Achraf Safsafi

DSC 540

Project: Milestone 5

load the first table into a database

In [1]:
```python
import pandas
import sqlite3
import csv_to_sqlite

conn = sqlite3.connect("project.db")

conn.execute("CREATE TABLE if not exists df1 (State TEXT,County TEXT,Violent_crime INT(10),Murder INT(10),"
            "Rape INT(10),Robbery INT(10),Aggravated_assault INT(10),Property_crime INT(10),Burglary INT(10),"
            "Larceny_theft INT(10),Motor_vehicle_theft INT(10),Arson INT(10))")


df1 = pandas.read_csv("df1.csv")
df1
```

Out[1]:

	State	County	Violent_crime	Murder	Rape	Robbery	Aggravated_assault	Property_crime	Burglary	Larceny_theft	Motor_
0	ALABAMA	Autauga	51.0	0.0	6.0	5.0	40.0	372.0	92.0	240	
1	ALABAMA	Baldwin	223.0	0.0	9.0	37.0	177.0	615.0	173.0	397	
2	ALABAMA	Blount	375.0	1.0	19.0	5.0	350.0	796.0	191.0	492	
3	ALABAMA	Calhoun	14.0	0.0	5.0	7.0	2.0	144.0	49.0	95	
4	ALABAMA	Elmore	68.0	4.0	30.0	12.0	22.0	669.0	178.0	427	
...	...	...	...	...	...	...	...	...	...	...	
2351	WYOMING	Sublette	4.0	0.0	0.0	0.0	4.0	49.0	11.0	35	
2352	WYOMING	Sweetwater	22.0	0.0	8.0	0.0	14.0	77.0	17.0	52	
2353	WYOMING	Uinta	7.0	1.0	1.0	1.0	4.0	53.0	12.0	35	
2354	WYOMING	Washakie	0.0	0.0	0.0	0.0	0.0	17.0	4.0	13	
2355	WYOMING	Weston	5.0	0.0	1.0	0.0	4.0	0.0	0.0	0	

2356 rows × 12 columns

In [2]:
```python
df1.to_sql("table1", conn, if_exists='append', index=False)
c = conn.cursor()
c.execute("select * from table1 LIMIT 10")
print(tuple(d[0] for d in c.description))
for row in c:
    print(row)
```

```
('State', 'County', 'Violent_crime', 'Murder', 'Rape', 'Robbery', 'Aggravated_assault', 'Property_crime', 'Burglary', 'Larceny_theft', 'Motor_vehicle_theft', 'Arson')
('ALABAMA', 'Autauga', 51.0, 0.0, 6.0, 5.0, 40.0, 372.0, 92.0, 240, 40.0, 3.0)
('ALABAMA', 'Baldwin', 223.0, 0.0, 9.0, 37.0, 177.0, 615.0, 173.0, 397, 45.0, 3.0)
('ALABAMA', 'Blount', 375.0, 1.0, 19.0, 5.0, 350.0, 796.0, 191.0, 492, 113.0, 3.0)
('ALABAMA', 'Calhoun', 14.0, 0.0, 5.0, 7.0, 2.0, 144.0, 49.0, 95, 0.0, 3.0)
('ALABAMA', 'Elmore', 68.0, 4.0, 30.0, 12.0, 22.0, 669.0, 178.0, 427, 64.0, 3.0)
('ALABAMA', 'Etowah', 99.0, 2.0, 32.0, 1.0, 64.0, 444.0, 152.0, 233, 59.0, 3.0)
('ALABAMA', 'Geneva', 40.0, 1.0, 7.0, 2.0, 30.0, 202.0, 74.0, 112, 16.0, 3.0)
('ALABAMA', 'Greene', 31.0, 0.0, 2.0, 1.0, 28.0, 123.0, 45.0, 60, 18.0, 3.0)
('ALABAMA', 'Houston', 121.0, 1.0, 9.0, 19.0, 92.0, 657.0, 187.0, 400, 70.0, 3.0)
('ALABAMA', 'Jefferson', 729.0, 10.0, 45.0, 128.0, 546.0, 2255.0, 1181.0, 888, 186.0, 3.0)
```

load the second table into a database

In [3]:
```python
conn.execute("CREATE TABLE if not exists df2 (State TEXT,County TEXT,ANSI_Code INT(10),Total INT(10),Aged INT(10),"
            "Blind_and_disabled INT(10),Age_Under18 INT(10),"
            "Age_18-64 INT(10),Age65_or_older INT(10),SSI_recipients_also_receiving_OASDI INT(10),"
           "Amount_of_payments INT(10))")


df2 = pandas.read_csv("df2.csv")
df2
```

Out[3]:

	State	County	ANSI_Code	Total	Aged	Blind_and_disabled	Age_Under18	Age_18–64	Age_65_or_older	SSI_recipients_also_
0	ALABAMA	Autauga	1001	1439	60	1379	200	1027	212	
1	ALABAMA	Baldwin	1003	3505	200	3305	599	2345	561	
2	ALABAMA	Barbour	1005	1395	100	1295	188	883	324	
3	ALABAMA	Bibb	1007	896	28	868	74	665	157	
4	ALABAMA	Blount	1009	1227	57	1170	115	887	225	
...	...	...	...	...	...	...	...	...	...	
1632	VIRGINIA	Staunton	51790	825	50	775	101	566	158	
1633	VIRGINIA	Suffolk	51800	2356	132	2224	336	1616	404	
1634	VIRGINIA	Virginia Beach	51810	5610	627	4983	929	3583	1098	
1635	VIRGINIA	Waynesboro	51820	730	32	698	112	518	100	
1636	VIRGINIA	Winchester	51840	735	34	701	135	496	104	

1637 rows × 11 columns

In [4]:
```python
df2.to_sql("table2", conn, if_exists='append', index=False)
c = conn.cursor()
c.execute("select * from table2 LIMIT 10")
print(tuple(d[0] for d in c.description))
for row in c:
    print(row)
```

```
('State', 'County', 'ANSI_Code', 'Total', 'Aged', 'Blind_and_disabled', 'Age_Under18', 'Age_18-64', 'Age_65_or_older', 'SSI_recipients_also_receiving_OASDI', 'Amount_of_payments')
('ALABAMA', 'Autauga', 1001, 1439, 60, 1379, 200, 1027, 212, 509, 826)
('ALABAMA', 'Baldwin', 1003, 3505, 200, 3305, 599, 2345, 561, 1297, 1917)
('ALABAMA', 'Barbour', 1005, 1395, 100, 1295, 188, 883, 324, 557, 780)
('ALABAMA', 'Bibb', 1007, 896, 28, 868, 74, 665, 157, 368, 469)
('ALABAMA', 'Blount', 1009, 1227, 57, 1170, 115, 887, 225, 484, 654)
('ALABAMA', 'Bullock', 1011, 549, 26, 523, 96, 330, 123, 175, 329)
('ALABAMA', 'Butler', 1013, 1041, 61, 980, 112, 688, 241, 423, 560)
('ALABAMA', 'Calhoun', 1015, 4532, 152, 4380, 552, 3358, 622, 1624, 2637)
('ALABAMA', 'Chambers', 1017, 1422, 101, 1321, 165, 940, 317, 520, 768)
('ALABAMA', 'Cherokee', 1019, 830, 35, 795, 67, 627, 136, 296, 448)
```

load the third table into a database

In [5]:
```python
conn.execute("CREATE TABLE if not exists df3 (County TEXT,AGE65_69 INT(10),AGE70_74 INT(10),"
            "AGE75_79 INT(10),AGE80_84 INT(10),AGE85_over INT(10),State_Code INT(10),"
            "County_Code INT(10),State INT(10))")


df3 = pandas.read_csv("df3.csv")
df3
```

Out[5]:

	County	AGE65_69	AGE70_74	AGE75_79	AGE80_84	AGE85_over	State_Code	County_Code	State
0	Shelby	11940	8412	6232	3686	3025	1	117	ALABAMA
1	Talladega	4531	4226	2482	1910	1273	1	121	ALABAMA
2	Tuscaloosa	9310	7501	5411	2713	2537	1	125	ALABAMA
3	Pinal	27844	27162	18482	11173	6164	4	21	ARIZONA
4	Mendocino	6576	5744	2860	2417	1475	6	45	CALIFORNIA
...	...	...	...	...	...	...	...	...	...
833	Butte	14949	10320	8903	4207	4441	6	7	CALIFORNIA
834	El Dorado	14742	10686	6861	3685	4453	6	17	CALIFORNIA
835	Imperial	8092	5267	4116	3392	2645	6	25	CALIFORNIA
836	Kern	34857	24449	15794	12300	10676	6	29	CALIFORNIA
837	Marin	18825	14871	11099	7072	6346	6	41	CALIFORNIA

