In [4]:
```
from pandas import Series, DataFrame

import numpy as np

import string
import re
import matplotlib.pyplot as plt
from matplotlib.pyplot import rcParams

%matplotlib inline
from collections import Counter
```

In [5]:
```
import keras
```

```
Using TensorFlow backend.
C:\Users\orlan\anaconda3\lib\site-packages\tensorflow\python\framework\dtype
s.py:526: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is
deprecated; in a future version of numpy, it will be understood as (type,
(1,)) / '(1,)type'.
  _np_qint8 = np.dtype([("qint8", np.int8, 1)])
C:\Users\orlan\anaconda3\lib\site-packages\tensorflow\python\framework\dtype
s.py:527: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is
deprecated; in a future version of numpy, it will be understood as (type,
(1,)) / '(1,)type'.
  _np_quint8 = np.dtype([("quint8", np.uint8, 1)])
C:\Users\orlan\anaconda3\lib\site-packages\tensorflow\python\framework\dtype
s.py:528: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is
deprecated; in a future version of numpy, it will be understood as (type,
(1,)) / '(1,)type'.
  _np_qint16 = np.dtype([("qint16", np.int16, 1)])
C:\Users\orlan\anaconda3\lib\site-packages\tensorflow\python\framework\dtype
s.py:529: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is
deprecated; in a future version of numpy, it will be understood as (type,
(1,)) / '(1,)type'.
  _np_quint16 = np.dtype([("quint16", np.uint16, 1)])
C:\Users\orlan\anaconda3\lib\site-packages\tensorflow\python\framework\dtype
s.py:530: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is
deprecated; in a future version of numpy, it will be understood as (type,
(1,)) / '(1,)type'.
  _np_qint32 = np.dtype([("qint32", np.int32, 1)])
C:\Users\orlan\anaconda3\lib\site-packages\tensorflow\python\framework\dtype
s.py:535: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is
deprecated; in a future version of numpy, it will be understood as (type,
(1,)) / '(1,)type'.
  np_resource = np.dtype([("resource", np.ubyte, 1)])
```

In [6]:
```
from csv import reader
from datetime import datetime
```

In [7]:
```
import pandas as pd
import json
import sys
import warnings
```

In [8]:
```
import sklearn
from sklearn import datasets, linear_model
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier, BaggingClassifier
from sklearn.linear_model import LinearRegression
```

In [9]:
```
df = ("C:\\Users\\orlan\Median_Household_Income_by_State_by_Race.csv")
```

In [10]:
```
data1 = pd.read_csv(df)
```

In [11]:
```
print (data1)
```

	State	White	African American	American Indian	Asian	Hispanic
0	AL	55265	32188	48188	62639	55511
1	AK	84799	58209	49834	69685	63478
2	AZ	58435	45310	35251	74430	59806
3	AR	49581	30758	37801	65919	35947
4	CA	74276	49334	53019	91623	72432
5	CO	71221	49634	46082	73189	66528
6	CT	82950	47856	41094	93665	57214
7	DE	70154	48297	45227	96657	54239
8	DC	132040	43564	42788	103898	120500
9	FL	56008	39586	45307	68777	52087
10	GA	63543	42085	38167	77008	39182
11	HI	77486	70100	47298	83089	62348
12	ID	53738	39970	40898	53435	39051
13	IL	69194	37244	47573	85828	53958
14	IN	57269	33342	41801	62136	50789
15	IA	60123	31053	36941	60638	42077
16	KS	59641	35829	43943	68821	51903
17	KY	50267	33642	33200	62655	34885
18	LA	58632	29508	41664	61351	56667
19	ME	56030	38655	32670	55656	70500
20	MD	90964	65039	69955	102786	89265
21	MA	81977	48382	42686	91713	79919
22	MI	59077	33649	42336	82733	58547
23	MN	71415	34879	36429	75437	72917
24	MS	54244	29690	33297	59478	26287
25	MO	56701	35710	40824	67526	48696
26	MT	53813	41484	32072	58774	36380
27	NE	61342	34122	37398	56627	42063
28	NV	61412	39726	41478	65460	57109
29	NH	74468	51630	47902	83301	49014
30	NJ	85423	51309	53507	116131	48429
31	NM	50947	38490	33552	65019	47311
32	NY	73584	46178	41267	72131	47227
33	NC	58171	37242	38206	80500	48398
34	ND	66213	34565	36710	62223	48715
35	OH	58885	31669	33682	73058	47894
36	OK	54612	34138	42820	56996	40486
37	OR	60183	37078	42047	75929	60028
38	PA	63110	36847	37702	72699	51705
39	RI	67362	41630	34414	72907	41155
40	SC	58825	33371	39484	61898	35280
41	SD	59465	31957	27045	51288	38958
42	TN	54085	36683	43212	72881	26466
43	TX	62679	44688	52094	84851	53659
44	UT	70199	42739	41942	70759	64594
45	VT	60577	43548	41793	55568	54258
46	VA	76860	49273	61850	102735	79474
47	WA	71466	50487	45558	90131	65024
48	WV	45467	32070	29927	58521	60318
49	WI	61974	30002	41594	66408	38836
50	WY	63116	44712	49352	54025	137572

