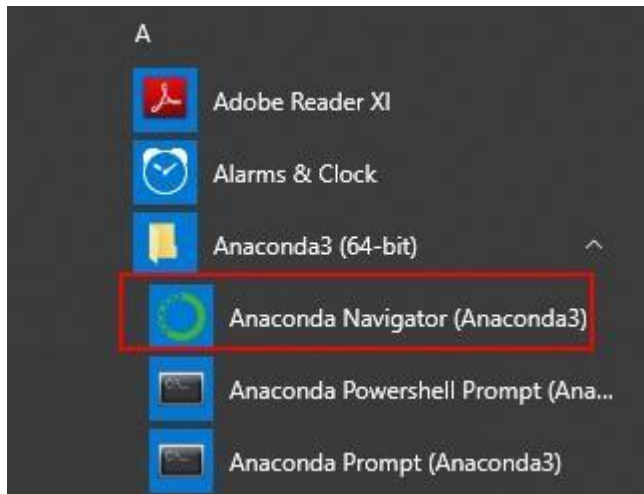


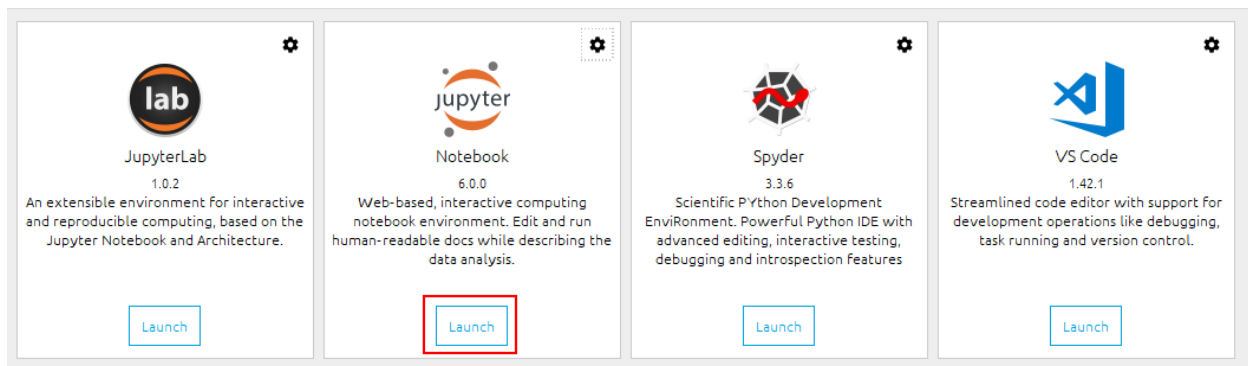
Data Science with Python Module 5: Hands-on: 1

Create a plot with a grid

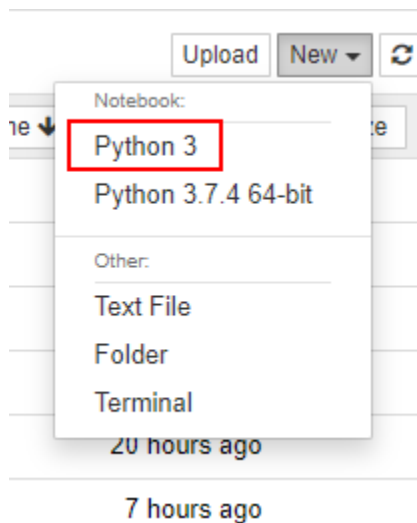
Step 1: Open Anaconda Navigator



Step 2: Click on Launch button under jupyter notebooks.



Step 3: After the notebook opens click on new and Python 3.



Step 4: Import numpy and matplotlib.pyplot by typing the following code in the notebook and run it by pressing shift + enter

```
In [1]: import numpy as np
import matplotlib.pyplot as plt
```

Step 5: Type the following code to create a plot with grids.

Step 5.1: Generate some data to plot.

```
# generate data
x = [_ for _ in range(0, 10)]
y = [_*_ for _ in range(0, 10)]

# create a figure
fig = plt.figure()

# get current axes
ax = fig.gca()

# set xticks and yticks
ax.set_xticks(np.arange(0, 10, 1))
ax.set_yticks(np.arange(0, 100, 10))

# create a scatter plot
plt.scatter(x, y)

# create a grid
plt.grid()

# show the grid
plt.show()
```

Step 5.2: Create a figure to plot data on.

```
# generate data
x = [_ for _ in range(0, 10)]
y = [* _ for _ in range(0, 10)]

# create a figure
fig = plt.figure()

# get current axes
ax = fig.gca()

# set xticks and yticks
ax.set_xticks(np.arange(0, 10, 1))
ax.set_yticks(np.arange(0, 100, 10))

# create a scatter plot
plt.scatter(x, y)

# create a grid
plt.grid()

# show the grid
plt.show()
```

Step 5.3: Get the current axes of the figure.

```
# generate data
x = [_ for _ in range(0, 10)]
y = [* _ for _ in range(0, 10)]

# create a figure
fig = plt.figure()

# get current axes
ax = fig.gca()

# set xticks and yticks
ax.set_xticks(np.arange(0, 10, 1))
ax.set_yticks(np.arange(0, 100, 10))

# create a scatter plot
plt.scatter(x, y)

# create a grid
plt.grid()

# show the grid
plt.show()
```

Step 5.4: Set xticks and yticks on the axes.

```
# generate data
x = [_ for _ in range(0, 10)]
y = [* _ for _ in range(0, 10)]

# create a figure
fig = plt.figure()

# get current axes
ax = fig.gca()

# set xticks and yticks
ax.set_xticks(np.arange(0, 10, 1))
ax.set_yticks(np.arange(0, 100, 10))

# create a scatter plot
plt.scatter(x, y)

# create a grid
plt.grid()

# show the grid
plt.show()
```

Step 5.5: Generate a scatter plot based on the data generated on previous step.

```
# generate data
x = [_ for _ in range(0, 10)]
y = [* _ for _ in range(0, 10)]

# create a figure
fig = plt.figure()

# get current axes
ax = fig.gca()

# set xticks and yticks
ax.set_xticks(np.arange(0, 10, 1))
ax.set_yticks(np.arange(0, 100, 10))

# create a scatter plot
plt.scatter(x, y)

# create a grid
plt.grid()

# show the grid
plt.show()
```

Step 5.6: Create a grid.

```
# generate data
x = [_ for _ in range(0, 10)]
y = [_*_ for _ in range(0, 10)]

# create a figure
fig = plt.figure()

# get current axes
ax = fig.gca()

# set xticks and yticks
ax.set_xticks(np.arange(0, 10, 1))
ax.set_yticks(np.arange(0, 100, 10))

# create a scatter plot
plt.scatter(x, y)

# create a grid
plt.grid()

# show the grid
plt.show()
```

Step 5.7: Show the plot.

```
# generate data
x = [_ for _ in range(0, 10)]
y = [_*_ for _ in range(0, 10)]

# create a figure
fig = plt.figure()

# get current axes
ax = fig.gca()

# set xticks and yticks
ax.set_xticks(np.arange(0, 10, 1))
ax.set_yticks(np.arange(0, 100, 10))

# create a scatter plot
plt.scatter(x, y)

# create a grid
plt.grid()

# show the grid
plt.show()
```

Step 6: Run the code by pressing Shift + Enter and observe the output.

