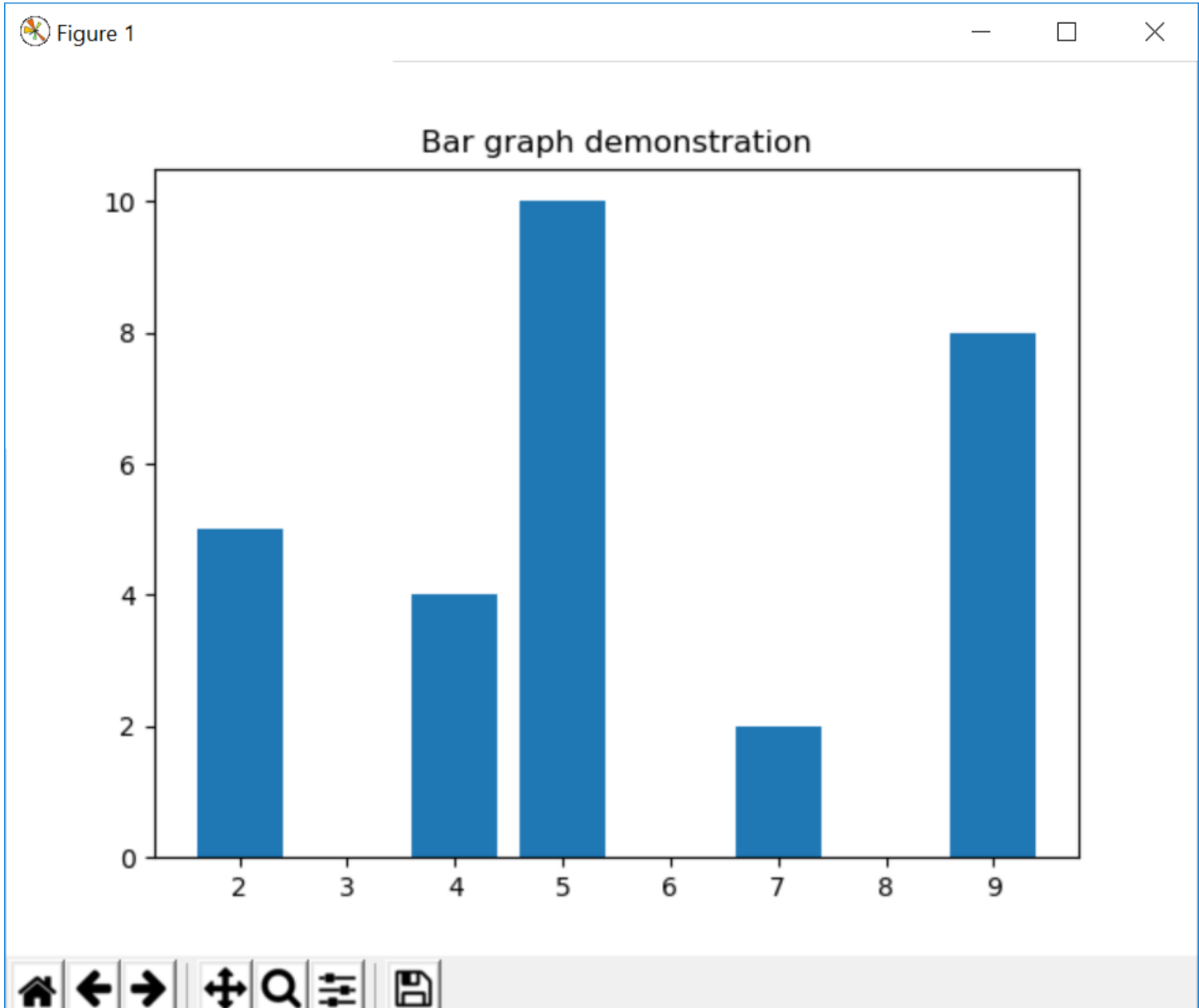
Lab9a.py

```
# importing matplotlib module
from matplotlib import pyplot as plt
def Lab9a():
    # x-axis values
    x = [5, 2, 9, 4, 7]

    # Y-axis values
    y = [10, 5, 8, 4, 2]

    # Create a new figure
    plt.figure(1)
    # Function to plot the bar
    plt.bar(x,y)

    plt.title("Bar graph demonstration")

    plt.show()
```

main.py

```
from Lab9a import Lab9a
Lab9a()
```





Update ./Lab9/venv/tcl/tcl8.5/init.tcl:

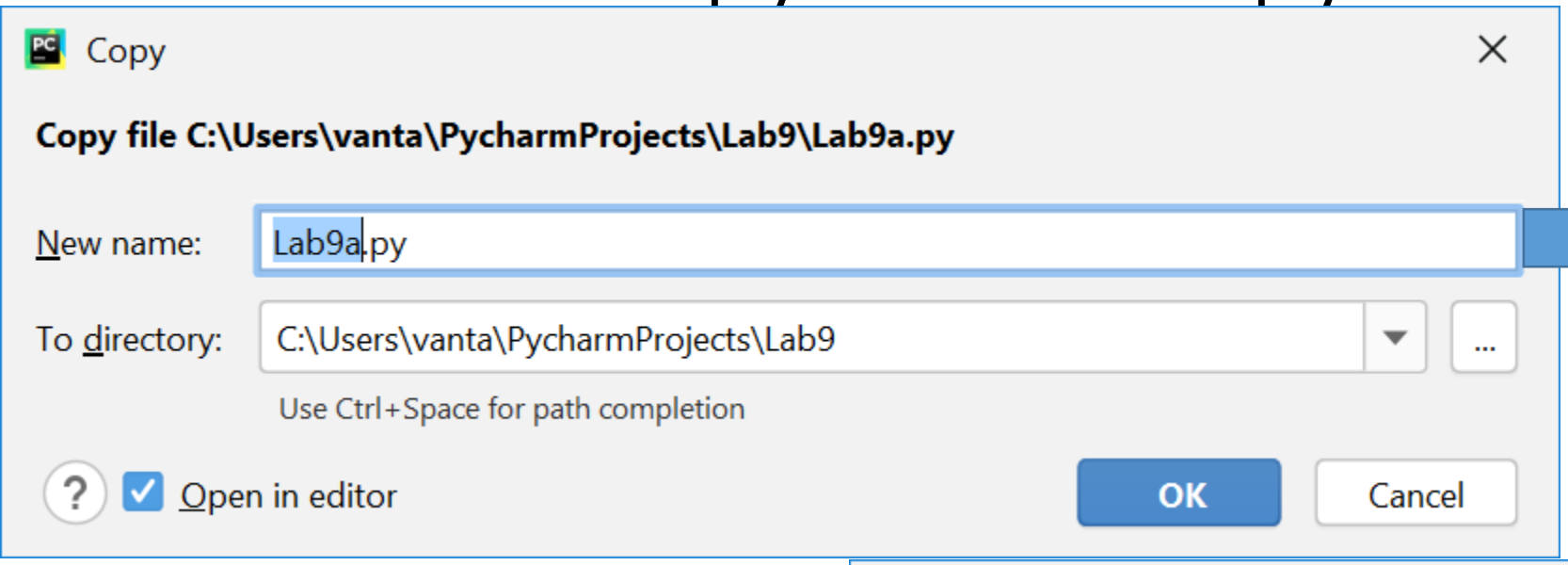
- Fix: package require -exact Tcl 8.5.15