838 rows × 9 columns

In [6]:
```python
df3.to_sql("table3", conn, if_exists='append', index=False)
c = conn.cursor()
c.execute("select * from table3 LIMIT 10")
print(tuple(d[0] for d in c.description))
for row in c:
    print(row)
```

```
('County', 'AGE65_69', 'AGE70_74', 'AGE75_79', 'AGE80_84', 'AGE85_over', 'State_Code', 'County_Code', 'State')
('Shelby ', 11940, 8412, 6232, 3686, 3025, 1, 117, ' ALABAMA')
('Talladega ', 4531, 4226, 2482, 1910, 1273, 1, 121, ' ALABAMA')
('Tuscaloosa ', 9310, 7501, 5411, 2713, 2537, 1, 125, ' ALABAMA')
('Pinal ', 27844, 27162, 18482, 11173, 6164, 4, 21, ' ARIZONA')
('Mendocino ', 6576, 5744, 2860, 2417, 1475, 6, 45, ' CALIFORNIA')
('Orange ', 142280, 124254, 80776, 58499, 65417, 6, 59, ' CALIFORNIA')
('Sacramento ', 70410, 57822, 36056, 25650, 27506, 6, 67, ' CALIFORNIA')
('Citrus ', 16405, 13054, 10321, 7242, 6810, 12, 17, ' FLORIDA')
('Peoria ', 9622, 7828, 4479, 3712, 4560, 17, 143, ' ILLINOIS')
('Hancock ', 3932, 4173, 2182, 1487, 1174, 18, 59, ' INDIANA')
```

Merge table1 and table2

In [7]:
```python
merged_df1= df2.reset_index().merge(df1.reset_index(),left_index=True, right_index=True,  how='outer',on = ['County', 'State'])
merged_df1 = merged_df1.dropna(axis=0)
merged_df1
```

Out[7]:

	index_x	State	County	ANSI_Code	Total	Aged	Blind_and_disabled	Age_Under18	Age_18–64	Age_65_or_older	...	Viole
0	0.0	ALABAMA	Autauga	1001.0	1439.0	60.0	1379.0	200.0	1027.0	212.0	...	
1	1.0	ALABAMA	Baldwin	1003.0	3505.0	200.0	3305.0	599.0	2345.0	561.0	...	
2	2.0	ALABAMA	Barbour	1005.0	1395.0	100.0	1295.0	188.0	883.0	324.0	...	
3	3.0	ALABAMA	Bibb	1007.0	896.0	28.0	868.0	74.0	665.0	157.0	...	
4	4.0	ALABAMA	Blount	1009.0	1227.0	57.0	1170.0	115.0	887.0	225.0	...	
...	...	...	...	...	...	...	...	...	...	...	...	
1632	1632.0	VIRGINIA	Staunton	51790.0	825.0	50.0	775.0	101.0	566.0	158.0	...	
1633	1633.0	VIRGINIA	Suffolk	51800.0	2356.0	132.0	2224.0	336.0	1616.0	404.0	...	
1634	1634.0	VIRGINIA	Virginia Beach	51810.0	5610.0	627.0	4983.0	929.0	3583.0	1098.0	...	
1635	1635.0	VIRGINIA	Waynesboro	51820.0	730.0	32.0	698.0	112.0	518.0	100.0	...	
1636	1636.0	VIRGINIA	Winchester	51840.0	735.0	34.0	701.0	135.0	496.0	104.0	...	

1637 rows × 23 columns

In [8]:
```python
merged_df1.isnull().sum()
```

Out[8]:
```
index_x                                0
State                                  0
County                                 0
ANSI_Code                              0
Total                                  0
Aged                                   0
Blind_and_disabled                     0
Age_Under18                            0
Age_18-64                              0
Age_65_or_older                        0
SSI_recipients_also_receiving_OASDI    0
Amount_of_payments                     0
index_y                                0
Violent_crime                          0
Murder                                 0
Rape                                   0
Robbery                                0
Aggravated_assault                     0
Property_crime                         0
Burglary                               0
Larceny_theft                          0
Motor_vehicle_theft                    0
Arson                                  0
dtype: int64
```

In [9]:
```python
merged_df1.shape
```

Out[9]:
```
(1637, 23)
```

Merge all data sets into one table

In [10]:
```python
merged_df = df3.reset_index().merge(merged_df1.reset_index(), left_index=True, right_index=True, how='outer',on = ['County','State'])
merged_df = merged_df.dropna(axis=0)
merged_df = merged_df.drop('index_x', 1)
merged_df = merged_df.drop('index_y', 1)
merged_df
```

Out[10]:

	County	AGE65_69	AGE70_74	AGE75_79	AGE80_84	AGE85_over	State_Code	County_Code	State	ANSI_Code	...	Vio
0	Shelby	11940.0	8412.0	6232.0	3686.0	3025.0	1.0	117.0	ALABAMA	1001.0	...	
1	Talladega	4531.0	4226.0	2482.0	1910.0	1273.0	1.0	121.0	ALABAMA	1003.0	...	
2	Tuscaloosa	9310.0	7501.0	5411.0	2713.0	2537.0	1.0	125.0	ALABAMA	1005.0	...	
3	Pinal	27844.0	27162.0	18482.0	11173.0	6164.0	4.0	21.0	ARIZONA	1007.0	...	
4	Mendocino	6576.0	5744.0	2860.0	2417.0	1475.0	6.0	45.0	CALIFORNIA	1009.0	...	
...	...	...	...	...	...	...	...	...	...	...	...	
833	Butte	14949.0	10320.0	8903.0	4207.0	4441.0	6.0	7.0	CALIFORNIA	29101.0	...	
834	El Dorado	14742.0	10686.0	6861.0	3685.0	4453.0	6.0	17.0	CALIFORNIA	29105.0	...	
835	Imperial	8092.0	5267.0	4116.0	3392.0	2645.0	6.0	25.0	CALIFORNIA	29107.0	...	
836	Kern	34857.0	24449.0	15794.0	12300.0	10676.0	6.0	29.0	CALIFORNIA	29109.0	...	
837	Marin	18825.0	14871.0	11099.0	7072.0	6346.0	6.0	41.0	CALIFORNIA	29113.0	...	

838 rows × 28 columns

In [11]:
```python
merged_df.isnull().sum()
```

Out[11]:
```
County                                 0
AGE65_69                               0
AGE70_74                               0
AGE75_79                               0
AGE80_84                               0
AGE85_over                             0
State_Code                             0
County_Code                            0
State                                  0
ANSI_Code                              0
Total                                  0
Aged                                   0
Blind_and_disabled                     0
Age_Under18                            0
Age_18-64                              0
Age_65_or_older                        0
SSI_recipients_also_receiving_OASDI    0
Amount_of_payments                     0
Violent_crime                          0
Murder                                 0
Rape                                   0
Robbery                                0
Aggravated_assault                     0
Property_crime                         0
Burglary                               0
Larceny_theft                          0
Motor_vehicle_theft                    0
Arson                                  0
dtype: int64
```

In [12]:
```python
merged_df.shape
```

Out[12]:
```
(838, 28)
```

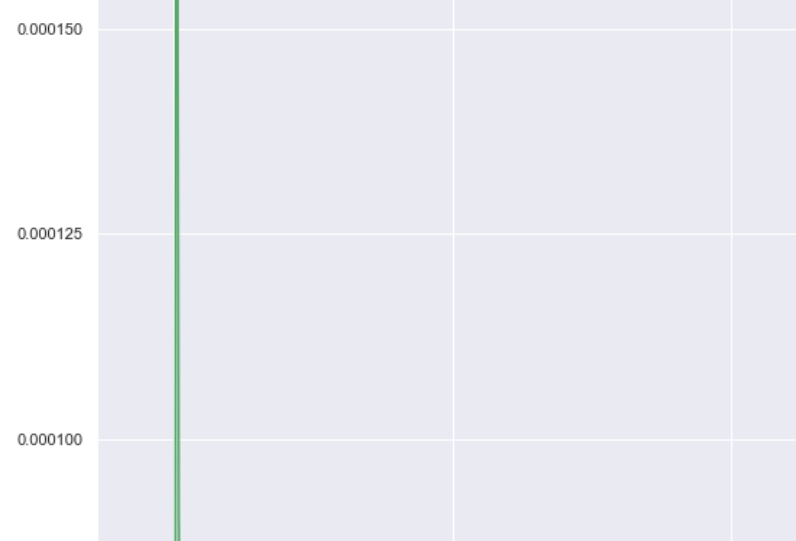
In [13]:
```python
merged_df.to_sql("merged_table", conn, if_exists='append', index=False)
c = conn.cursor()
c.execute("select * from merged_table LIMIT 10")
print(tuple(d[0] for d in c.description))
for row in c:
    print(row)
```

```
('County', 'AGE65_69', 'AGE70_74', 'AGE75_79', 'AGE80_84', 'AGE85_over', 'State_Code', 'County_Code', 'State', 'ANSI_Code', 'Total', 'Aged', 'Blind_and_disabled', 'Age_Under18', 'Age_18-64', 'Age_65_or_older', 'SSI_recipients_also_receiving_OASDI', 'Amount_of_payments', 'Violent_crime', 'Murder', 'Rape', 'Robbery', 'Aggravated_assault', 'Property_crime', 'Burglary', 'Larceny_theft', 'Motor_vehicle_theft', 'Arson')
('Shelby ', 11940.0, 8412.0, 6232.0, 3686.0, 3025.0, 1.0, 117.0, ' ALABAMA', 1001.0, 1439.0, 60.0, 1379.0, 200.0, 1027.0, 212.0, 509.0, 826.0, 51.0, 0.0, 6.0, 5.0, 40.0, 372.0, 92.0, 240, 40.0, 3.0)
('Talladega ', 4531.0, 4226.0, 2482.0, 1910.0, 1273.0, 1.0, 121.0, ' ALABAMA', 1003.0, 3505.0, 200.0, 3305.0, 599.0, 2345.0, 561.0, 1297.0, 1917.0, 223.0, 0.0, 9.0, 37.0, 177.0, 615.0, 173.0, 397, 45.0, 3.0)
('Tuscaloosa ', 9310.0, 7501.0, 5411.0, 2713.0, 2537.0, 1.0, 125.0, ' ALABAMA', 1005.0, 1395.0, 100.0, 1295.0, 188.0, 883.0, 324.0, 557.0, 780.0, 375.0, 1.0, 19.0, 5.0, 350.0, 796.0, 191.0, 492, 113.0, 3.0)
('Pinal ', 27844.0, 27162.0, 18482.0, 11173.0, 6164.0, 4.0, 21.0, ' ARIZONA', 1007.0, 896.0, 28.0, 868.0, 74.0, 665.0, 157.0, 368.0, 469.0, 14.0, 0.0, 5.0, 7.0, 2.0, 144.0, 49.0, 95, 0.0, 3.0)
('Mendocino ', 6576.0, 5744.0, 2860.0, 2417.0, 1475.0, 6.0, 45.0, ' CALIFORNIA', 1009.0, 1227.0, 57.0, 1170.0, 115.0, 887.0, 225.0, 484.0, 654.0, 68.0, 4.0, 30.0, 12.0, 22.0, 669.0, 178.0, 427, 64.0, 3.0)
('Orange ', 142280.0, 124254.0, 80776.0, 58499.0, 65417.0, 6.0, 59.0, ' CALIFORNIA', 1011.0, 549.0, 26.0, 523.0, 96.0, 330.0, 123.0, 175.0, 329.0, 99.0, 2.0, 32.0, 1.0, 64.0, 444.0, 152.0, 233, 59.0, 3.0)
('Sacramento ', 70410.0, 57822.0, 36056.0, 25650.0, 27506.0, 6.0, 67.0, ' CALIFORNIA', 1013.0, 1041.0, 61.0, 980.0, 112.0, 688.0, 241.0, 423.0, 560.0, 40.0, 1.0, 7.0, 2.0, 30.0, 202.0, 74.0, 112, 16.0, 3.0)
('Citrus ', 16405.0, 13054.0, 10321.0, 7242.0, 6810.0, 12.0, 17.0, ' FLORIDA', 1015.0, 4532.0, 152.0, 4380.0, 552.0, 3358.0, 622.0, 1624.0, 2637.0, 31.0, 0.0, 2.0, 1.0, 28.0, 123.0, 45.0, 60, 18.0, 3.0)
('Peoria ', 9622.0, 7828.0, 4479.0, 3712.0, 4560.0, 17.0, 143.0, ' ILLINOIS', 1017.0, 1422.0, 101.0, 1321.0, 165.0, 940.0, 317.0, 520.0, 768.0, 121.0, 1.0, 9.0, 19.0, 92.0, 657.0, 187.0, 400, 70.0, 3.0)
('Hancock ', 3932.0, 4173.0, 2182.0, 1487.0, 1174.0, 18.0, 59.0, ' INDIANA', 1019.0, 830.0, 35.0, 795.0, 67.0, 627.0, 136.0, 296.0, 448.0, 729.0, 10.0, 45.0, 128.0, 546.0, 2255.0, 1181.0, 888, 186.0, 3.0)
```

Visualizing Data

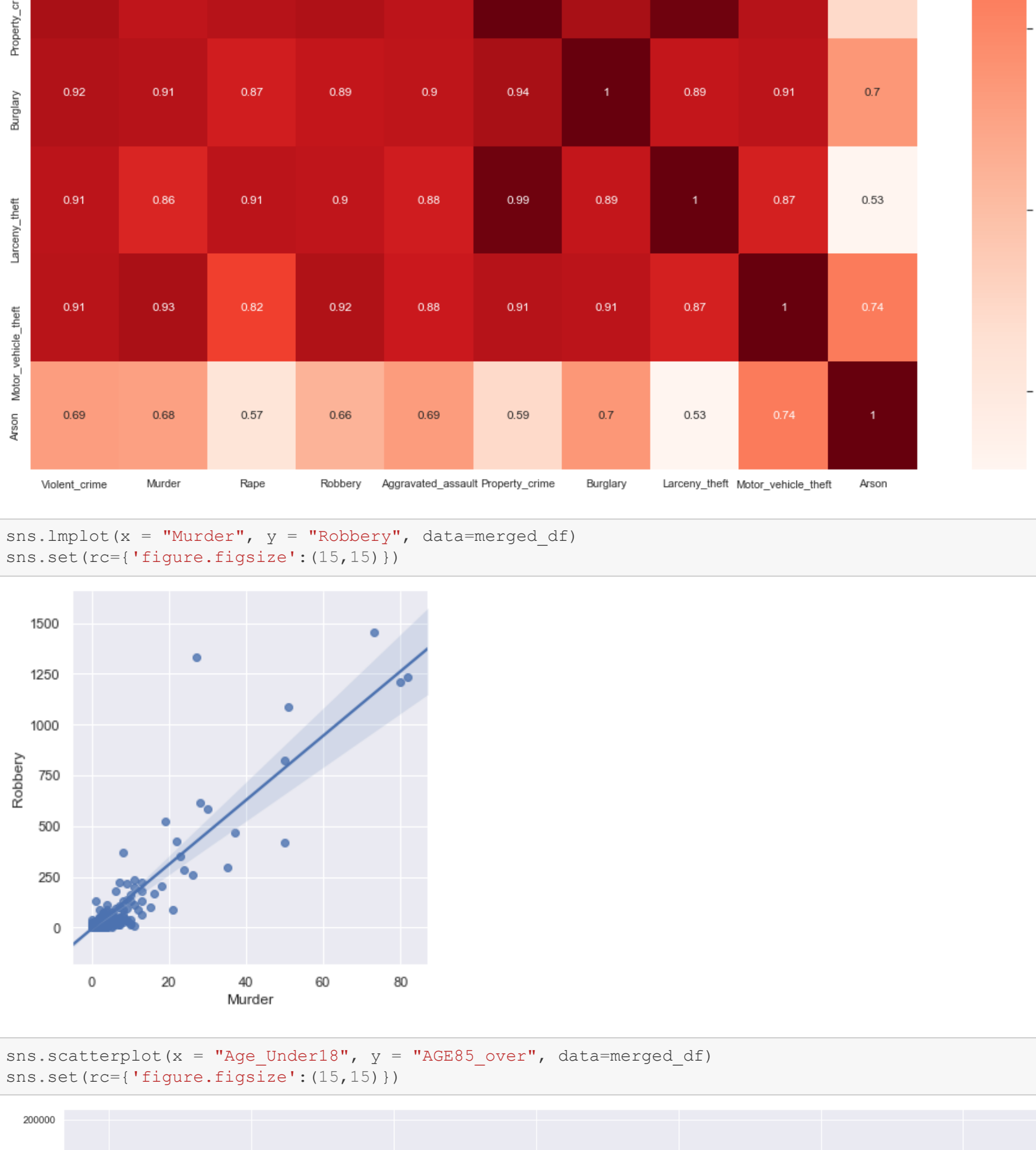
In [14]:
```python
import seaborn as sns
```

In [15]:
```python
import matplotlib.pyplot as plt
for col in merged_df[['AGE65_69', 'AGE70_74', 'AGE75_79', 'AGE80_84', 'AGE85_over']]:
    sns.kdeplot(merged_df[col], shade=True)
sns.set(rc={'figure.figsize':(20,20)})
```

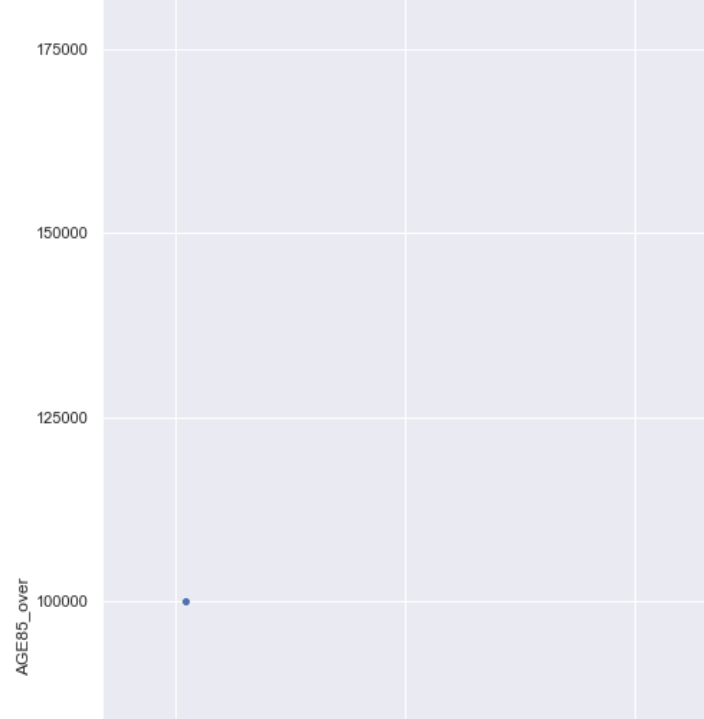


In [16]:
```python
for col in merged_df[['Aged', 'Blind_and_disabled', 'Age_Under18', 'Age_18-64', 'Age_65_or_older',
                     'SSI_recipients_also_receiving_OASDI', 'Amount_of_payments']]:
    sns.kdeplot(merged_df[col], shade=True)
sns.set(rc={'figure.figsize':(20,20)})
```

