Attwifi: WP5i-UiF9-gA

Boot Camp Notes

https://techbootcamps.extendedcampus.utexas.edu/wp-content/uploads/sites/26/2017/06/uta\_curriculum\_overview\_data.pdf

Accept invite for Slack 2 weeks prior to starting bootcamp (contact student success manager if you don’t get email)

Install Mongo DB—instructions here: https://docs.mongodb.com/manual/tutorial/install-mongodb-on-os-x/

[Python Virtual Environment Docs](https://docs.python.org/3/tutorial/venv.html)

HW 1:Due Next Saturday (Recommended Target Thursday of Next Week)

Command+T to look sell

Request.get—will get a value

Json()—return value

Request.get().json()—chained

Highlight a range of numbers and rename those cells whatever you want

+N("Added buffer of 25%") –way to add comment to excel cell; will return value of 0

Matplot lib—source for conditional color formatting that is neutral

10/07/17

cd ~

mkdir repo

mkdir work

cd repo

pwd

open .

git clone paste link

excel tricks:

Learn about index-match

index(the range where your return value is, row number, [OPT

match(the value to look for, the range to look in , use 0 for example)

index-match(a row ex $A$4:$A$7, match(A

sparkline—an embed a graph in a cell.

Look at Colorgorical

3-5 Colors/dashboard

generates colors that go well together

Adobe Color CC

Data Cleaning

When you filter out elements like “blank” from data table, chart created reflects that change





Data>Data Analysis>Moving Average

Movine average helps smooth out data





Meetup—Feminist Hack

Linear Regression: powerful, fast, good for large data sets.

R-squared can be biased.

-download extensions for visual studio code

-pre-work

-do poll (slack)

-career services poll (in slack)

10/09/17

dim—new variable

as String—create array

Excel index with 0 for 1st storage place for array

End if—to end conditional

Dim example() as string leaving array quantity empty allows you to add as many variables as you want

Var—variable

VBA—all code starts with sub and end sub

pwd

/Users/anais/repo/UTAAU201710DATA1-Class-Repository-DATA

Sub=={} in other programming languages—asigns a functional group

MsgBox—example of camel block

*To grab something from git*

**repo** git clone https://github.com/the-Coding-Boot-Camp-at-UT/UTAAU201710DATA1-Class-Repository-DATA.git

fatal: destination path 'UTAAU201710DATA1-Class-Repository-DATA' already exists and is not an empty directory.

*It already exists so use ls to find what it’s called.*

**➜ repo** ls

**UTAAU201710DATA1-Class-Repository-DATA**

*To tell your terminal to find it*

**➜ repo** cd UTAAU201710DATA1-Class-Repository-DATA

**➜ UTAAU201710DATA1-Class-Repository-DATA** **git:(master)** git pull

To find where a file is

**➜ UTAAU201710DATA1-Class-Repository-DATA** **git:(master)** pwd

/Users/anais/repo/UTAAU201710DATA1-Class-Repository-DATA

**➜ UTAAU201710DATA1-Class-Repository-DATA** **git:(master)**

mkdir-make directory

CellsAndRanges():

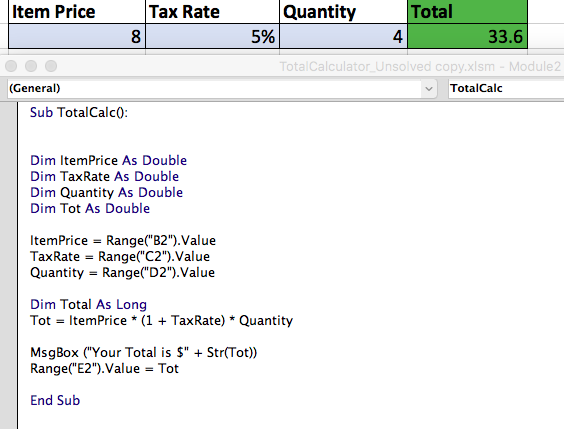
Sub Chessboard()

Cells(Row, Column). Value= “Cat”

Range(“F1”).Value= “text”