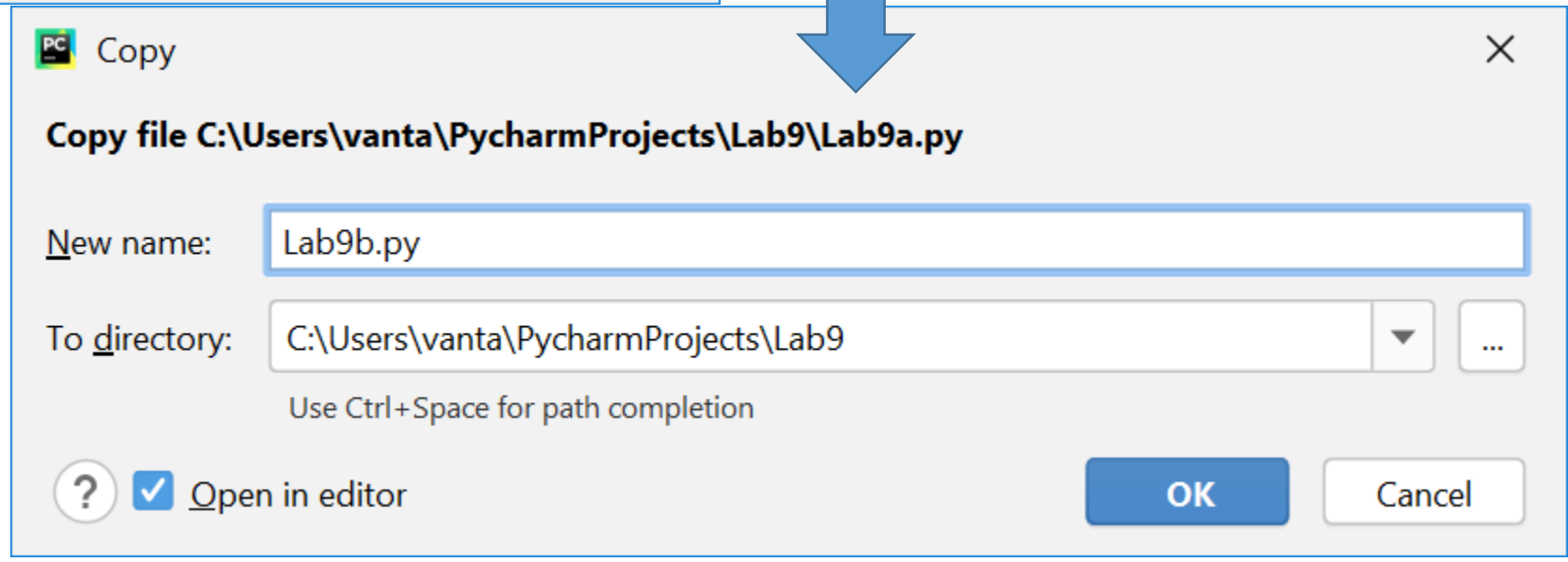
Update ./Lab9/venv/tcl/tk8.5/tk.tcl

- Fix: package require -exact Tk 8.5.15

- You may fix tk.tcl and tcl8.5/init.tcl files in the Python folder

Save Lab9a.py as Lab9b.py





Lab9b.py

```
# importing matplotlib module
from matplotlib import pyplot as plt
def Lab9a():
    # x-axis values
    x = [5, 2, 9, 4, 7]
    # Y-axis values
    y = [10, 5, 8, 4, 2]
    # Create a new figure
    plt.figure(1)
    # Function to plot the bar
    plt.bar(x,y)
    plt.title("Bar graph demonstration")
    plt.show()
```

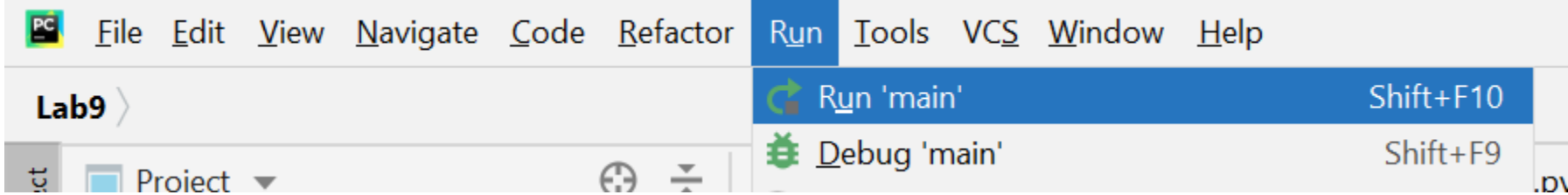
Lab9b.py

```
# importing matplotlib module
from matplotlib import pyplot as plt
def Lab9b():
    # x-axis values
    x = [5, 2, 9, 4, 7]
    # Y-axis values
    y = [10, 5, 8, 4, 2]
    # Create a new figure
    plt.figure(1)
    # Function to plot the bar
    plt.bar(x,y)
    plt.title("Bar graph demonstration")
    plt.show()
```

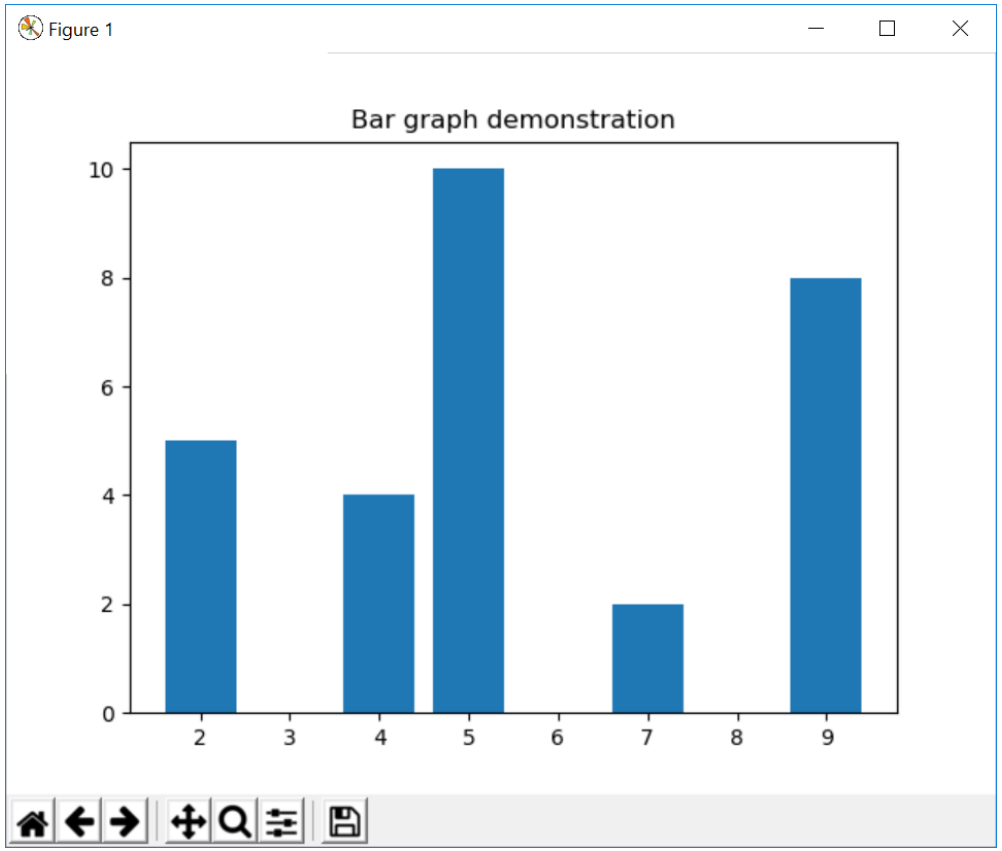
Lab9b.py

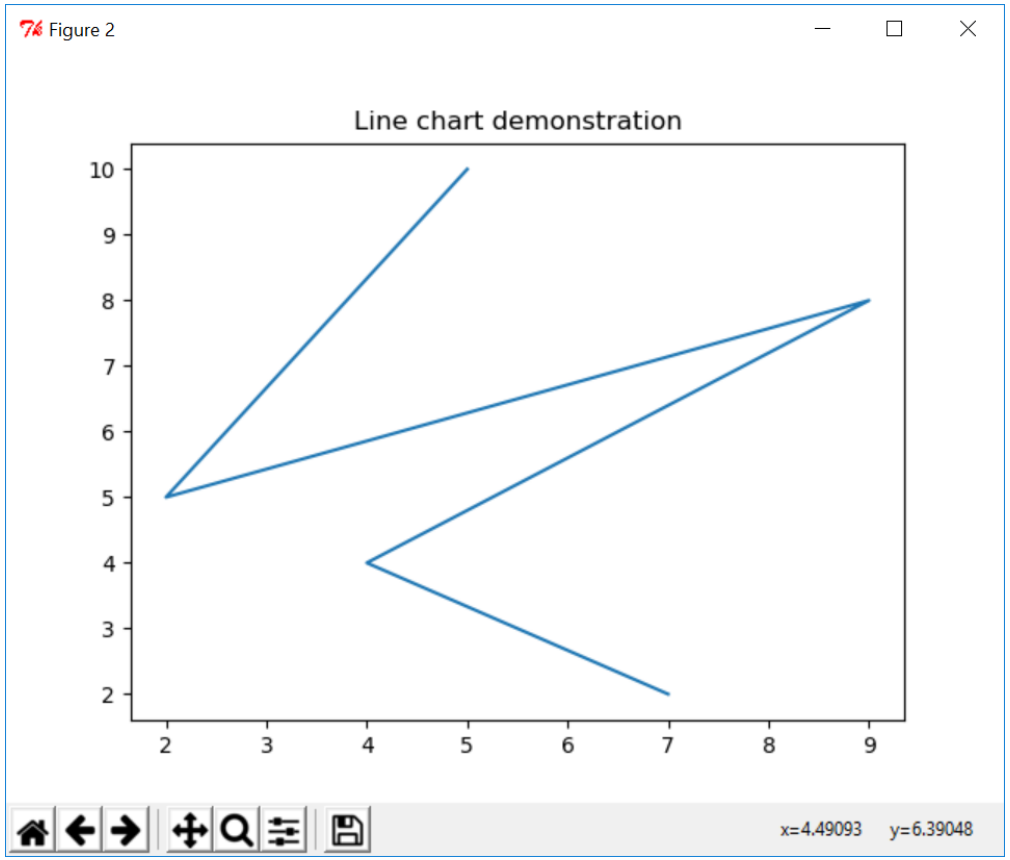
```
# importing matplotlib module
from matplotlib import pyplot as plt
def Lab9b():
    # x-axis values
    x = [5, 2, 9, 4, 7]
    # Y-axis values
    y = [10, 5, 8, 4, 2]
    # Create a new figure
    plt.figure(1)
    # Function to plot the bar
    plt.bar(x,y)
    plt.title("Bar graph demonstration")
    plt.figure(2)
    # Function to plot the line chart
    plt.plot(x,y)
    # function to show the plot
    plt.title("Line chart demonstration")
    plt.show()
```

main.py




```
from Lab9b import Lab9b
Lab9b()
```





How can we plot 2 charts in the same figure?

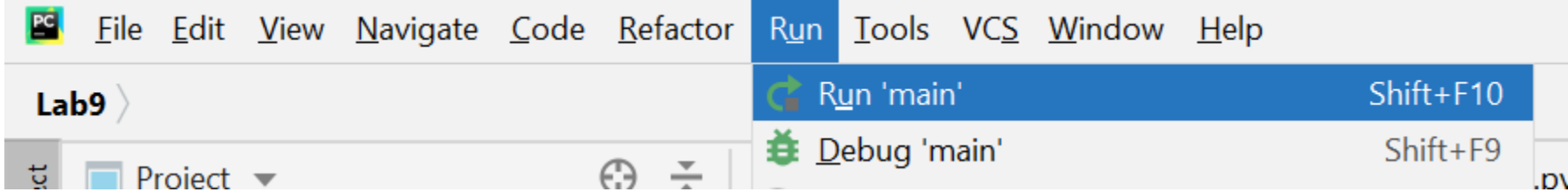
- We can use subplot function

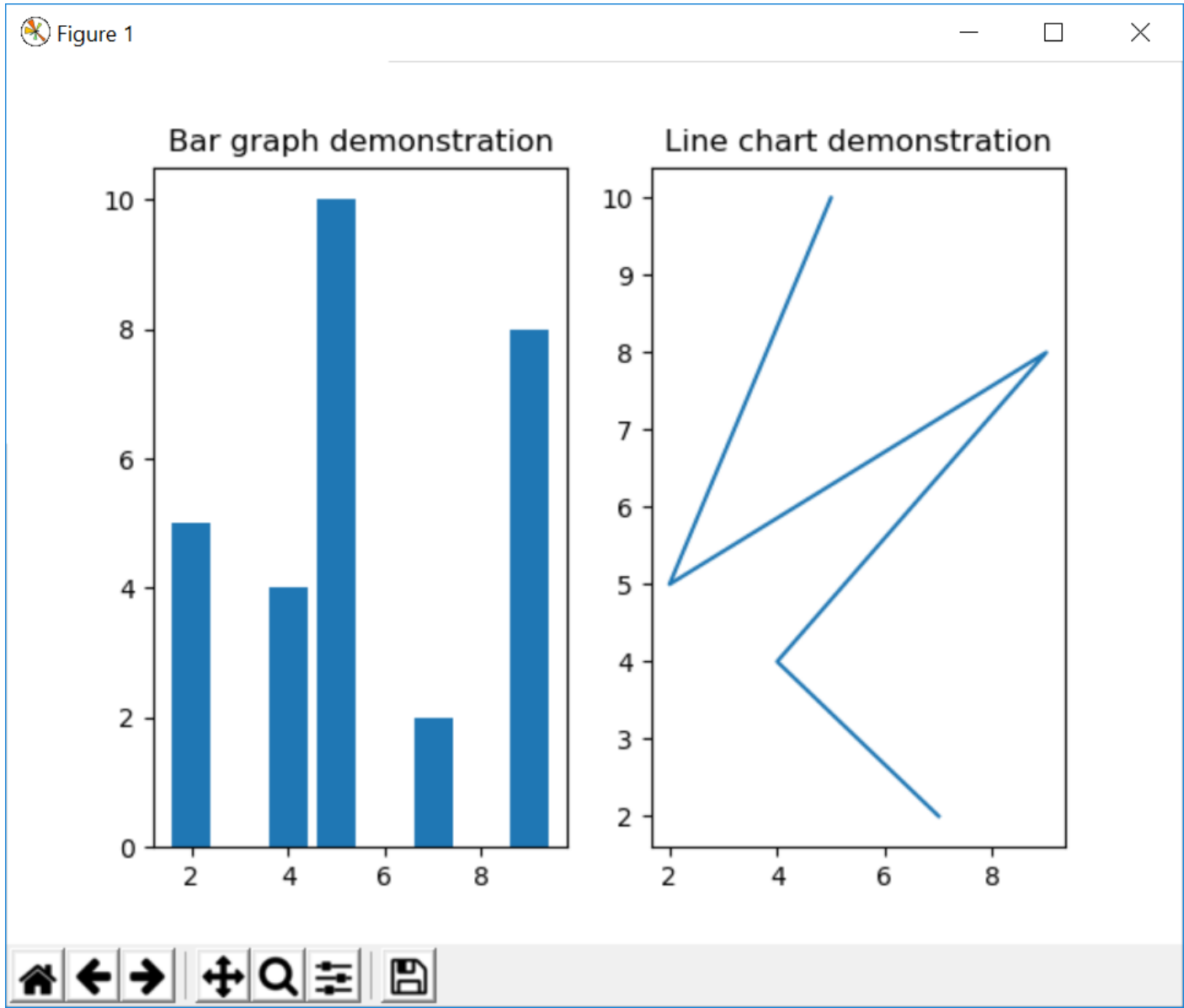
Lab9b.py

```
# importing matplotlib module
from matplotlib import pyplot as plt
def Lab9b():
    # x-axis values
    x = [5, 2, 9, 4, 7]
    # Y-axis values
    y = [10, 5, 8, 4, 2]
    # Create a new figure
    plt.subplot(1, 2, 1)
    # Function to plot the bar
    plt.bar(x,y)
    plt.title("Bar graph demonstration")
    plt.subplot(1, 2, 2)
    # Function to plot the line chart
    plt.plot(x,y)
    # function to show the plot
    plt.title("Line chart demonstration")
    plt.show()
```

main.py

```
from Lab9b import Lab9b
Lab9b()
```





Create Python file: Lab9c.py

File Edit View Navigate Code Refactor Run Tools VCS Window Help Lab9 - main.py

Lab9

main

Project

main.py Lab9d.py tk.tcl init.tcl

New
Cut Ctrl+X
Copy
Paste Ctrl+V
Find Usages Alt+F7
Find in Path... Ctrl+Shift+F
Replace in Path... Ctrl+Shift+R
Inspect Code...
Refactor
Clean Python Compiled Files
Add to Favorites
Reformat Code Ctrl+Alt+L
Optimize Imports Ctrl+Alt+O
Show in Explorer
Directory Path Ctrl+Alt+F12
Open in Terminal
Local History
Reload from Disk
Compare With... Ctrl+D
Mark Directory as
Remove BOM
Create Gist...

File
New Scratch File Ctrl+Alt+Shift+Insert
Directory
Python Package
Python File
HTML File
EditorConfig File
Resource Bundle

```
# Use a breakpoint in the code line below to debug your script.
print("Hi, {0}".format(name))  # Press Ctrl+F8 to toggle the breakpoint.


# Press the green button in the gutter to run the script.
if __name__ == '__main__':
```

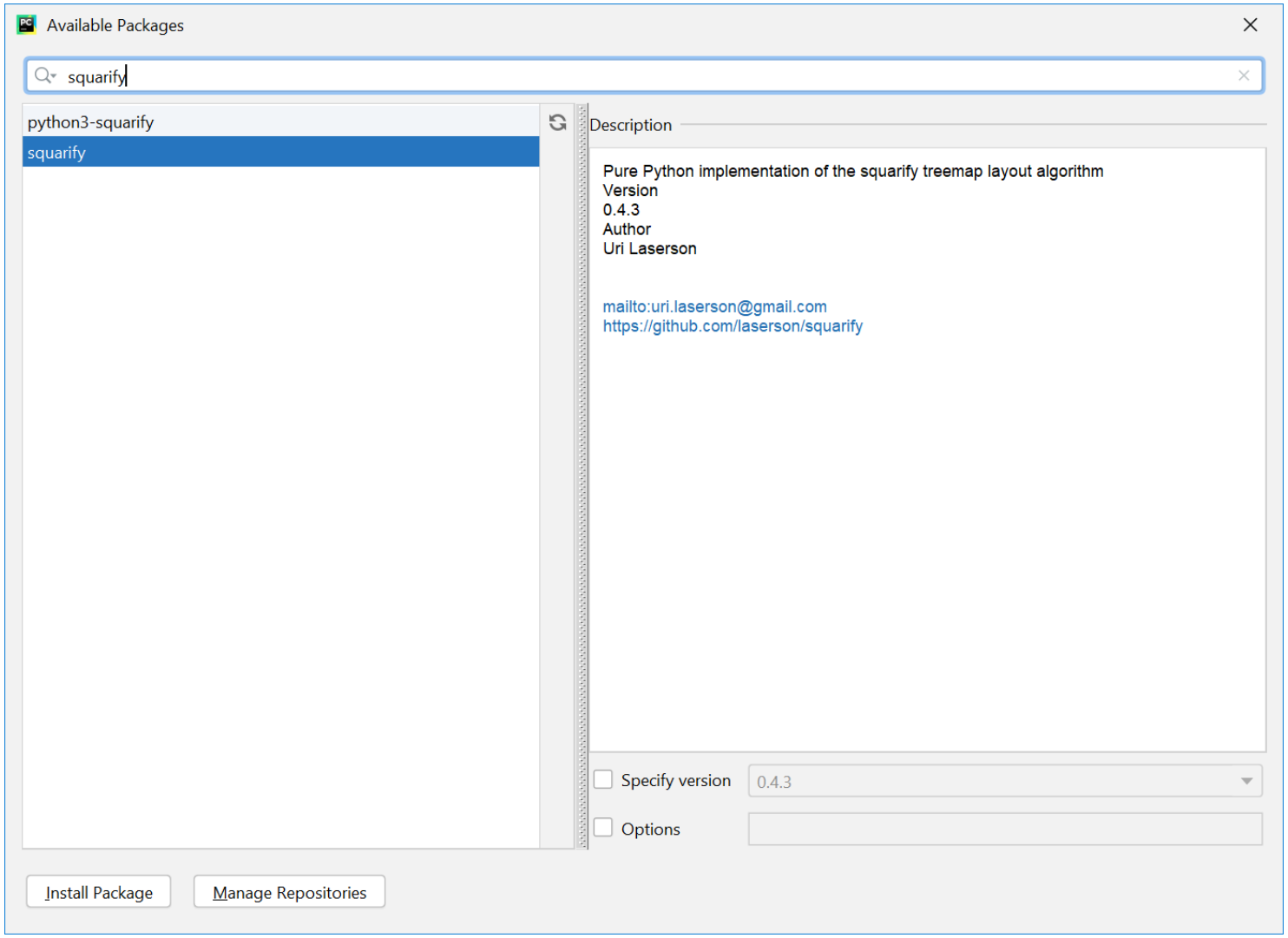
Run:

```
cts\Lab9\venv\Scripts\python.exe C:/Users/vanta/PycharmProjects/Lab9/main.py

code 0
```

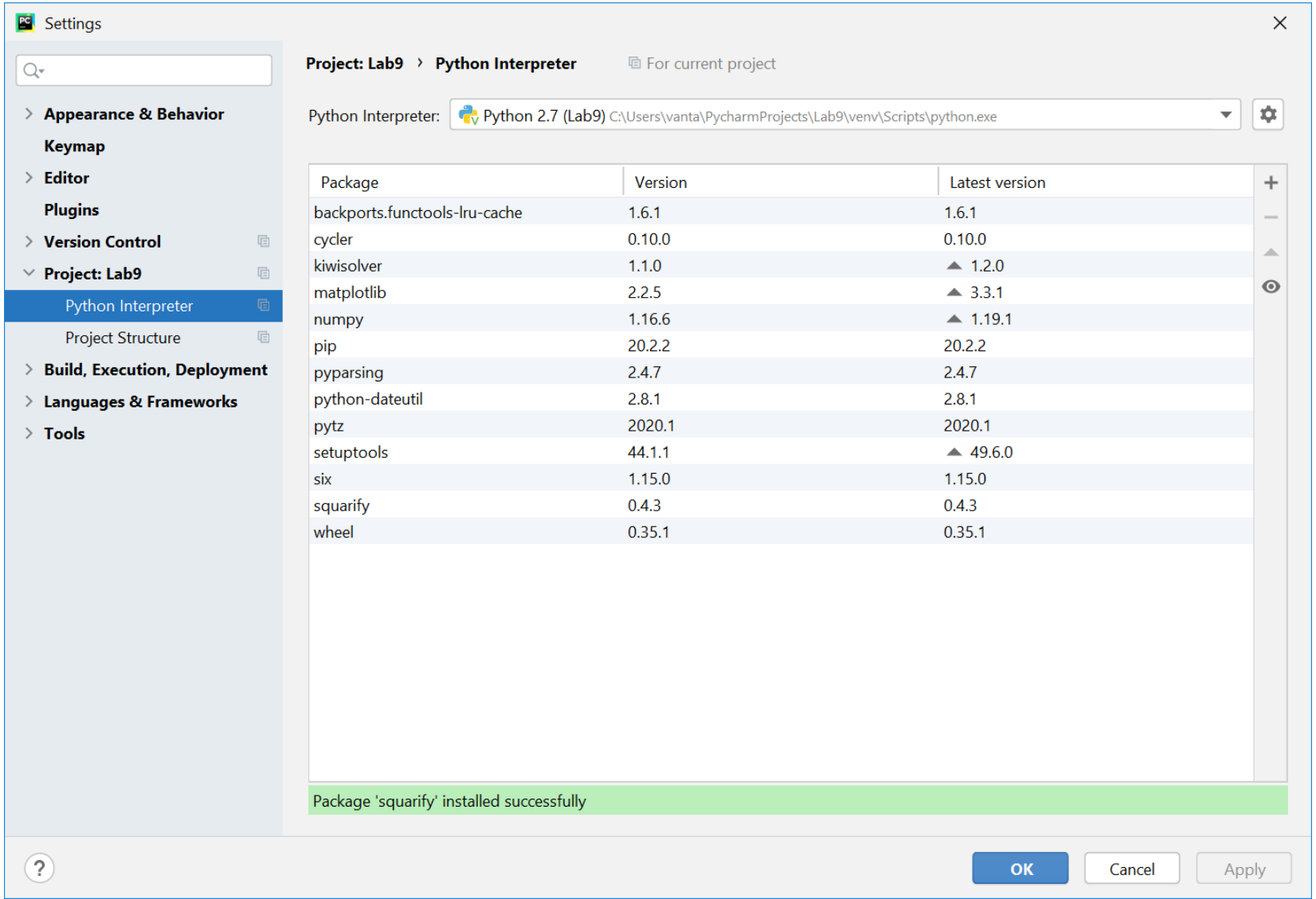
4: Run TODO 6: Problems 5: Debug Terminal Python Console Event Log

Creates a Python file from the specified template 19:1 CRLF UTF-8 4 spaces Python 2.7 (Lab9)

Install squarify



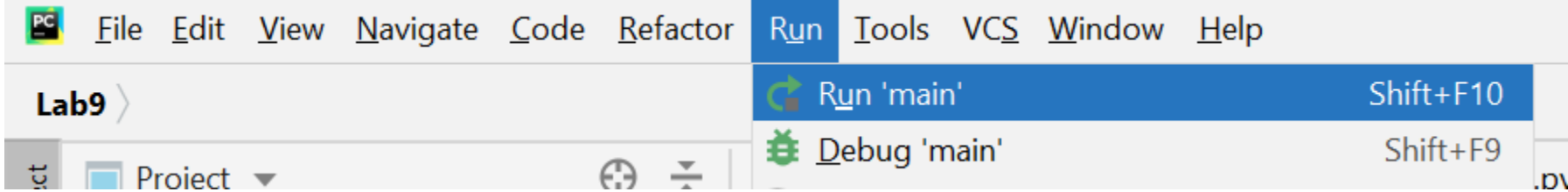
Installation finished



Lab9c.py

```
import matplotlib.pyplot as plt
import squarify

def Lab9c():
    sizes=[663267.26, 268580.82, 163695.57, 147042.40, 121589.48,
113998.30, 110560.71, 104093.57, 98380.64, 97813.56]
    label=["Alaska","Texas","California","Montana","New
Mexico","Arizona","Nevada","Colorado","Oregon","Wyoming"]
    squarify.plot(sizes=sizes, label=label, alpha=0.6 )
    plt.axis('off')
    plt.show()
```

main.py




```
from Lab9c import Lab9c
Lab9c()
```

Lab9c.py

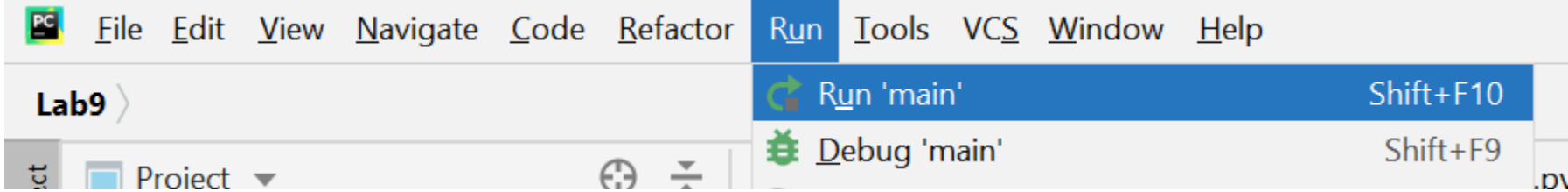
```
import matplotlib.pyplot as plt
import squarify

def Lab9c():
    sizes=[663267.26, 268580.82, 163695.57, 147042.40, 121589.48,
113998.30, 110560.71, 104093.57, 98380.64, 97813.56]
    label=["Alaska","Texas","California","Montana","New
Mexico","Arizona","Nevada","Colorado","Oregon","Wyoming"]
    color=['red','blue','green','grey','yellow', 'orange', 'brown', 'cyan', 'purple',
'olive']
    squarify.plot(sizes=sizes, label=label, color=color, alpha=0.6 )
    plt.axis('off')
    plt.show()
```

Supported Colors

black	bisque	forestgreen	slategrey
dimgray	darkorange	limegreen	lightsteelblue
dimgrey	burlywood	darkgreen	cornflowerblue
gray	antiquewhite	green	royalblue
grey	tan	lime	ghostwhite
darkgray	navajowhite	seagreen	lavender
darkgrey	blanchedalmond	mediumseagreen	midnightblue
silver	papayawhip	springgreen	navy
lightgray	moccasin	mintcream	darkblue
lightgrey	orange	mediumspringgreen	mediumblue
gainsboro	wheat	mediumaquamarine	blue
whitesmoke	oldlace	aquamarine	slateblue
white	floralwhite	turquoise	darkslateblue
snow	darkgoldenrod	lightseagreen	mediumslateblue
rosybrown	goldenrod	mediumturquoise	mediumpurple
lightcoral	cornsilk	azure	rebeccapurple
indianred	gold	lightcyan	blueviolet
brown	lemonchiffon	paleturquoise	indigo
firebrick	khaki	darkslategray	darkorchid
maroon	palegoldenrod	darkslategrey	darkviolet
darkred	darkkhaki	teal	mediumorchid
red	ivory	darkcyan	thistle
mistyrose	beige	aqua	plum
salmon	lightyellow	cyan	violet
tomato	lightgoldenrodyellow	darkturquoise	purple
darksalmon	olive	cadetblue	darkmagenta
coral	yellow	powderblue	fuchsia
orangered	olivedrab	lightblue	magenta
lightsalmon	yellowgreen	deepskyblue	orchid
sienna	darkolivegreen	skyblue	mediumvioletred
seashell	greenyellow	lightskyblue	deeppink
chocolate	chartreuse	steelblue	hotpink
saddlebrown	lawngreen	aliceblue	lavenderblush
sandybrown	honeydew	dodgerblue	palevioletred
peachpuff	darkseagreen	lightslategray	crimson
peru	palegreen	lightslategrey	pink
linen	lightgreen	slategray	lightpink

main.py

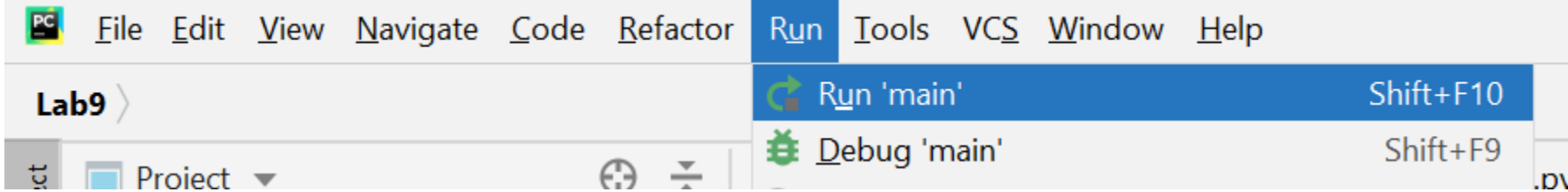
```
from Lab9c import Lab9c
Lab9c()
```



Lab9c.py

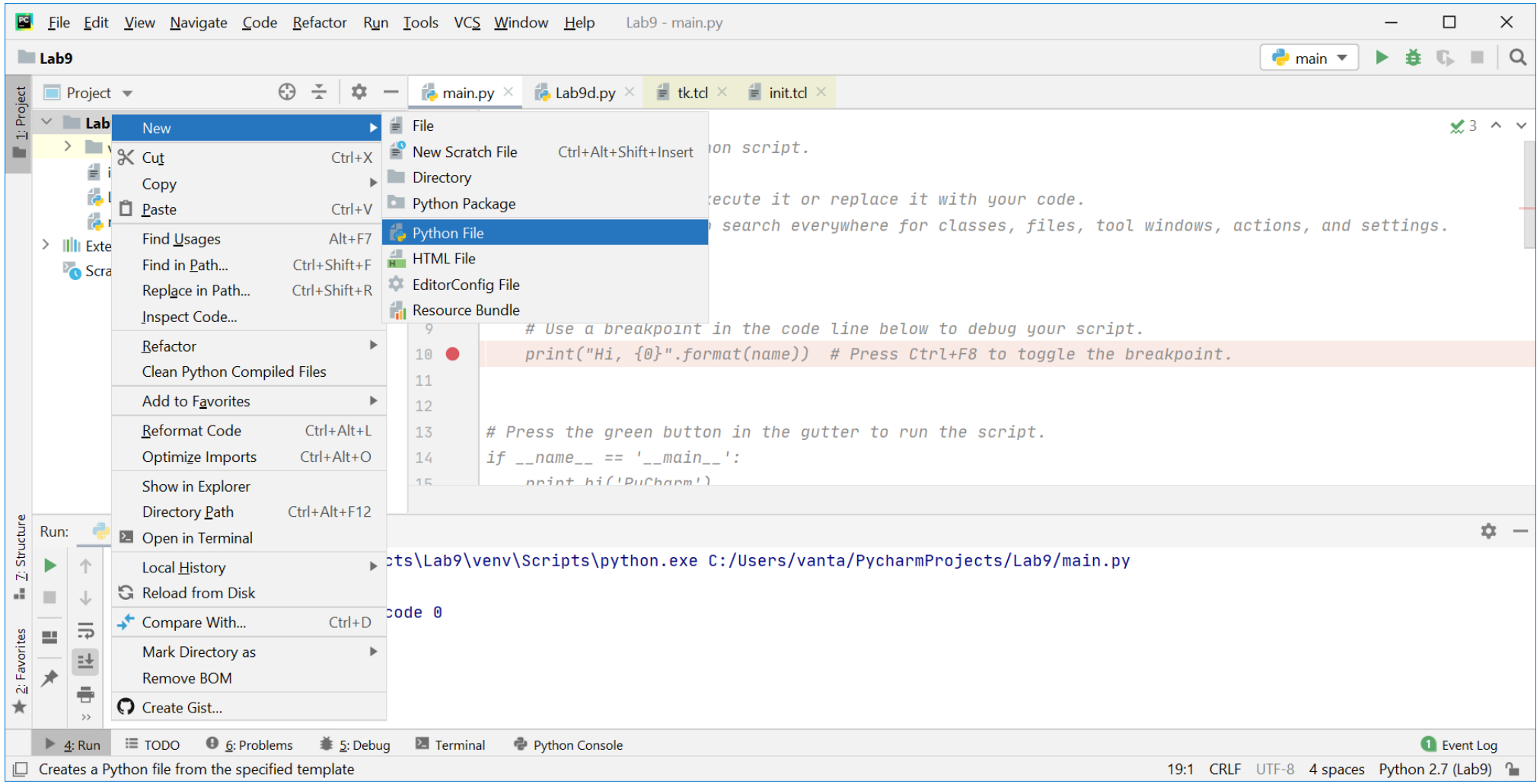
```
import matplotlib.pyplot as plt
import squarify

def Lab9c():
    sizes=[663267.26, 268580.82, 163695.57, 147042.40, 121589.48,
113998.30, 110560.71, 104093.57, 98380.64, 97813.56]
    label=["Alaska","Texas","California","Montana","New
Mexico","Arizona","Nevada","Colorado","Oregon","Wyoming"]
    color=['red','blue','green','grey','yellow', 'orange', 'brown', 'cyan', 'purple',
'olive']
    squarify.plot(sizes=sizes, label=label, color=color, alpha=0.6 )
    plt.axis('off')
    plt.title("Treemap of 10 largest US States")
    plt.show()
```

main.py

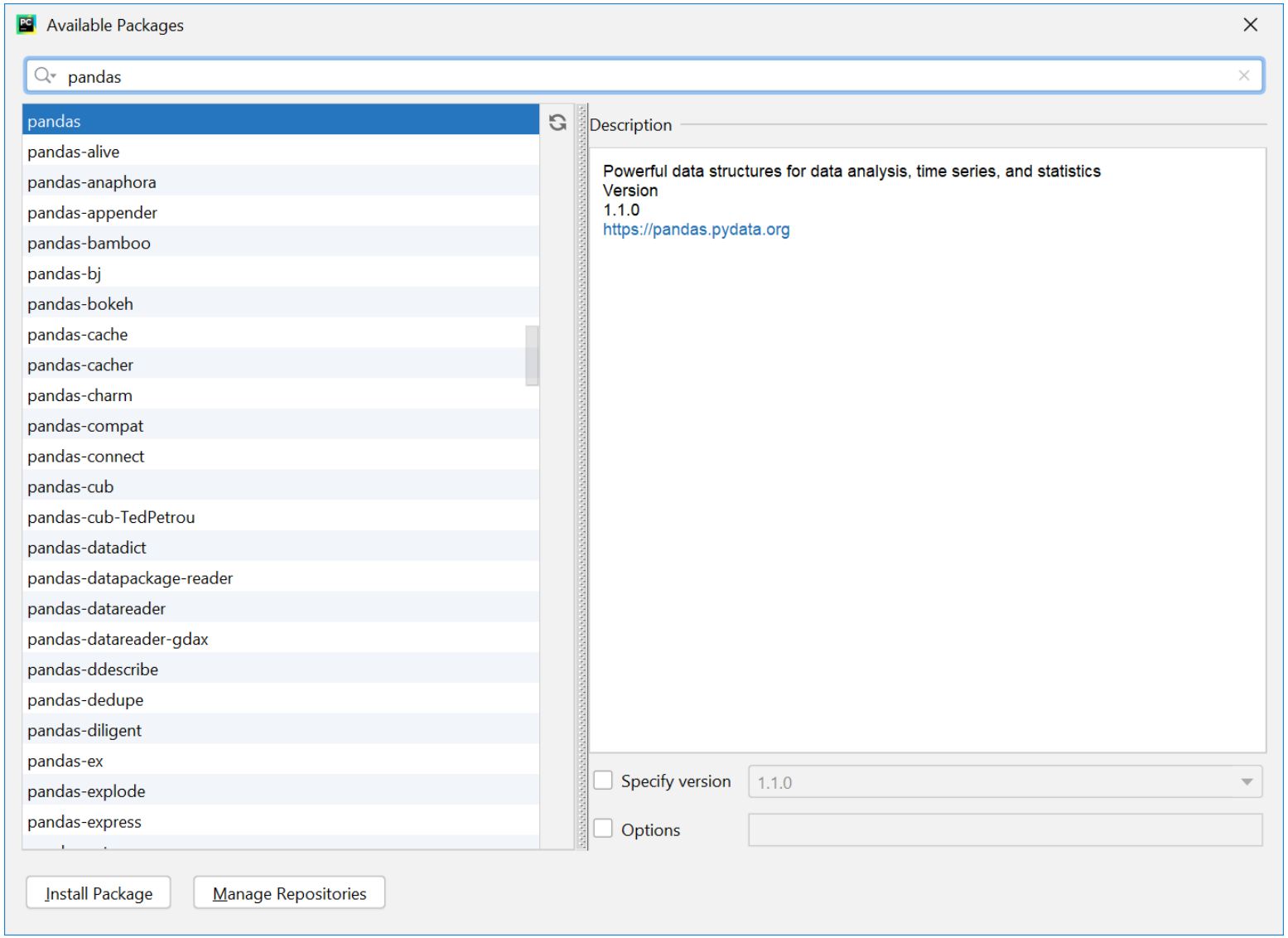



```
from Lab9c import Lab9c
Lab9c()
```

Create Python file: Lab9d.py



Install pandas



Installation finished

Settings

Appearance & Behavior
Keymap
Editor
Plugins
Version Control
Project: Lab9
Python Interpreter
Project Structure
Build, Execution, Deployment
Languages & Frameworks
Tools

Project: Lab9 › Python Interpreter For current project

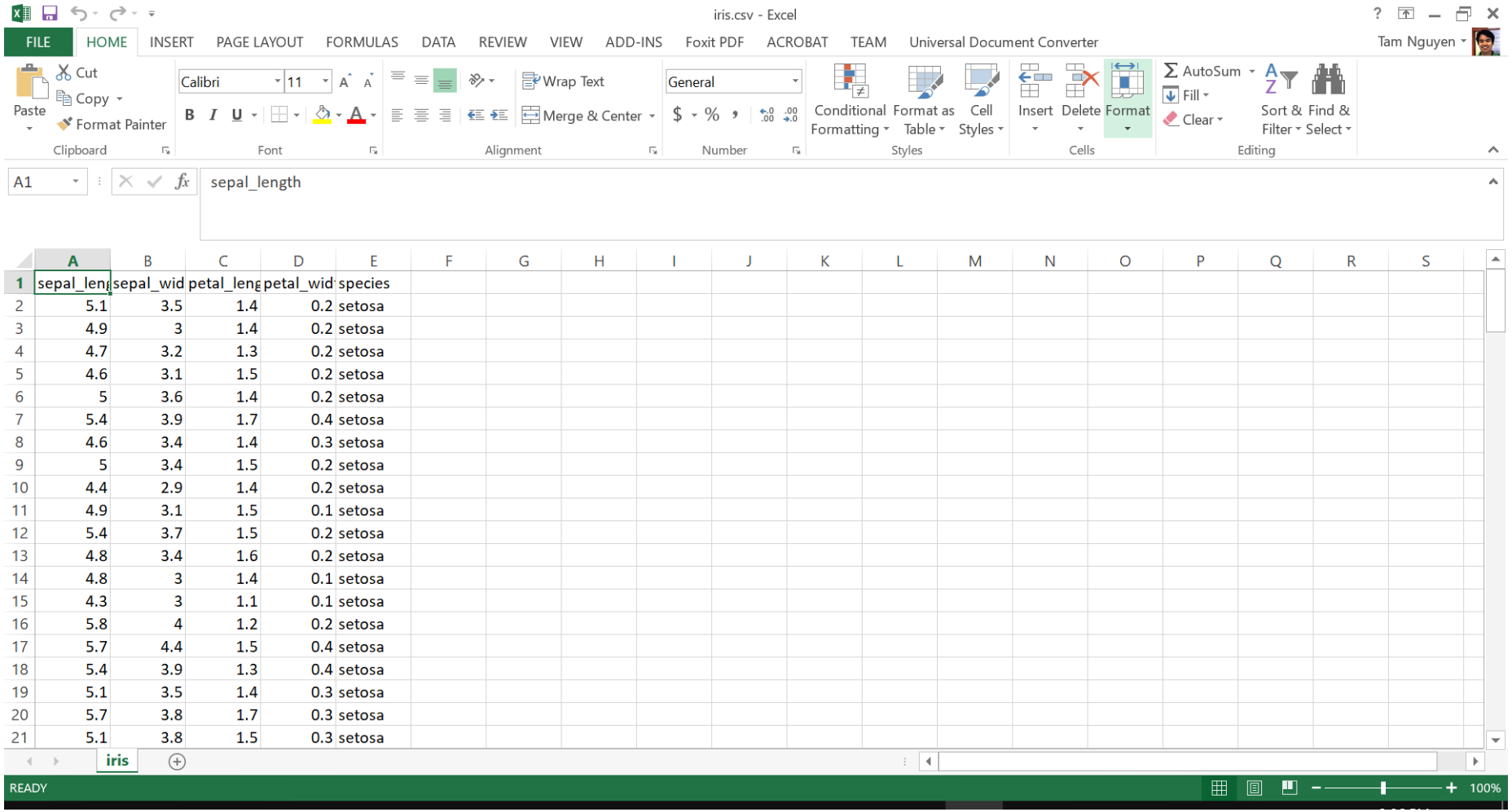
Python Interpreter: Python 2.7 (Lab9) C:\Users\vanta\PycharmProjects\Lab9\venv\Scripts\python.exe

Package	Version	Latest version
backports.functools-lru-cache	1.6.1	1.6.1
cycler	0.10.0	0.10.0
kiwisolver	1.1.0	▲ 1.2.0
matplotlib	2.2.5	▲ 3.3.1
numpy	1.16.6	▲ 1.19.1
pandas	0.24.2	▲ 1.1.0
pip	20.2.2	20.2.2
pyparsing	2.4.7	2.4.7
python-dateutil	2.8.1	2.8.1
pytz	2020.1	2020.1
setuptools	44.1.1	▲ 49.6.0
six	1.15.0	1.15.0
squarify	0.4.3	0.4.3
wheel	0.35.1	0.35.1

Package 'pandas' installed successfully

OK Cancel Apply

Download iris.csv from isidore



	A	B	C	D	E
1	sepal_leng	sepal_wid	petal_leng	petal_wid	species
2	5.1	3.5	1.4	0.2	setosa
3	4.9	3	1.4	0.2	setosa
4	4.7	3.2	1.3	0.2	setosa
5	4.6	3.1	1.5	0.2	setosa
6	5	3.6	1.4	0.2	setosa
7	5.4	3.9	1.7	0.4	setosa
8	4.6	3.4	1.4	0.3	setosa
9	5	3.4	1.5	0.2	setosa
10	4.4	2.9	1.4	0.2	setosa
11	4.9	3.1	1.5	0.1	setosa
12	5.4	3.7	1.5	0.2	setosa
13	4.8	3.4	1.6	0.2	setosa
14	4.8	3	1.4	0.1	setosa
15	4.3	3	1.1	0.1	setosa
16	5.8	4	1.2	0.2	setosa
17	5.7	4.4	1.5	0.4	setosa
18	5.4	3.9	1.3	0.4	setosa
19	5.1	3.5	1.4	0.3	setosa
20	5.7	3.8	1.7	0.3	setosa
21	5.1	3.8	1.5	0.3	setosa

Lab9d.py

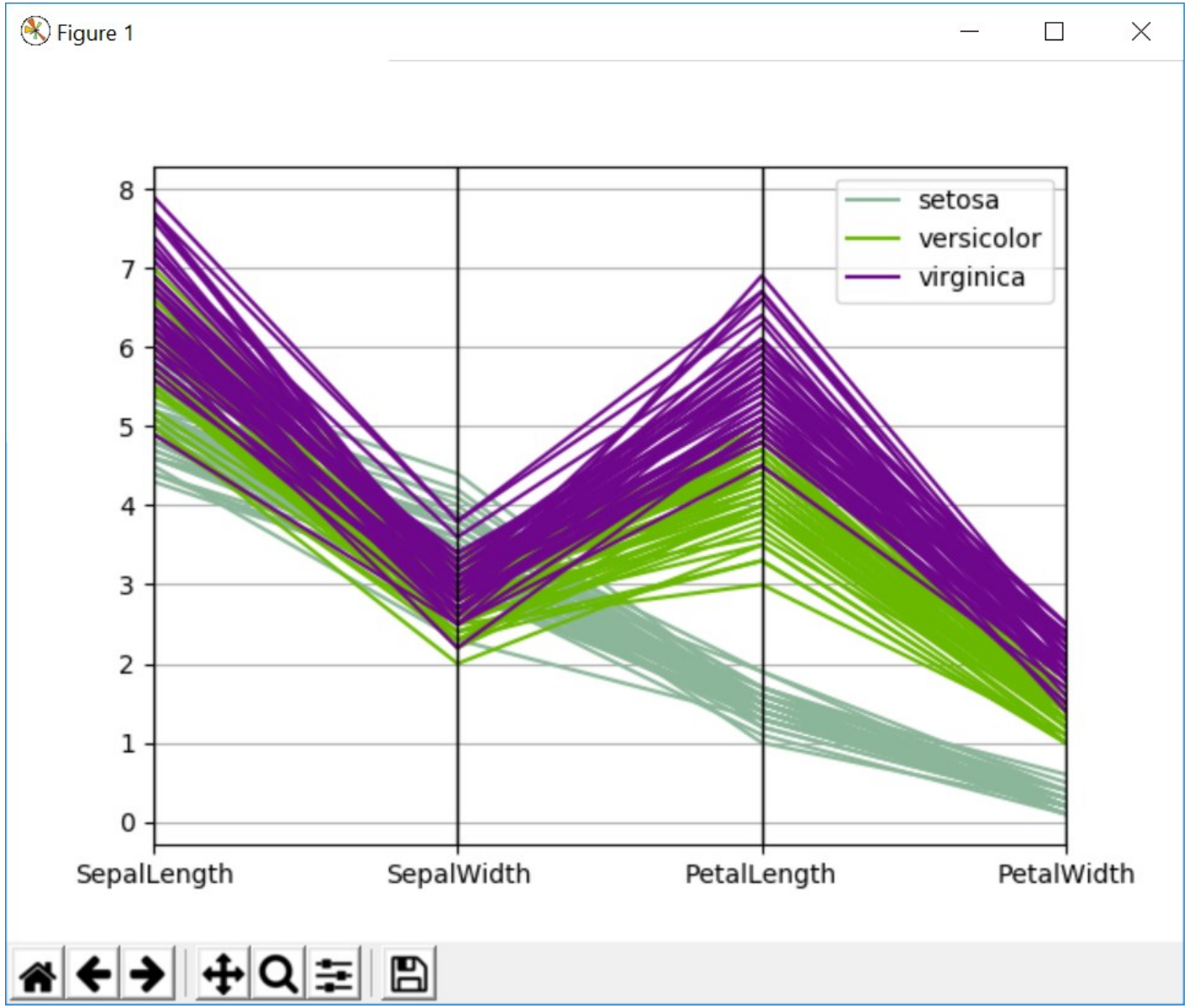
```
import pandas
import matplotlib.pyplot as plt
from pandas.plotting import parallel_coordinates

def Lab9d():
    data = pandas.read_csv('iris.csv', sep=',')
```

Lab9d.py

```
import pandas
import matplotlib.pyplot as plt
from pandas.plotting import parallel_coordinates

def Lab9d():
    data = pandas.read_csv('iris.csv', sep=',')
    parallel_coordinates(data, 'Name')
    plt.show()
```

main.py

```
from Lab9d import Lab9d
Lab9d()
```

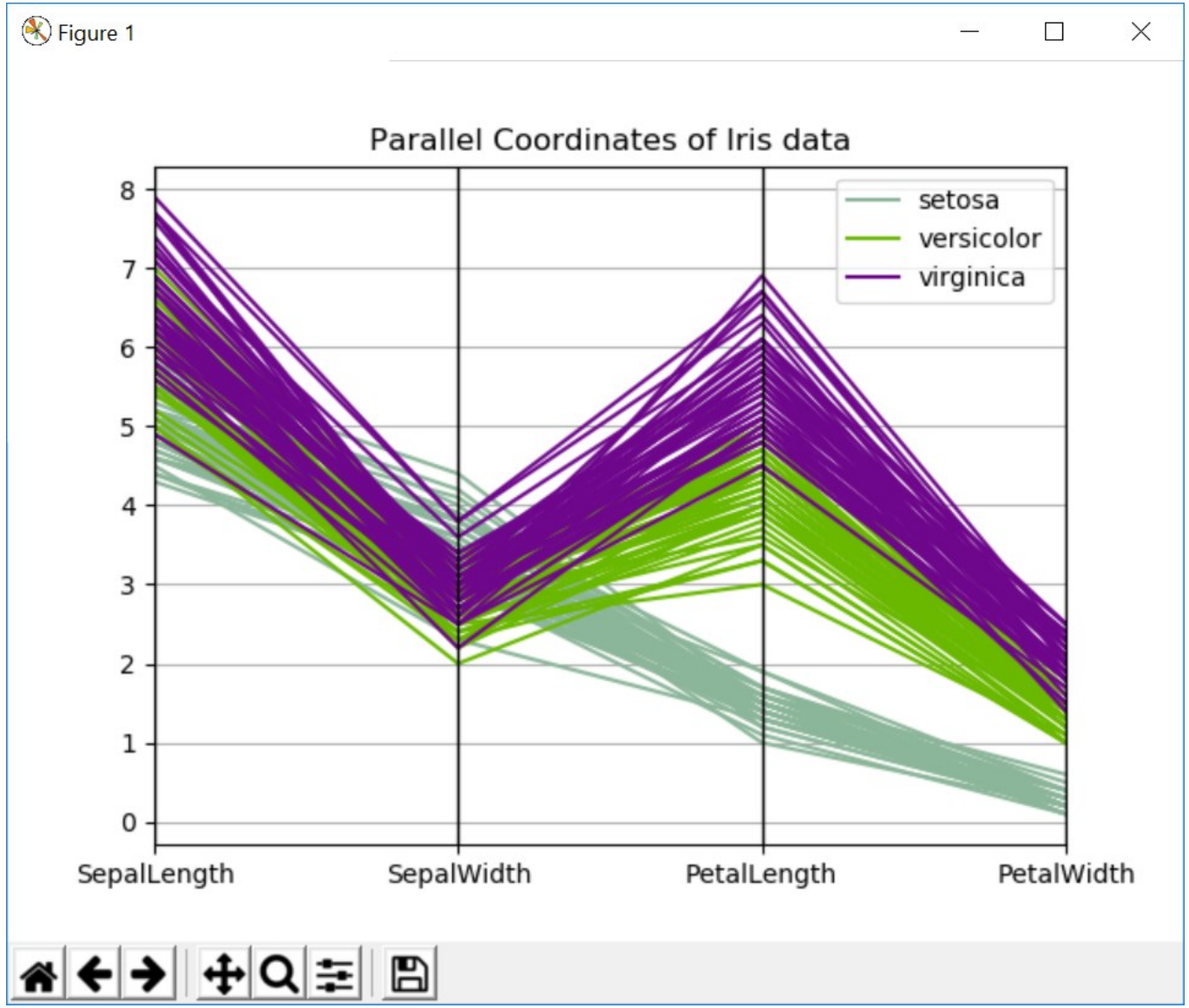


Lab9d.py

```
import pandas
import matplotlib.pyplot as plt
from pandas.plotting import parallel_coordinates

def Lab9d():
    data = pandas.read_csv('iris.csv', sep=',')
    parallel_coordinates(data, 'Name')
    plt.title("Parallel Coordinates of Iris data")
    plt.show()
```

main.py

```
from Lab9d import Lab9d
Lab9d()
```

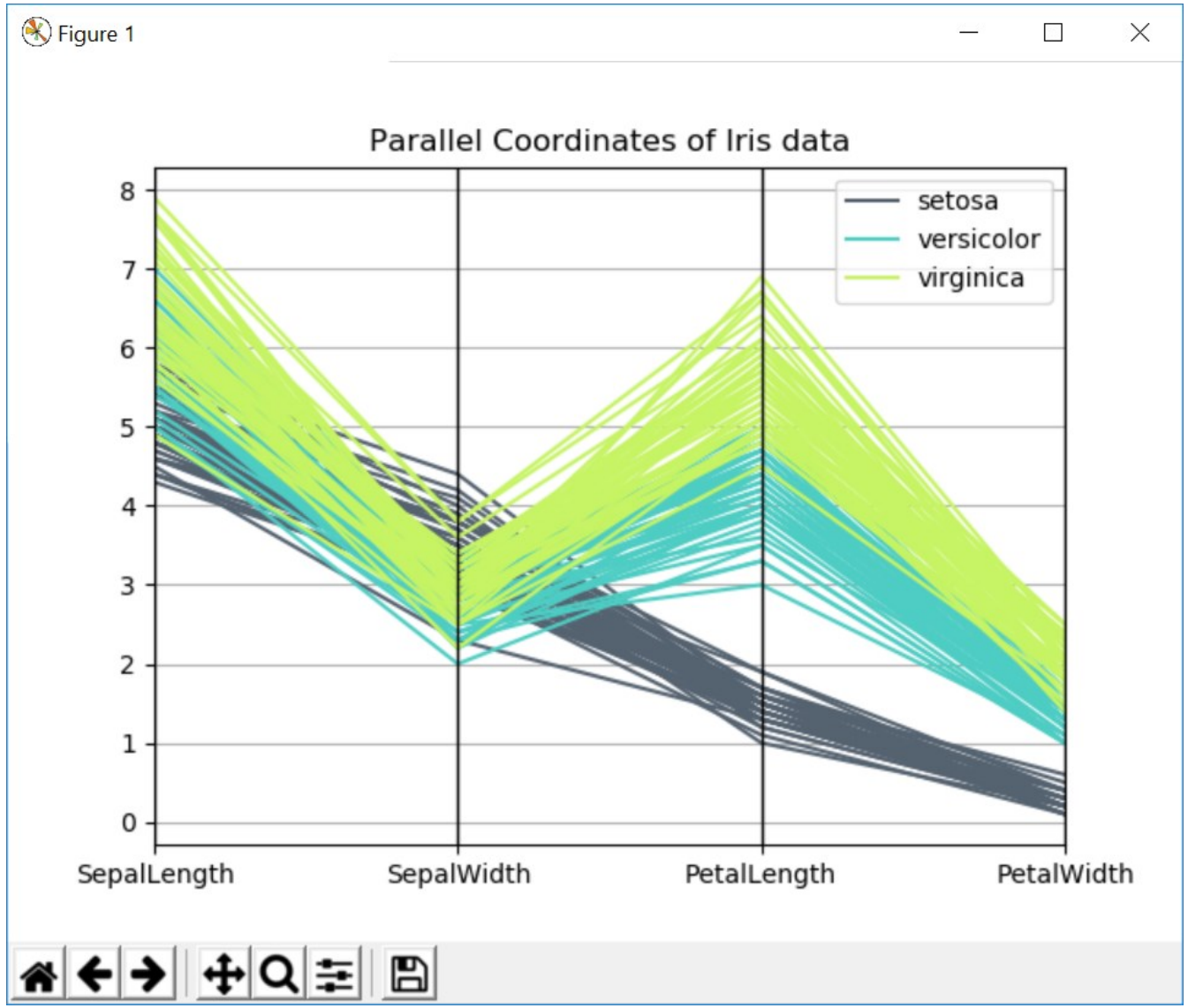


Lab9d.py

```
import pandas
import matplotlib.pyplot as plt
from pandas.plotting import parallel_coordinates

def Lab9d():
    data = pandas.read_csv('iris.csv', sep=',')
    parallel_coordinates(data, 'Name', color=('#556270',
'#4ECDC4', '#C7F464'))
    plt.title("Parallel Coordinates of Iris data")
    plt.show()
```

main.py

```
from Lab9d import Lab9d
Lab9d()
```



Q&A