In [12]:
```
data1.describe()
```

Out[12]:

	White	African American	American Indian	Asian	Hispanic
count	51.000000	51.000000	51.000000	51.000000	51.000000
mean	65279.764706	40767.666667	41978.058824	73168.470588	55589.921569
std	13971.377871	8882.822296	7707.701710	15068.651369	20113.193127
min	45467.000000	29508.000000	27045.000000	51288.000000	26287.000000
25%	56985.000000	33885.500000	37169.500000	62017.000000	42070.000000
50%	61342.000000	38655.000000	41664.000000	70759.000000	52087.000000
75%	71318.000000	45744.000000	45432.500000	82911.000000	61333.000000
max	132040.000000	70100.000000	69955.000000	116131.000000	137572.000000

In [13]:
```
data1.min()
```

Out[13]:
```
State                AK
White             45467
African American  29508
American Indian   27045
Asian             51288
Hispanic          26287
dtype: object
```

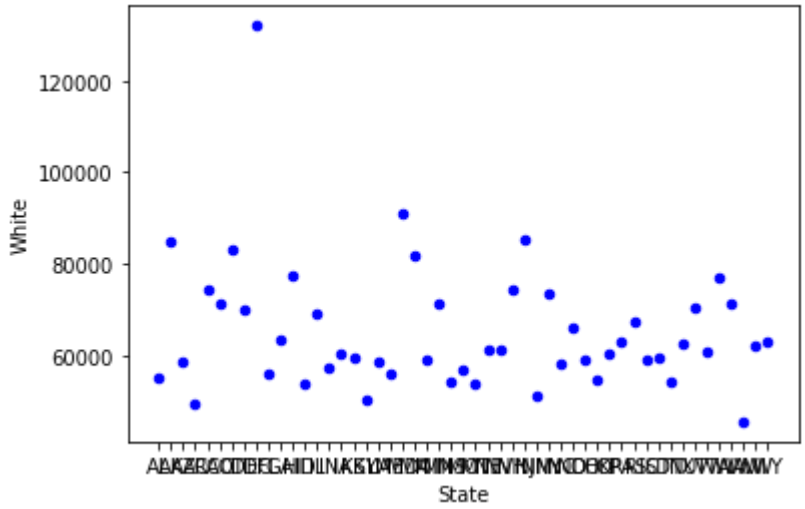
In [14]:
```
data1.max()
```

Out[14]:
```
State                WY
White            132040
African American  70100
American Indian   69955
Asian            116131
Hispanic         137572
dtype: object
```

In [15]:
```
import scipy
import math
```

In [16]:
```
data1.plot(kind='scatter', x='State', y='White', c=['blue'])
```

Out[16]: <matplotlib.axes._subplots.AxesSubplot at 0x29da2cfe788>

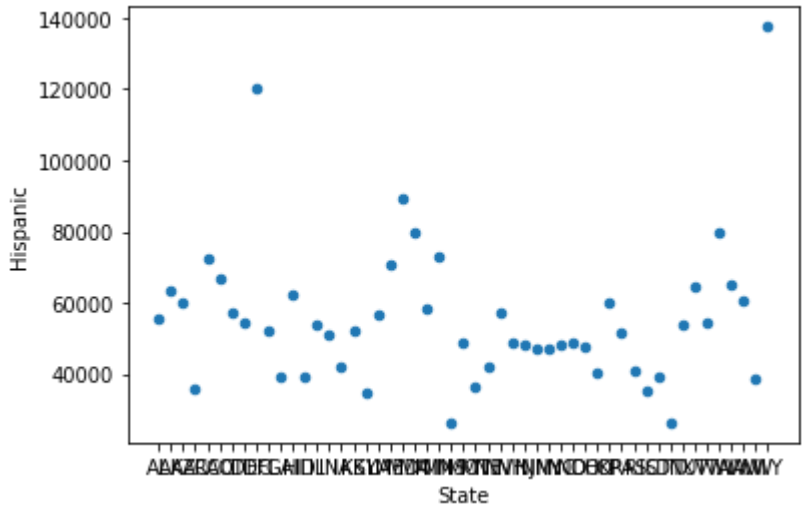


In [ ]:
```

```

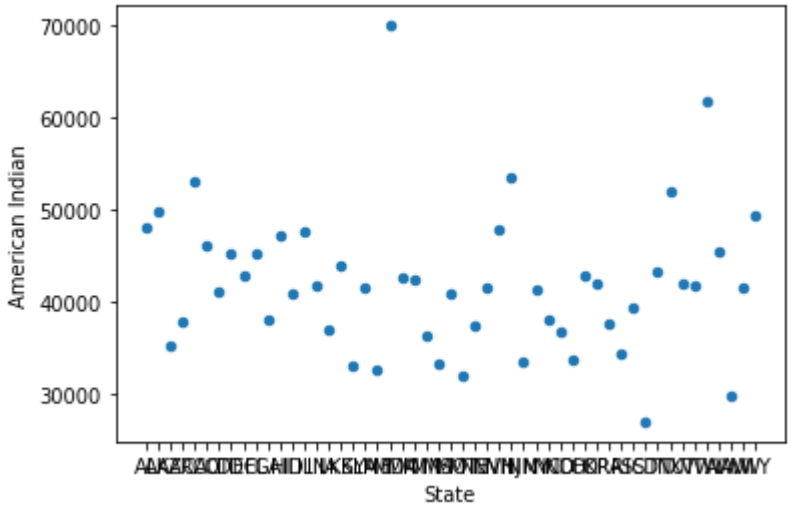
In [17]:
```
data1.plot(kind='scatter', x='State', y='Hispanic')
```

Out[17]: <matplotlib.axes._subplots.AxesSubplot at 0x29da2e5ff08>



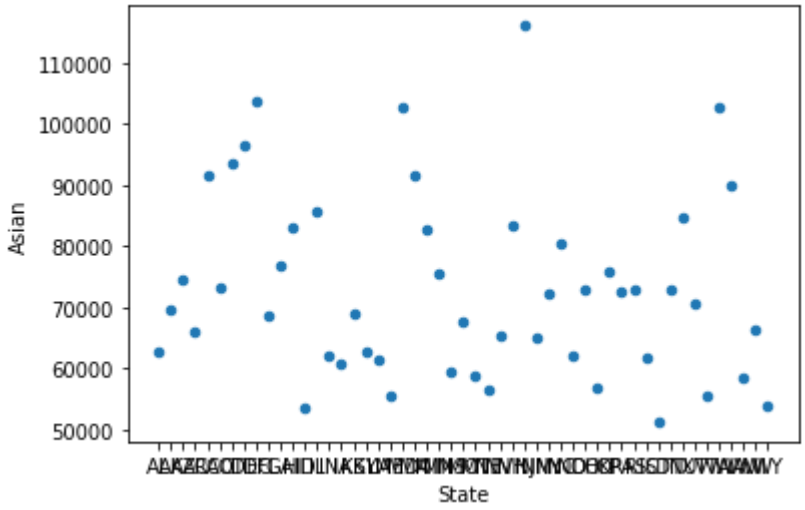
In [18]:
```
data1.plot(kind='scatter', x='State', y='American Indian')
```

Out[18]: <matplotlib.axes._subplots.AxesSubplot at 0x29da2e933c8>



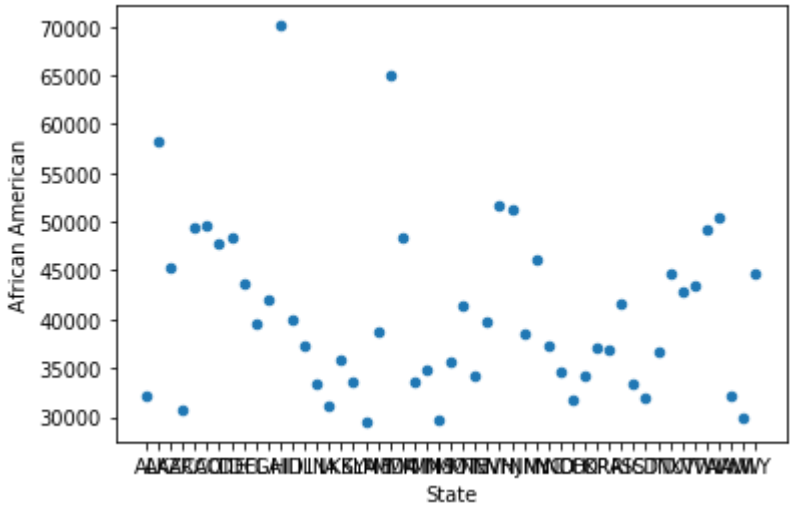
In [19]:
```
data1.plot(kind='scatter', x='State', y='Asian')
```

Out[19]: <matplotlib.axes._subplots.AxesSubplot at 0x29da3046408>



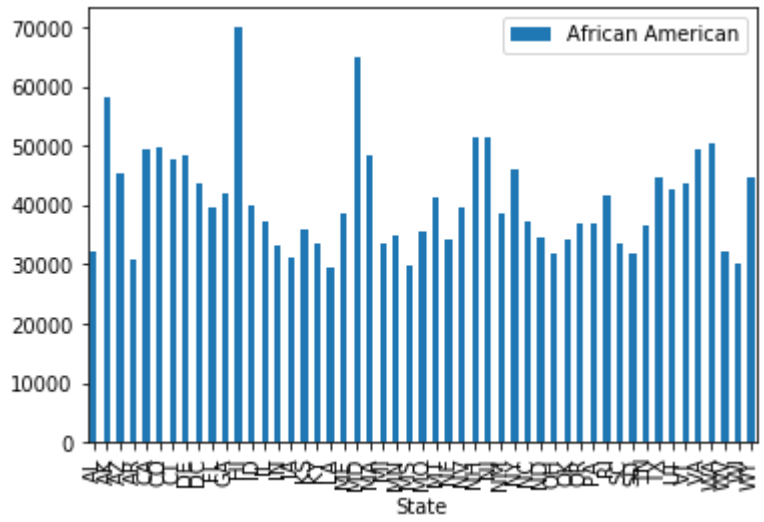
In [20]:
```
data1.plot(kind='scatter', x='State', y='African American')
```

Out[20]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x29da2bef848>
```



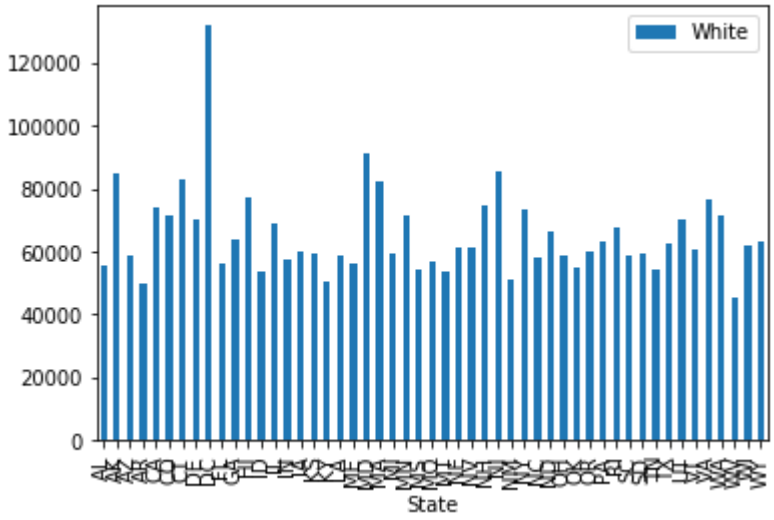
In [21]:
```
data1.plot(kind='bar', x='State', y='African American')
```

Out[21]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x29da4109148>
```



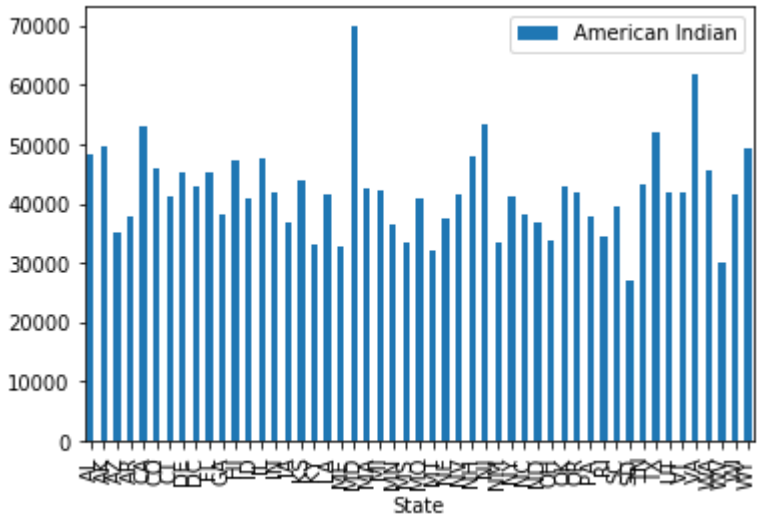
In [22]:
```
data1.plot(kind='bar', x='State', y='White')
```

Out[22]: <matplotlib.axes._subplots.AxesSubplot at 0x29da29c22c8>



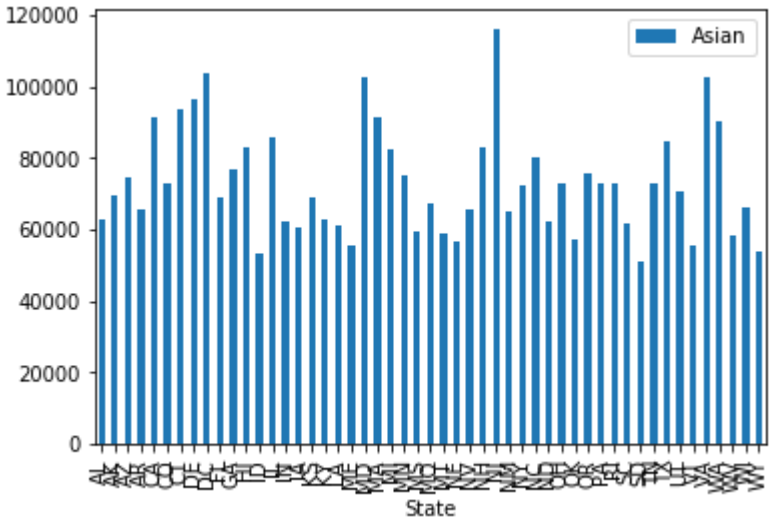
In [23]:
```
data1.plot(kind='bar', x='State', y='American Indian')
```

Out[23]: <matplotlib.axes._subplots.AxesSubplot at 0x29da44c2bc8>



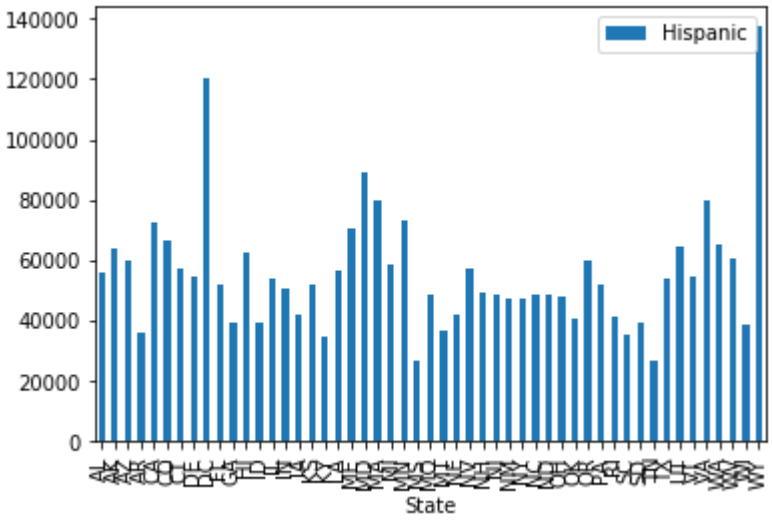
In [24]:
```
data1.plot(kind='bar', x='State', y='Asian')
```