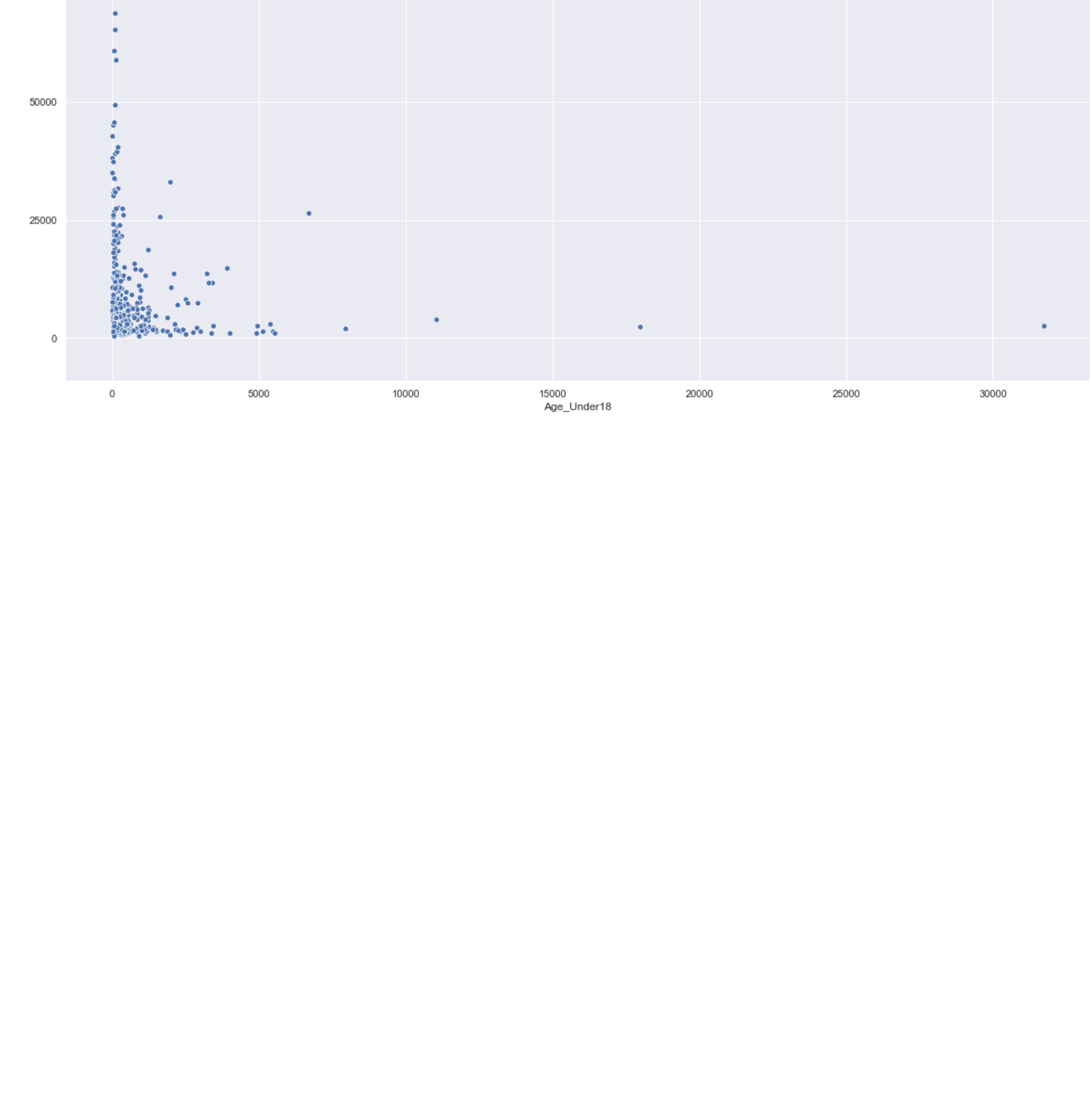
In [17]:
```python
# Pearson Correlatio
data = merged_df[['Violent_crime', 'Murder', 'Rape', 'Robbery', 'Aggravated_assault', 'Property_crime',
 'Burglary', 'Larceny_theft', 'Motor_vehicle_theft', 'Arson']]
cor = data.corr()
sns.heatmap(cor, annot=True, cmap=plt.cm.Reds)
sns.set(rc={'figure.figsize':(20,20)})
```



In [18]:
```python
sns.lmplot(x = "Murder", y = "Robbery", data=merged_df)
sns.set(rc={'figure.figsize':(15,15)})
```

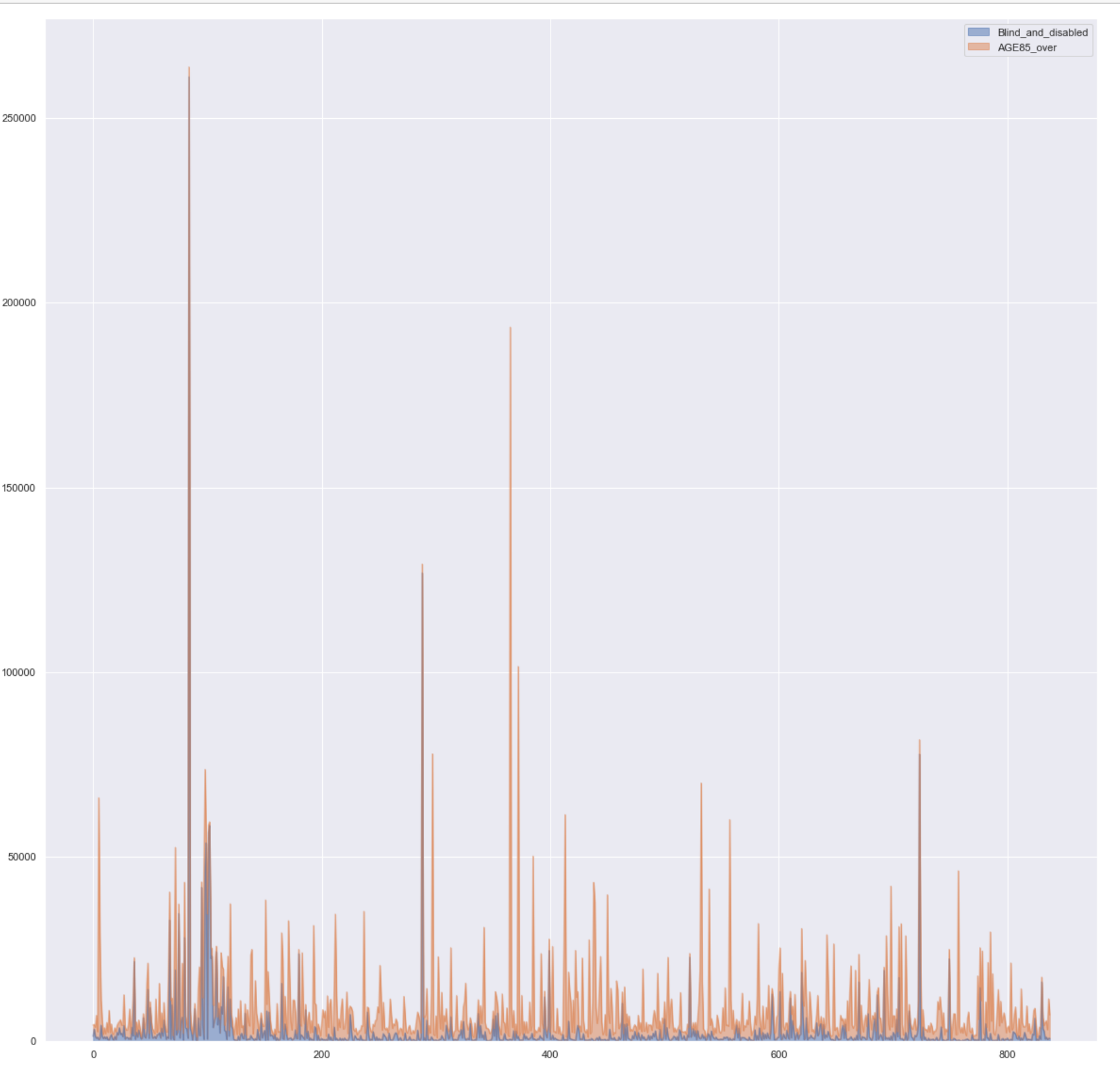


In [19]:
```python
sns.scatterplot(x = "Age_Under18", y = "AGE85_over", data=merged_df)
sns.set(rc={'figure.figsize':(15,15)})
```



In [21]:

```
data1 = merged_df[["Blind_and_disabled","AGE85_over"]]
data1.plot.area(alpha=0.5)
sns.set(rc={'figure.figsize':(15,15)})
```



END