Range(“F5:F7”).Value= “text”

End Sub



Dim age1=5

Msgbox(“my age is” +str(age1))

Str(age1)—converts variable to text

Sub SimpleArrays():

Dim Ingredients (4\*) as String

Ingredients(0)= “Chocolate Bar”

Ingredients(1)= “PB”

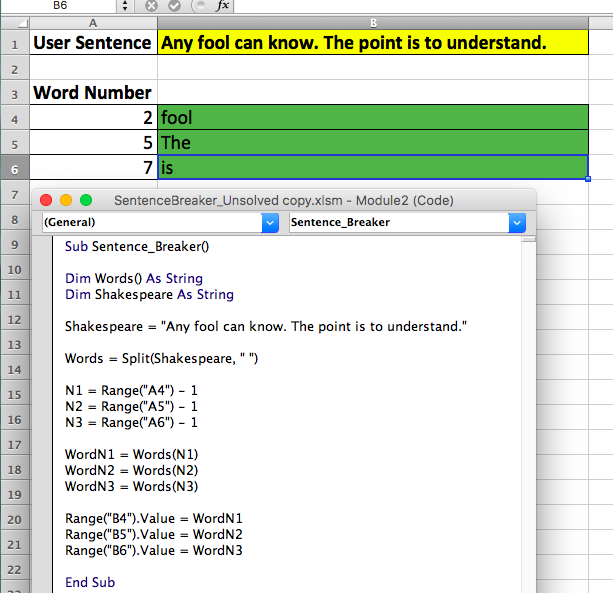
Ingredients(2)= “Jelly”

Ingredients(3)= “Macaroni”

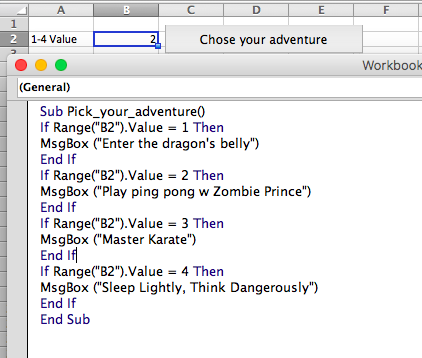
MsgBox(Ingredients (2))

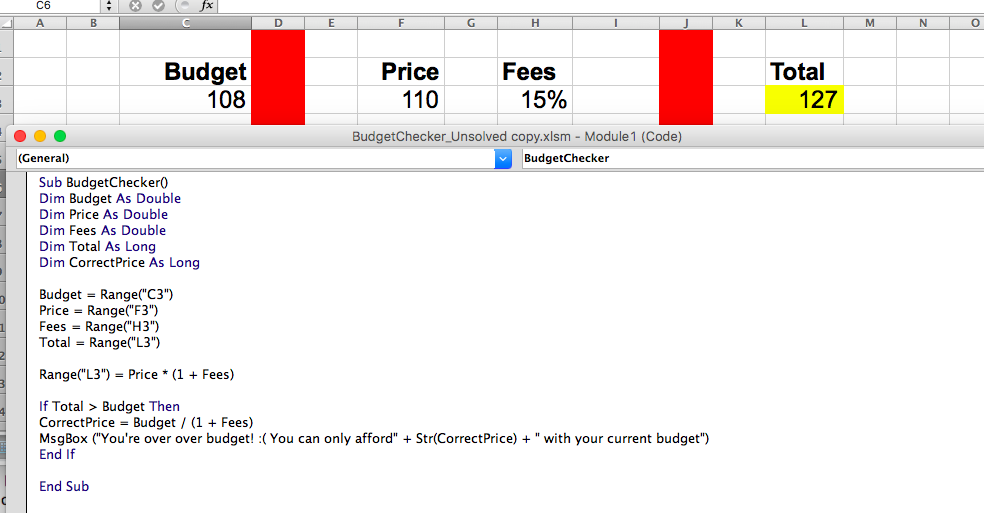
End Sub

* could replace with Ingredients()



Conditionals





Sub forLoop()

For i=1 to 20

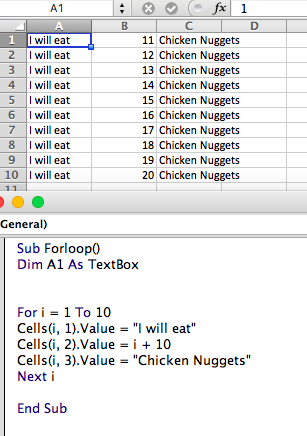
Cells(i, 1).Value=1

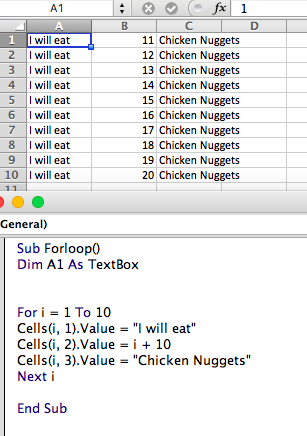
Cells(1, i).Value=5

Cells(i+1,2).Value=i+1

Next i

End Sub





Sub modulo()

Mod=yields remainder of division of two integers

2.4 mod 2 = 0

sub conditional\_loops()

if cells (I,1).Value mod 2=0 then

cell

Else

Cells(i, 2).Value = i

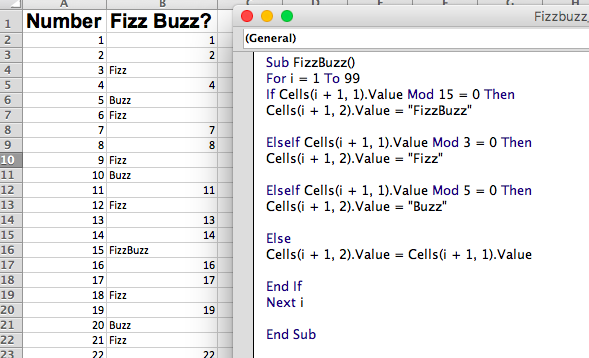
For i = 0 To 98

If Cells(i + 1, 1).Value Mod 3 = 0 Then

Cells(i + 1, 2).Value = "Fizz"

End If

Next i



EsleIf Lotto = 5865187

Range("F3") = First

Range("G3") = Last

Range("H3") = Lotto

Else: Lotto = 1841402

Range("F4") = First

Range("G4") = Last

Range("H4") = Lotto

Sub Lotto()

Dim First As Long

Dim Last As Long

Dim Lotto As Long

For i = 2 To 10001

Cells(i, 1).Value = First

Cells(i, 2).Value = Last

Cells(i, 3).Value = Lotto

If Lotto = 3957481 Then

Range("F2") = First

Range("G2") = Last

Range("H2") = Lotto

MsgBox ("Congratulations " + Range("F2").Value + " " + Range("G2").Value)

End If

Next i

End Sub

Exitfor---exits forloop when condition is met/terminates forloop

Sub Classscanner ()

For i=1 to 3

For j=1 to 5

MsbBox(“Row:” + str(i) + “Column:” + str(j) + “|” cells(i,j).value

Nextj

Nexti

End sub

10/14/17

Interview Tips

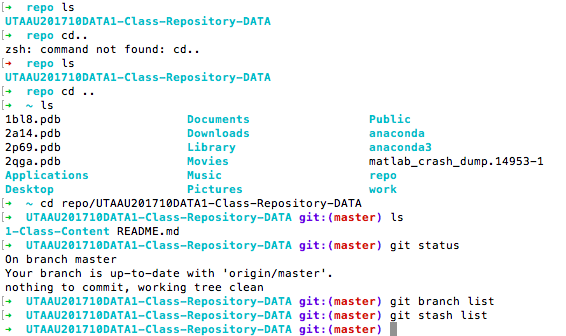
* STAR
* Clean up online presence and brand accordingly
* Github/LI
* Show your market strength
* What’s your story
* Thing outside of “formal education”

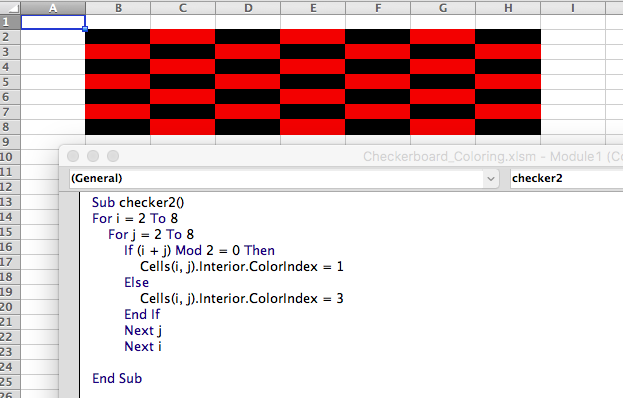
\*\*\*Top three bullets on linkedin\*\*\*

-further experience on request

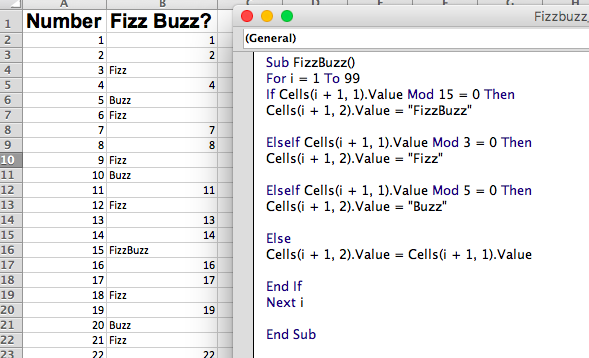
-make resume categorical

Range("A1,A3,A5,A7,B2,B4,B6,B8,C1,C3,C5,C7,D2,D4,D6,D8,E1,E3,E5,E7,F2,F4,F6,F8,G1,G3,G5,G7,H2,H4,H6,H8").Interior = 1









Dim yr1 as String

Dim yr2 as String

Dim yr3 as String

Dim yr4 as String

Dim yr5 as String

yr1=Range("C1")

yr2=Range("D1")

yr3=Range("E1")

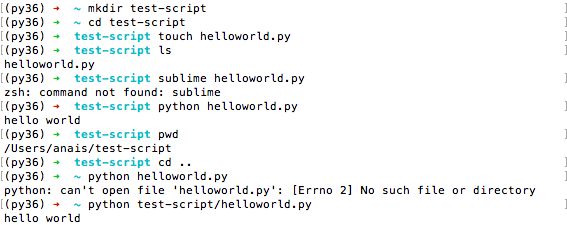
yr4=Range("F1")

yr5=Range("G1")

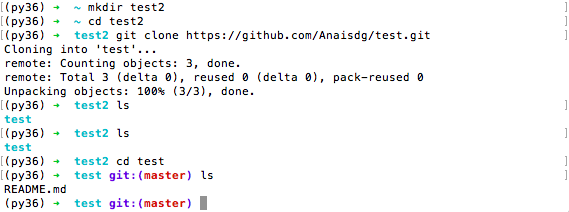
10/16/2017

PYTHON

How to create a folder through the terminal



1st go to git hub>+>new repository>1. Name 2.check public and initialize read.me>copy and paste link into terminal with git clone \*\*to create python repo in whatever folder that you’re in \*\*to make sure it works, look inside the created repo to make sure that you have a read.me file>



Now open up your python script in sublime. Change something to the read.me. save it

<p>Three Main Conclusions:</p>

<p>1. Charter schools outperform District schools in passing rate</p>

<p>2. There isn't a linear relationship between spedning per student and % passing</p>

<p>3. % Passing Math could be correlated with smaller size schools, but a larger sample size would be needed </p>

bins = [0,1000,2000,5000]

group\_names = ["Small (< 1000)", "Medium (1000-2000)", "Large (2000-5000)"]

pd.cut(school\_summary["Total Students"], bins, labels = group\_names)

school\_summary["Size"]= pd.cut(school\_summary["Total Students"], bins, labels = group\_names)

scores\_df = school\_summary[["Size", "Average Math Score","Average Reading Score", "% Passing Math", "% Passing Reading"]]

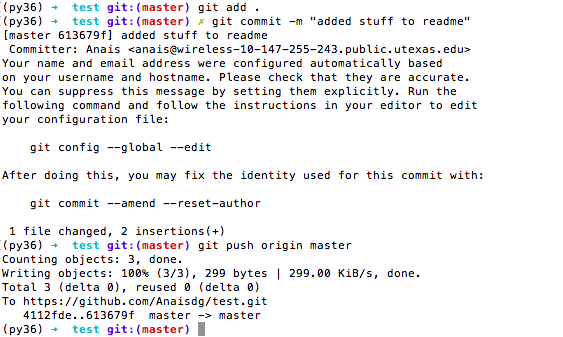
scores\_df = scores\_df.groupby("Size").max()

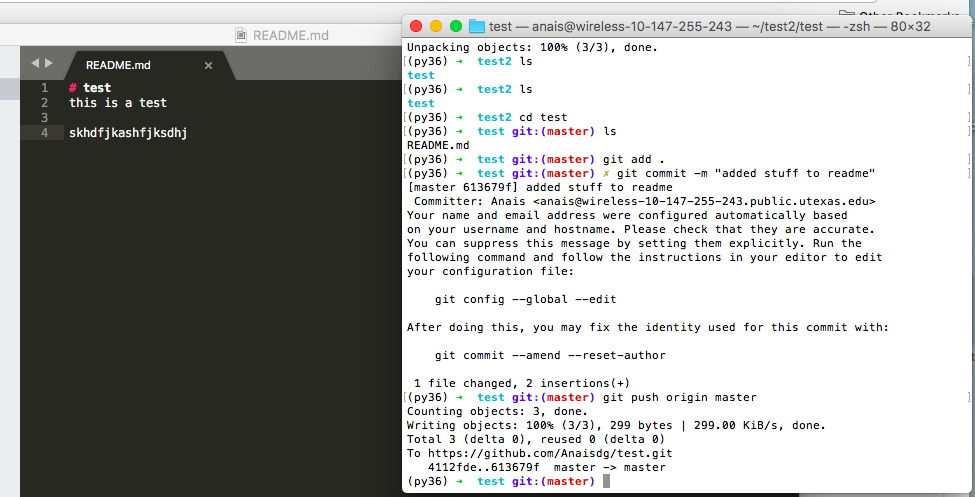
scores\_df

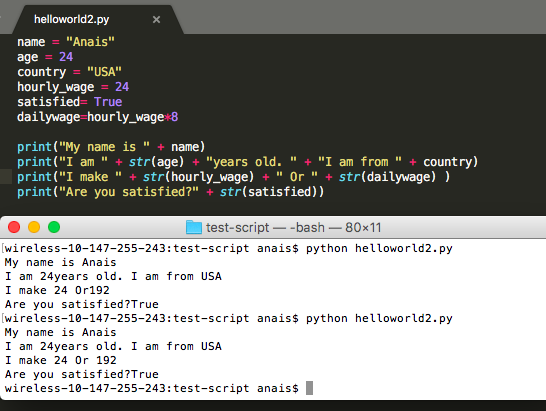
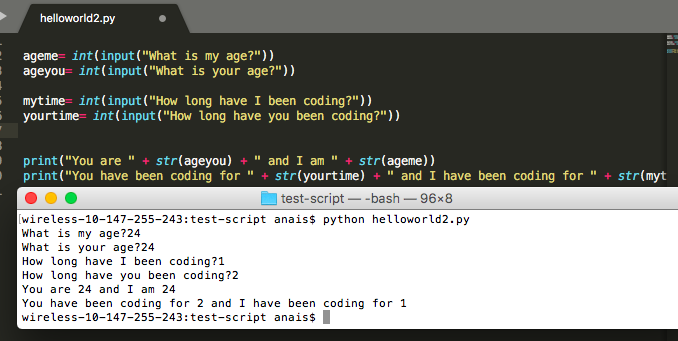
# Convert Total Students into int

school\_summary["Total Sutents"] = school\_summary["Total Students"].astype(float)

TO UPDATE/PUSH changes to git:







10/23/17

go to appropriate directory in terminal

“jupyter notebook” to open

“cntrl + C” to exit

to run the cell “shift + enter”

ITS NOT LINEAR—rather chronological (the cell that was most recently ran)

PANDAS

Pandas series==one dimensional—eg column

Pandas database==two dimensional—eg spreadsheet

Pandas orders dataframes columns ALPHABETICALLY

.head() returns first 5 sets of data by default

.describe() return statistical data on columns with numbers

data\_file\_pd[“Headername”] to get the column

data\_file\_pd[ [“Headername1”, “Headername2”] ]

.mean()

.sum()

.unique() The unique method shows every element of the series that appears only once

.value\_counts() Counts unique values in a column

(eg F F F F M F M F M == F: 6 & M:3)

#create new column

thousands\_of\_dollars = data\_file\_pd[“Amount”]/1000

#to add on to dataframe

data\_file\_pd[“Thousands of Dollars”] = thousands\_of\_dollars

.columns() returns an Index with you column names

.rename(columns={“Old name”: “Desired new name”, “Old name 2”: “Desired new

name 2”})

df=df\_original.set\_index(“last\_name”) changes the first row from an index row to another one of your rows

df.loc/iloc[“rows you want to select”, [“column1 you want to select”, “column2”]]

df.loc/iloc[“index of column of interest”, [“subsequent columns with index number relative to index column of interest”]]

PANDAS iloc[1:5] goes up to 5 BUT DOES NOT INCLUDE 5

NUMBERS==up to and not including

LETTERS==up to and including 🡪 df.loc

.loc[‘country’== ‘us’, :]

Iloc returns a dataframe

TO SELECT ALL just put “:”

DATA CLEANING

del df[‘title of column’]

df.head()

df.rename(columns={“state”: “sum of sitings”})

df[“duration (seconds)”].sum()

CONVERT OBJECT TO NUMERIC

df.to\_numeric(df[“column you want to convert”])

INCOMPLETE ROWS

df.count() returns a count for every column that contains a non no value—that contain data that can be used

DROP ALL ROWS WITH MISSSING INFORMATION

df= df.dropna(how= ‘any’) removes all rows where some information is null

VERIFY DROPPED ROWS

df.count()

now all columns should have the same value

df.dtypes

tells you the type of all of your data

df.Dataframe({“title1”:series 1, “title2:series 2})

dg.groupby(“filter1”, “filter2”)

df.sort\_values(‘column name’) DEFAULT DESCENDING

df.sort\_values(‘column name’, ascending=False)

\*\*\*after you sort as you like run

df.reset\_index(drop=True)

without “drop=True” you create a new column with the old index

CONVERSTS STRINGS TO NUMERIC DATA TYPES

df[“Amount”] = pd.to\_numeric(df[“Amount])

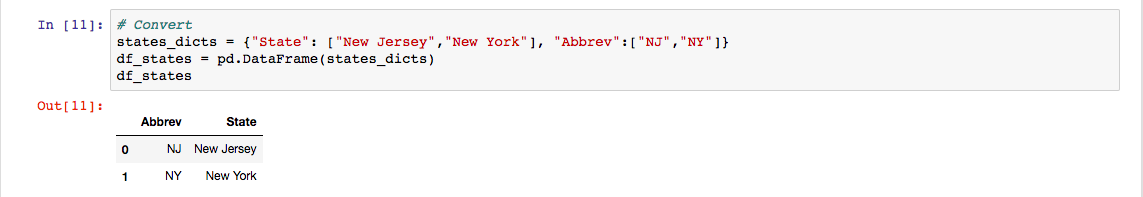
CLEAN UP CATEGORY

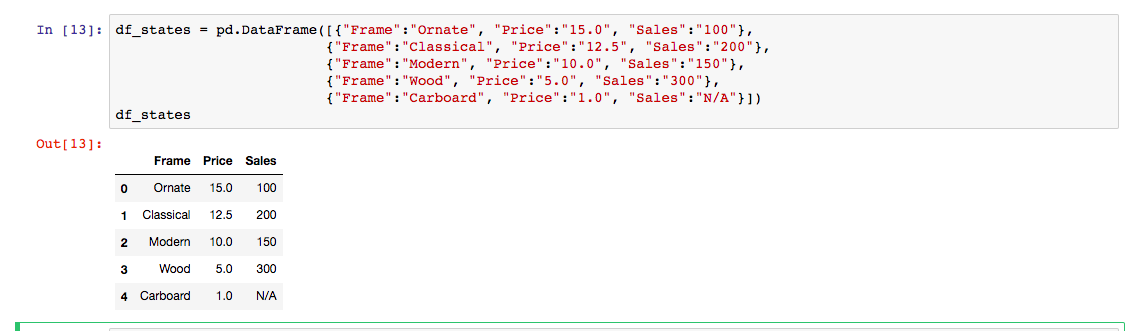
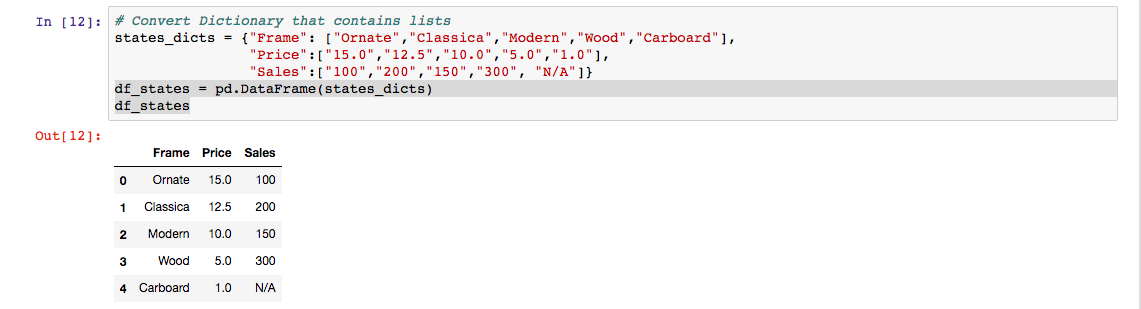
df[“Employer”]= df[“Employer”].replace({“Self Employed”: “Self”, “Self-employed”: “Self”})

GIVES SUMMARY STATISTICS FOR COLUMNS WITH NUMERIC VALUES

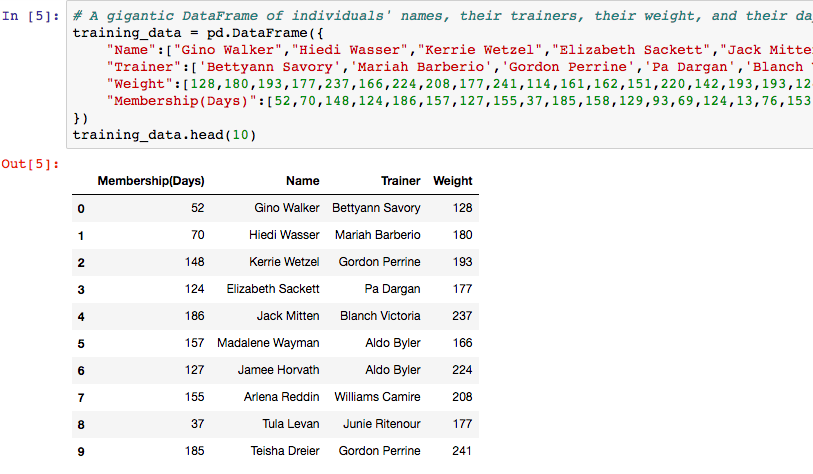
df.describe()

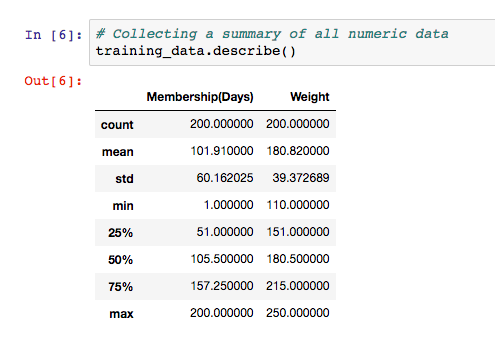




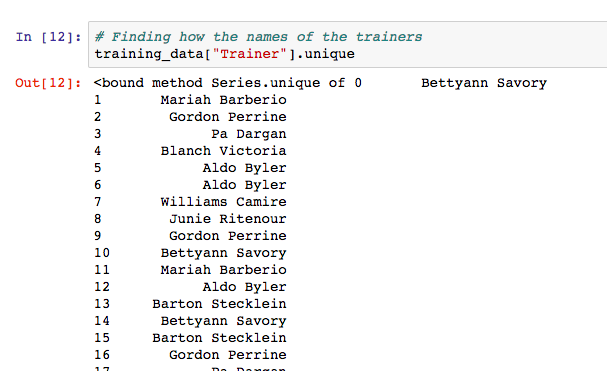


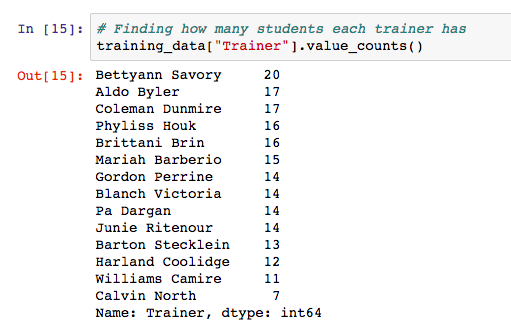


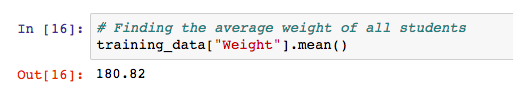












file\_one = “Resources/DataOne.csv”

file\_one\_df= pd.read\_csv(file\_one, encoding = “ISO-8859-1”)

file\_one\_df.head()

MERGE DATAFRAMES

Inner join only matches rows that exists in both files

Outer join will fill missing information with Na

Right join all the rows in one of the tables and leaves out rows from other tables that don’t have a match—from second df

Left join same as right join but only includes data from first df

DEFAULT = Inner

.merge(left df, right df, on = “column dfs have in common”, how = “outer”)

FILE PATH PANDAS

“…/Resource/file.csv”

BUNNING DATA

Bins = [0, 25, 50, 75, 100]

Group\_names= [‘Low’, ‘Okay’, ‘Good’, ‘Great’]

pd.cut(df[‘Test Score’], bins, labels=group\_names)

returns a pandas series that corresponds to the “test score” column

create a new column in your df

original\_df[‘Test Score Summary’] = pd.cut(df[‘Test Score’], bins, labels=group\_names)

JSON

Json\_path = “resources/purchase\_data.json”

df = pd.read\_rson(json\_path)

df.to\_jason(“resources.json\_output.json”, orient=’records’)

orient = records converts each row into a dictionary and exports the data as a list of dictionaries

np.arange(1,7,1)

DOES NOT INCLUDE 7

Will return 1,2,3,4,5,6

Plt.plot(0,0,10, alpha=0.25)

Creates horizontal line with transparency of .25

Y =0

X=0 to 10

Sine\_Handle, = plt.plot(x\_axis, sin, marker = ‘o’, color= “blue”, label = “Sine”)

Cosine\_handle, = plt.plot(x\_axis, cos, marker = ‘^’, color= “blue”, label = “Cosine”)

^, @ the end of handle creates the handles as a tuple

Need to create a tuples with one value in each

Plt.legend(handles=[sine\_handle, cosine\_handle], loc= “lower right”)

Plt.savefig(“lineConfig.png)

Plt.show()

<https://matplotlib.org/users/pyplot_tutorial.html>

markers

<https://matplotlib.org/api/markers_api.html>

APIS

import requests as req

impor json

url = “ “

response = req.get(response)

response

response = response.json()

print(json.dumps(response, indent=4, sort\_keys=True))

to append id

response = req.get(url + “append part”).json()

print(response

pip install citipy

for index, row in df.iterrrows():