[Video Notes](https://zoom.us/rec/play/XcjsP6M475BoHYRa2vERdDvbmfwOQcYFNdaWBfvJiv-7TjYHsOjKAK7MQ0sNLEZdSeeQCcGLDk-HP8-C.mdXUslbnjdt5Yuph)

<https://docs.google.com/document/d/1Gd4lz3AtldzVjR0b-gTMry7dAF-j7Rvkn17n1OQFaoI/edit?usp=sharing>

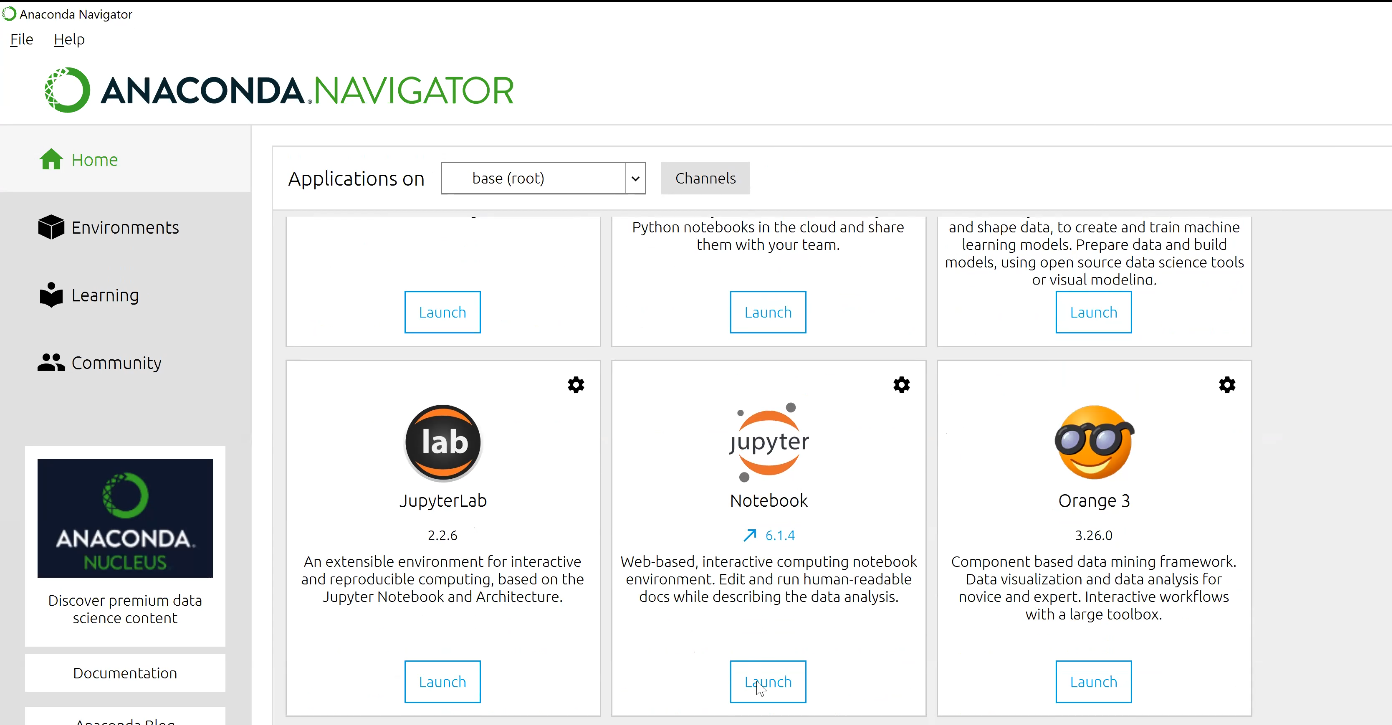
[00:03:00]

Data Analysis in Python

* Install Anaconda from <https://www.anaconda.com/products/individual>

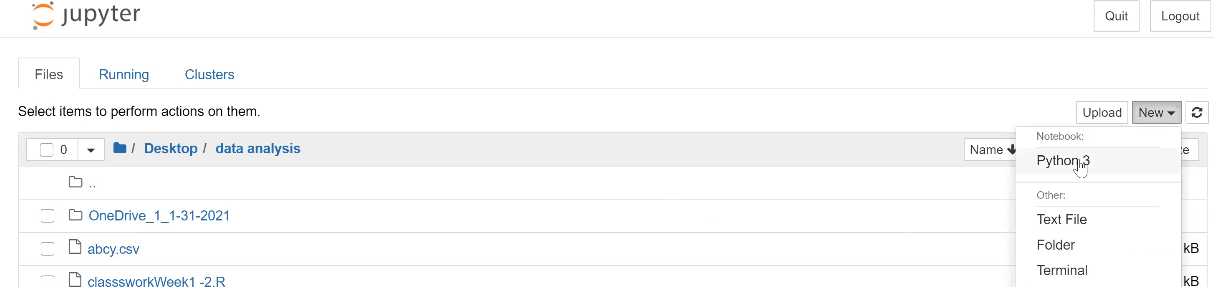
[00:17:33]

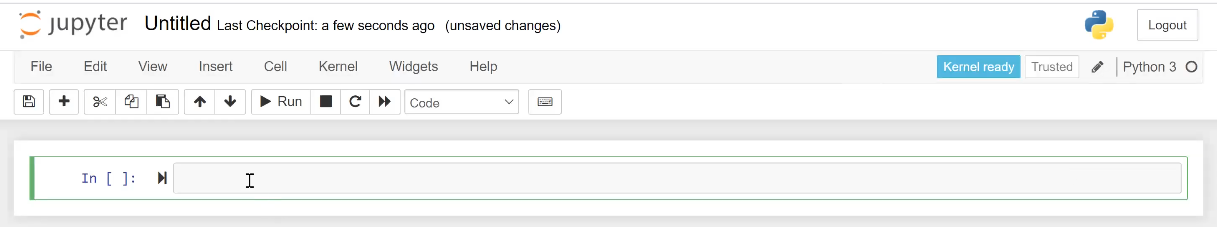
* Open Anaconda Navigator and launch Jupyter Notebook:



[00:18:39]

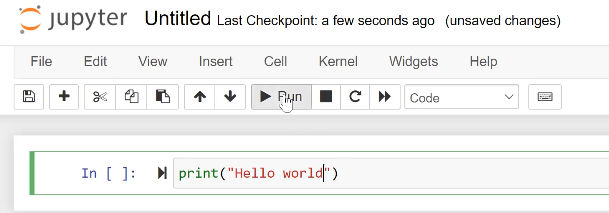
* The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations, and narrative text.
* Now create a python file in a folder by going to New -> Python3

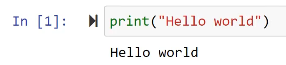




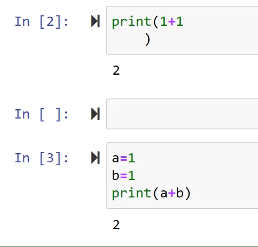
[00:19:41]

* Here we will print hello world.
* Click the Run button to execute the code.





* Use the Insert button from the top menu to insert a box.



[00:21:20]

**Activity**:

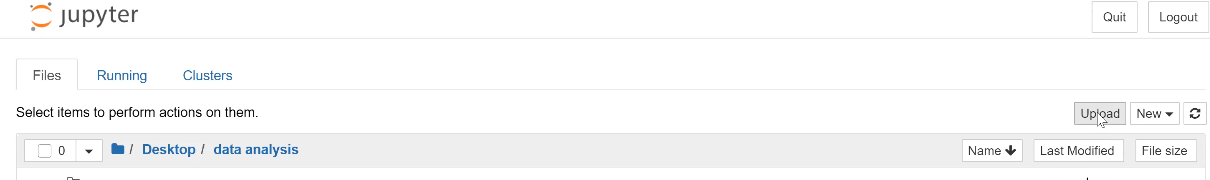
* In Python get an array of all numbers from 1 to 10
* Take that array and apply the function (2 \* x + 1) to each element
* Sum all the elements in the array

[00:37:45]

|  |  |
| --- | --- |
| Group | Code |
| 1 | array = [1,2,3,4,5,6,7,8,9,10]  sum = 0  for i in array:  sum += 2 \* i + 1  print(sum)  120 |
| 2 | numArray = list(range(1,11))  total = 0  for num in range(len(numArray)):  total += numArray[num] \* 2 + 1    print(total) |
| 3 | arr = [ i+1 for i in range(10) ]  arr = [ 2\*x+1 for x in arr ]  total = sum(arr)  print(total) |
| 4 |  |
| 5 | myarray = [1,2,3,4,5,6,7,8,9,10]  total = 0  for x in myarray:  x = (2\*x)+1  total += x  print(myarray)  print(total) |
| 6 | a\_list = list(range(1,11))  b\_list = [2 \* x + 1 for x in a\_list]  total = sum(b\_list)  print(total) |
| 7 | array = [1,2,3,4,5,6,7,8,9,10]  total = 0  for x in array:  total += (2 \* x + 1)  print(total) |

[00:41:18]

* Here we will upload the World Indicators file in jupyter.
* To do that, use the Upload button:



[00:42:55]

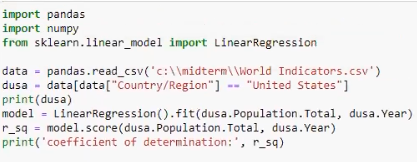
**Activity**:

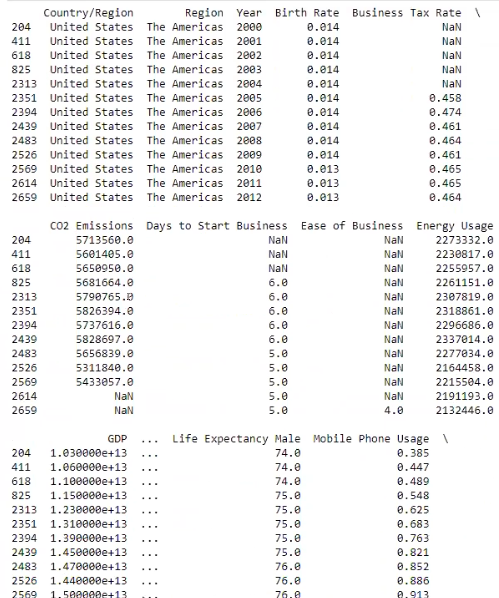
* With the help of Google, solve first part of problem 1 in the Midterm
* Read the World Indicators File.
* Filter by United States
* Create a Linear Model of Population by Year
* Predict Population in 2015

*Paste your code on the table. Your numbers should be exactly the same as the ones in the midterm.*

|  |  |
| --- | --- |
| Group | Code |
| 1 | import pandas  import numpy  from sklearn.linear\_model import LinearRegression  data = pandas.read\_csv('c:\\midterm\\World Indicators.csv')  dusa = data[data["Country/Region"] == "United States"]  y = list(dusa["Population Total"])  x = numpy.array(dusa["Year"]).reshape(-1, 1)  model = LinearRegression().fit(x, y)  print(model.predict(numpy.array([[2015]]))) |
| 2 | import numpy as np  import pandas as pd  from sklearn.linear\_model import LinearRegression  data = pd.read\_csv("WorldIndicators.csv")  newdata = data[data['Country/Region'] == "United States"]  model = np.polyfit(newdata["Year"], newdata['Population Total'], 1)  predict = np.poly1d(model)  year = 2015  predict(year) |
| 3 | import pandas  import numpy  from sklearn.linear\_model import LinearRegression  data = pandas.read\_csv('c:\\midterm\\World Indicators.csv')  dusa = data[data["Country/Region"] == "United States"]  y = list(dusa["Population Total"])  x = numpy.array(dusa["Year"]).reshape(-1, 1)  model = LinearRegression().fit(x, y)  print(model.predict(numpy.array([[2015]]))) |
| 4 |  |
| 5 | import pandas as pd  import os  import numpy as np  import statsmodels.api as sm  Pwd  variable = pd.read\_csv(r"C:\DataAnalytics\World Indicators.csv")  variable.head()  is\_usa = variable['Country/Region']=='United States'  print(is\_usa.head())  var\_usa = variable[is\_usa]  print(var\_usa)  x = list(var\_usa["Year"])  x = sm.add\_constant(x)  y = list(var\_usa["Population Total"])  model = sm.OLS(y,x).fit()  predictions = model.predict([1,2015])  model.summary()  print(predictions) |

[01:21:40]



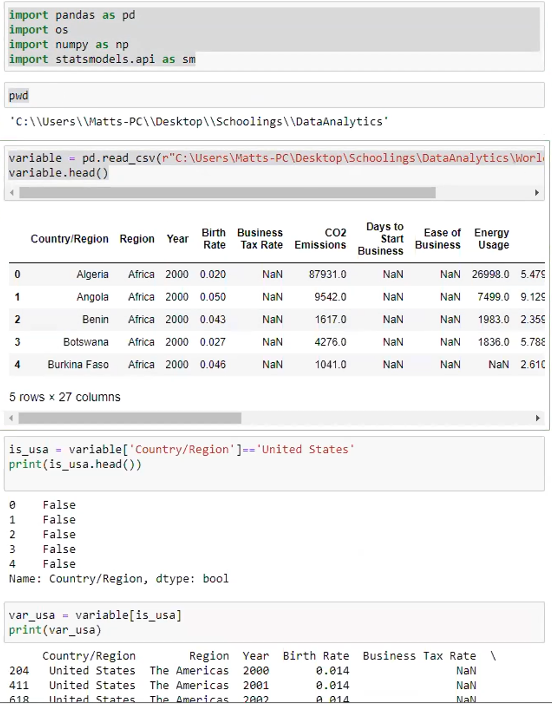


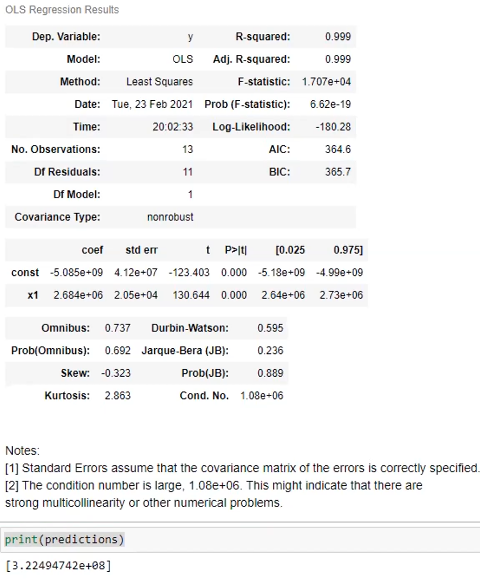


[01:50:00]



[01:53:30]





[02:05:13]

**Activity**:

* Model for Europe
* Life Expectancy Female vs GDP + Health Expense as %GDP, Infant mortality Life Expectancy Male

GDP = 1000000000,

Health.Exp...GDP = 0.04,

Infant.Mortality.Rate = 0.05,

Life.Expectancy.Male = 80