Problem 1

The object "dat" created in the assignment code will import the survey data for the assignment using read_csv, thereby creating a tibble. Using that object as your data, use select() to create a new tibble that include only the columns for educational level, whether the respondent has an educational loan, employment status, and Trump approval. Display that object. Hint: consult the codebook to identify the correct column names.

Write your code below:

Import Libraries

In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import statsmodels.api as sm
import datetime
from datetime import datetime, timedelta
import scipy.stats
import pandas_profiling
from pandas_profiling import ProfileReport


%matplotlib inline
#sets the default autosave frequency in seconds
%autosave 60
sns.set_style('dark')
sns.set(font_scale=1.2)

plt.rc('axes', titlesize=9)
plt.rc('axes', labelsize=14)
plt.rc('xtick', labelsize=12)
plt.rc('ytick', labelsize=12)

import warnings
warnings.filterwarnings('ignore')

# Use Folium library to plot values on a map.
#import folium

#import feature_engine.missing_data_imputers as mdi
#from feature_engine.outlier_removers import Winsorizer
#from feature_engine import categorical_encoders as ce

pd.set_option('display.max_columns',None)
#pd.set_option('display.max_rows',None)
pd.set_option('display.width', 1000)
pd.option_context('float_format','{:.2f}'.format)

np.random.seed(0)
np.set_printoptions(suppress=True)
```

```
Autosaving every 60 seconds
```

In [2]:
```
df = pd.read_csv("cces_sample_coursera.csv")
```

In [3]:
```
df
```

Out[3]:

	caseid	region	gender	educ	edloan	race	hispanic	employ	marstat	pid7	ideo5	pew_religimp	newsint	faminc_new	union
0	417614315	3	1	2	2.0	1	2	5	3	6	3	1.0	2	1	3.0
1	415490556	1	2	6	2.0	1	1	1	1	2	2	3.0	3	12	3.0
2	414351505	3	2	3	2.0	2	2	1	4	2	3	1.0	3	4	3.0
3	411855339	1	2	5	2.0	6	2	5	3	3	1	2.0	1	6	2.0
4	417056957	2	1	2	NaN	4	2	8	5	1	1	4.0	2	4	3.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
995	410945270	2	1	6	2.0	1	2	5	1	3	3	2.0	1	14	3.0
996	414240025	2	2	5	1.0	2	2	1	5	1	2	2.0	3	6	3.0
997	420156780	2	2	5	2.0	1	2	1	5	2	3	3.0	1	6	3.0
998	412382118	2	2	6	NaN	3	2	5	1	5	4	1.0	1	9	3.0
999	414206234	2	1	1	NaN	7	2	4	5	4	5	1.0	2	1	3.0

1000 rows × 25 columns

In [4]:
```
df.columns
```

Out[4]:
```
Index(['caseid', 'region', 'gender', 'educ', 'edloan', 'race', 'hispanic', 'employ', 'marstat', 'pid7', 'ideo5', 'pew_religimp', 'newsint', 'faminc_new', 'union', 'investor', 'CC18_308a', 'CC18_310a', 'CC18_310b', 'CC18_310c', 'CC18_310d', 'CC18_325a', 'CC18_325b', 'CC18_325c', 'CC18_325d'], dtype='object')
```

In [5]:
```
df2 = df[["educ","edloan","CC18_308a"]]
df2
```

Out[5]:

	educ	edloan	CC18_308a
0	2	2.0	2
1	6	2.0	4
2	3	2.0	4
3	5	2.0	4
4	2	NaN	4
...	...	...	...
995	6	2.0	4
996	5	1.0	4
997	5	2.0	4
998	6	NaN	1
999	1	NaN	4

1000 rows × 3 columns

Problem 2

Continuing to use the new data table you created in Problem 1, use recode() to create a new column named "trump_approve_disapprove" that recodes the column for President Trump's job approval. A value of "1" should mean that the respondent either "strongly" or "somewhat" approves of the President, and a value of 0 should mean that the respondent either "strongly" or "somewhat" DISapproves of the president. Display the resulting object.

Write your code below:

In [6]:
```
df2["CC18_308a"].value_counts()
```

Out[6]:
```
4    487
1    256
2    167
3     90
Name: CC18_308a, dtype: int64
```

1 Strongly approve

2 Somewhat approve

3 Somewhat disapprove

4 Strongly disapprove

In [7]:
```
df2["CC18_308a"].replace(to_replace=2, value=1, inplace=True)
```

In [8]:
```
df2["CC18_308a"].replace(to_replace=3, value=0, inplace=True)
```

In [9]:
```
df2["CC18_308a"].replace(to_replace=4, value=0, inplace=True)
```

In [10]:
```
df2["CC18_308a"].value_counts()
```

Out[10]:
```
0    577
1    423
Name: CC18_308a, dtype: int64
```

In [11]:
```
df2
```

Out[11]:

	educ	edloan	CC18_308a
0	2	2.0	1
1	6	2.0	0
2	3	2.0	0
3	5	2.0	0
4	2	NaN	0
...	...	...	...
995	6	2.0	0
996	5	1.0	0
997	5	2.0	0
998	6	NaN	1
999	1	NaN	0

1000 rows × 3 columns

Problem 3

Use summarise() to create a summary table for survey respondents who are employed full time and are married. The table should have the mean and median for the importance of religion column.

Write your code below:

In [12]:
```
df.head()
```

Out[12]:

	caseid	region	gender	educ	edloan	race	hispanic	employ	marstat	pid7	ideo5	pew_religimp	newsint	faminc_new	union	in
0	417614315	3	1	2	2.0	1	2	5	3	6	3	1.0	2	1	3.0	
1	415490556	1	2	6	2.0	1	1	1	1	2	2	3.0	3	12	3.0	
2	414351505	3	2	3	2.0	2	2	1	4	2	3	1.0	3	4	3.0	
3	411855339	1	2	5	2.0	6	2	5	3	3	1	2.0	1	6	2.0	
4	417056957	2	1	2	NaN	4	2	8	5	1	1	4.0	2	4	3.0	

In [13]:
```
df3 = df[["employ","marstat","pew_religimp"]]
```

In [14]:
```
df4 = df3[(df3["employ"] == 1) & (df3["marstat"] == 1)]
df4
```

Out[14]:

	employ	marstat	pew_religimp
1	1	1	3.0
5	1	1	4.0
8	1	1	4.0
13	1	1	1.0
20	1	1	4.0
...	...	...	...
968	1	1	4.0
979	1	1	1.0
990	1	1	2.0
993	1	1	1.0
994	1	1	1.0

233 rows × 3 columns

In [15]:
```
df4.describe()
```

Out[15]:

	employ	marstat	pew_religimp
count	233.0	233.0	233.000000
mean	1.0	1.0	2.188841
std	0.0	0.0	1.184838
min	1.0	1.0	1.000000
25%	1.0	1.0	1.000000
50%	1.0	1.0	2.000000
75%	1.0	1.0	3.000000
max	1.0	1.0	4.000000