Out[24]: <matplotlib.axes._subplots.AxesSubplot at 0x29da4585d48>



In [25]:
```
data1.plot(kind='bar', x='State', y='Hispanic')
```

Out[25]: <matplotlib.axes._subplots.AxesSubplot at 0x29da4774108>



In [26]:
```
import scipy
import math
```

In [27]:
```
linreg = LinearRegression()
```

In [35]:
```
White = np.array([55265,84799,58435,49581,74276,71221,82950,70154,132040,56008
,63543,77486,53738,69194,57269,60123,59641,50267,58632,56030,90964,81977,59077
,71415,54244,56701,53813,61342,61412,74468,85423,50947,73584,58171,66213,58885
,54612,60183,63110,67362,58825,59465,54085,62679,70199,60577,76860,71466,45467
,61974,63116])
African_American = np.array([32188,58209,45310,30758,49334,49634,47856,48297,4
3564,39586,42085,70100,39970,37244,33342,31053,35829,33642,29508,38655,65039,4
8382,33649,34879,29690,35710,41484,34122,39726,51630,51309,38490,46178,37242,3
4565,31669,34138,37078,36847,41630,33371,31957,36683,44688,42739,43548,49273,5
0487,32070,30002,44712])
American_Indian = np.array([48188,49834,35251,37801,53019,46082,41094,45227,42
788,45307,38167,47298,40898,47573,41801,36941,43943,33200,41664,32670,69955,42
686,42336,36429,33297,40824,32072,37398,41478,47902,53507,33552,41267,38206,36
710,33682,42820,42047,37702,34414,39484,27045,43212,52094,41942,41793,61850,45
558,29927,41594,49352])
Asian = np.array([62639,69685,74430,65919,91623,73189,93665,96657,103898,68777
,77008,83089,53435,85828,62136,60638,68821,62655,61351,55656,102786,91713,8273
3,75437,59478,67526,58774,56627,65460,83301,116131,65019,72131,80500,62223,730
58,56996,75929,72699,72907,61898,51288,72881,84851,70759,55568,102735,90131,58
521,66408,54025])
Hispanic = np.array([55511,63478,59806,35947,72432,66528,57214,54239,120500,52
087,39182,62348,39051,53958,50789,42077,51903,34885,56667,70500,89265,79919,58
547,72917,26287,48696,36380,42063,57109,49014,48429,47311,47227,48398,48715,47
894,40486,60028,51705,41155,35280,38958,26466,53659,64594,54258,79474,65024,60
318,38836,137572])
```

In [36]:
```
White = White.reshape(-1, 1)
```

In [37]:
```
linreg.fit(White, African_American)
```

Out[37]:
```
LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None, normalize=Fals
e)
```

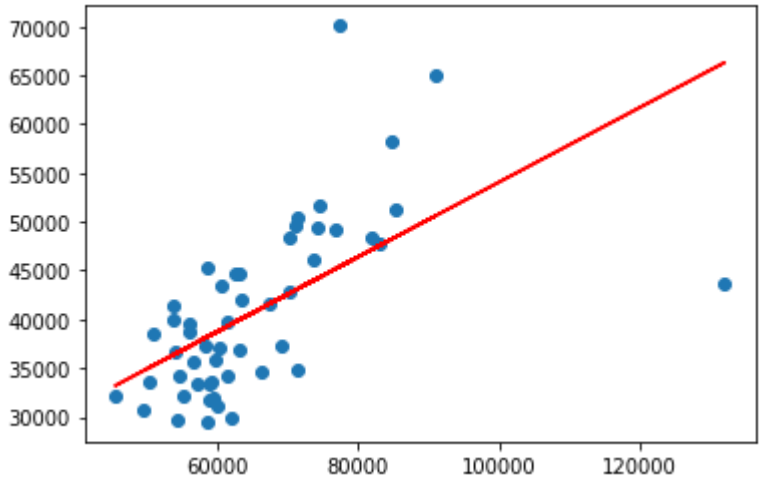
In [38]:
```
African_American_pred = linreg.predict(White)
```

In [39]:
```
American_Indian_pred = linreg.predict(White)
```

In [40]:
```
Asian_pred = linreg.predict(White)
```

In [41]:
```
Hispanic_pred = linreg.predict(White)
```

In [45]:
```
plt.scatter(White,African_American)
plt.plot(White, African_American_pred, color='red')
plt.show()
```



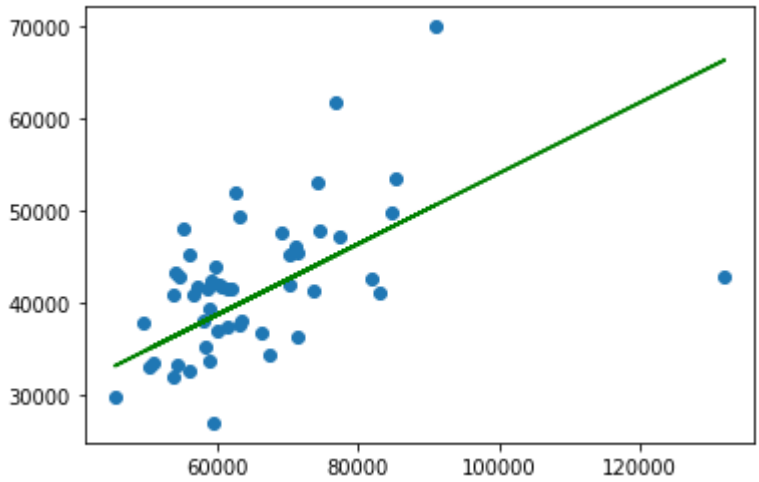
In [47]:
```
print(linreg.coef_)
```

```
[0.38258542]
```

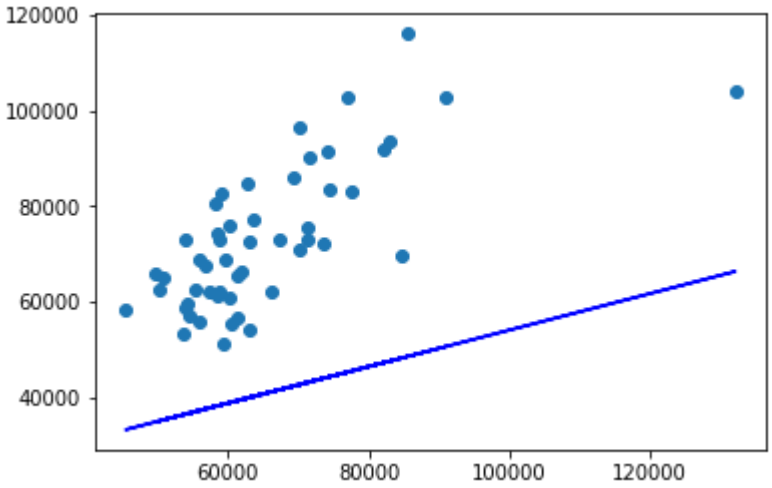
In [48]:
```
print(linreg.intercept_)
```

```
15792.580599162182
```

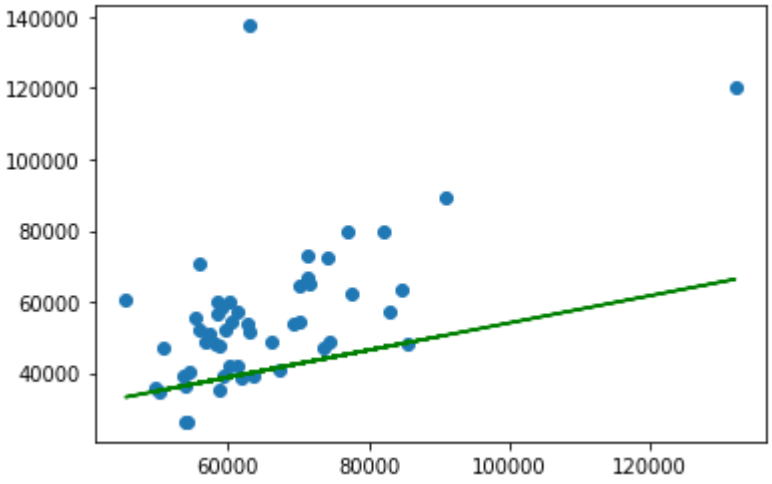
In [46]:
```
plt.scatter(White,American_Indian)
plt.plot(White, American_Indian_pred, color='green')
plt.show()
```



In [52]:
```
plt.scatter(White,Asian)
plt.plot(White, Asian_pred, color='blue')
plt.show()
```



In [53]:
```
plt.scatter(White,Hispanic)
plt.plot(White, Hispanic_pred, color='green')
plt.show()
```



In [ ]: