

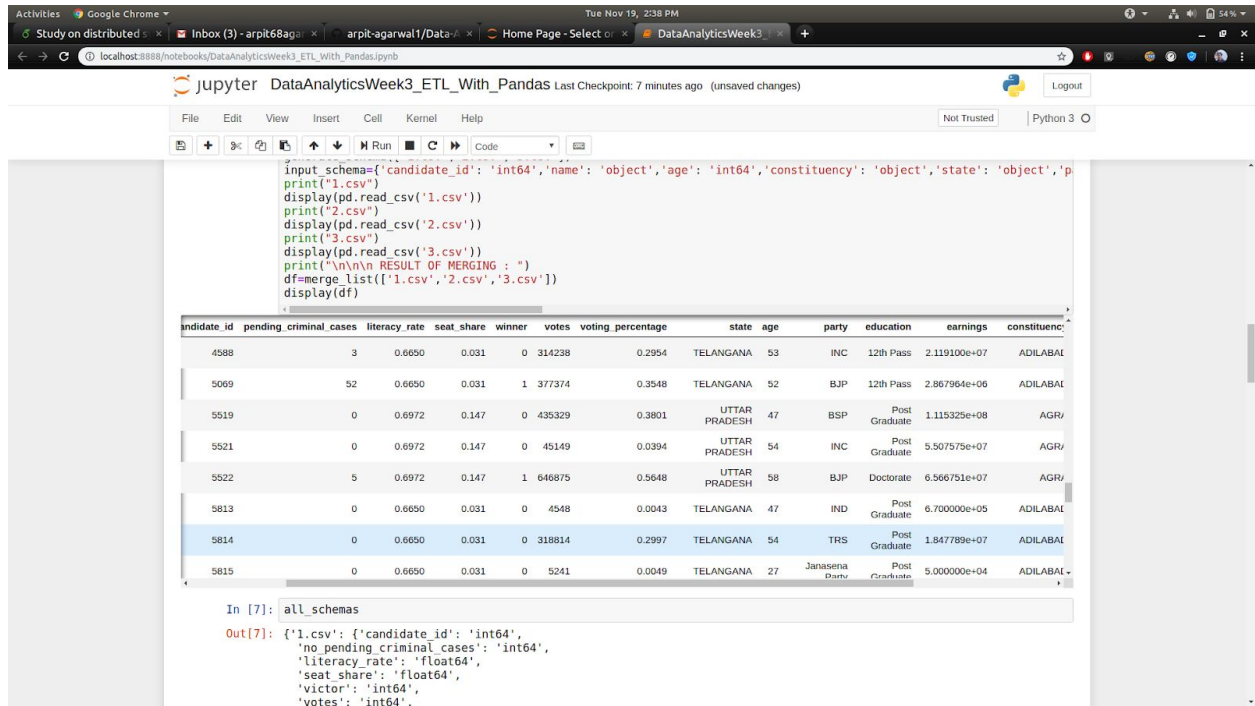
# Adithya Kiran || Chirag P Tubakad || Arpit Agarwal || Nandakrishna

## ASSIGNMENT 3

### Problem Statement:

Perform simple Extract Transform Load (ETL) functions with Python modules on two separate datasets and merge them into one.

**Dataset:** 3 custom sets made from dataset of assignment 1 (uploaded to repo)



The screenshot shows a Jupyter Notebook interface with the following code in a cell:

```
input_schema={'candidate_id': 'int64', 'name': 'object', 'age': 'int64', 'constituency': 'object', 'state': 'object', 'p
print("1.csv")
display(pd.read_csv('1.csv'))
print("2.csv")
display(pd.read_csv('2.csv'))
print("3.csv")
display(pd.read_csv('3.csv'))
print("\n\n\n RESULT OF MERGING : ")
df=merge_list(['1.csv', '2.csv', '3.csv'])
display(df)
```

The output of the code is a table with the following columns: candidate\_id, pending\_criminal\_cases, literacy\_rate, seat\_share, winner, votes, voting\_percentage, state, age, party, education, earnings, constituency. The table contains 10 rows of data.

candidate_id	pending_criminal_cases	literacy_rate	seat_share	winner	votes	voting_percentage	state	age	party	education	earnings	constituency
4588	3	0.6650	0.031	0	314238	0.2954	TELANGANA	53	INC	12th Pass	2.119100e+07	ADILABAT
5069	52	0.6650	0.031	1	377374	0.3548	TELANGANA	52	BJP	12th Pass	2.867964e+06	ADILABAT
5519	0	0.6972	0.147	0	435329	0.3801	UTTAR PRADESH	47	BSP	Post Graduate	1.115325e+08	AGRA
5521	0	0.6972	0.147	0	45149	0.0394	UTTAR PRADESH	54	INC	Post Graduate	5.507579e+07	AGRA
5522	5	0.6972	0.147	1	646875	0.5648	UTTAR PRADESH	58	BJP	Doctorate	6.566751e+07	AGRA
5813	0	0.6650	0.031	0	4548	0.0043	TELANGANA	47	IND	Post Graduate	6.700000e+05	ADILABAT
5814	0	0.6650	0.031	0	318814	0.2997	TELANGANA	54	TRS	Post Graduate	1.847789e+07	ADILABAT
5815	0	0.6650	0.031	0	5241	0.0049	TELANGANA	27	Janasena Party	Post Graduate	5.000000e+04	ADILABAT

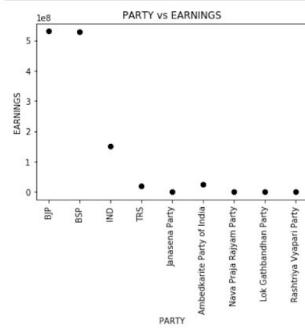
The output of the code is also shown below the table:

```
In [7]: all_schemas
Out[7]: {'1.csv': {'candidate_id': 'int64',
'no pending criminal cases': 'int64',
'literacy_rate': 'float64',
'seat_share': 'float64',
'winner': 'int64',
'votes': 'int64',
```

jupyter DataAnalyticsWeek3\_ETL\_With\_Pandas Last Checkpoint: 5 minutes ago (unsaved changes)

File Edit View Insert Cell Kernel Help Not Trusted Python 3

```
plt.title('PARTY vs EARNINGS')  
plt.show()  
partyVsEarnings()
```

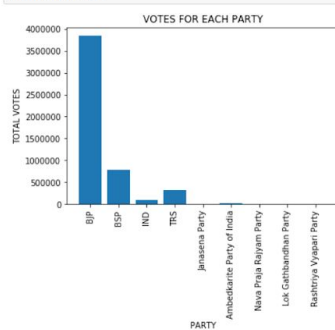


```
In [15]: def educationVsVotes():  
    list1 = []; list2 = []  
    list1.append(df['education'][0])  
    list2.append(df['votes'][0])  
    for index, row in df['education'][1:].items():  
        if df['education'][index] in list1:  
            index2 = list1.index(df['education'][index])  
            list2[index2] += df['votes'][index]  
        else:  
            list1.append(df['education'][index])  
            list2.append(df['votes'][index])
```

jupyter DataAnalyticsWeek3\_ETL\_With\_Pandas Last Checkpoint: 5 minutes ago (unsaved changes)

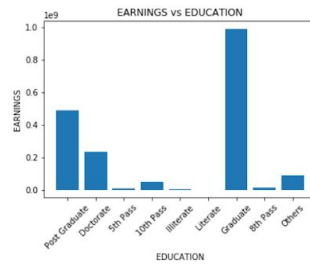
File Edit View Insert Cell Kernel Help Not Trusted Python 3

```
list1.append(df['party'][index])  
list2.append(df['votes'][index])  
x = list1[1:10]; y = list2[1:10]  
plt.bar(x, y)  
plt.xlabel('PARTY'); plt.ylabel('TOTAL VOTES')  
plt.xticks(x, x, rotation=90)  
plt.title('VOTES FOR EACH PARTY')  
plt.show()  
partyVsVotes()
```



```
In [14]: def partyVsEarnings():  
    list1 = []; list2 = []  
    list1.append(df['party'][0])  
    list2.append(df['earnings'][0])  
    for index, row in df['party'][1:].items():
```

```
list2.append(df['earnings'][1:index])  
x = list1[1:10]  
y = list2[1:10]  
plt.bar(x,y)  
plt.xlabel('EDUCATION');plt.ylabel('EARNINGS')  
plt.xticks(x,x,rotation=45)  
plt.title('EARNINGS vs EDUCATION')  
plt.show()  
educationVsEarning()
```



In [ ]: