```
                                               # read the data
                                               import pandas as pd
In [1]: In [2]:                                import numpy as np
                                               import seaborn as sns
                                               import matplotlib.pyplot as plt


                                               file_path="C:\\Users\\omkar\\OneDrive\\Do
                                               cuments\\Data science\\Naresh IT\\
                                               visa_df=pd.read_csv(file_path)
# import the packages                          visa_df
```

Out[2]: **case_id continent education_of_employee has_job_experience requires_job_traini** **0** EZYV01 Asia High

School N

1 EZYV02 Asia Master's Y

2 EZYV03 Asia Bachelor's N

3 EZYV04 Asia Bachelor's N

4 EZYV05 Africa Master's Y

... ...

25475 EZYV25476 Asia Bachelor's Y

25476 EZYV25477 Asia High School Y

25477 EZYV25478 Asia Master's Y

25478 EZYV25479 Asia Master's Y

25479 EZYV25480 Asia Bachelor's Y

25480 rows × 12 columns

```
                      value counts

In [ ]: In [4]:       visa_df['continent'].

# Continent colums    value_counts()
```

```
Out[4]: continent
         Asia 16861
         Europe 3732
         North America 3292
         South America 852
         Africa 551
         Oceania 192
         Name: count, dtype: int64
```

```
                       visa_df['case_status']
                       .value_counts()
In [5]:
```

```
Out[5]: case_status
         Certified 17018
         Denied 8462
         Name: count, dtype: int64
```

```
applicants how many got Visa
```

In [ ]: In [7]:

```
#Q) out of all Asian
applicants how many got Visa
# Out of all Europe
```

```
con1=visa_df['continent']=='A
sia'
con2=visa_df['case_status']==
'Certified' con=con1&con2
len(visa_df[con])
```

Out[7]: 11012

In [10]:

```
visa_df['continent'].uniq
ue()
visa_df['continent'].valu
e_counts().keys()
```

```
Out[10]: Index(['Asia', 'Europe', 'North America', 'South America', 'Africa',
          'Oceania'],
           dtype='object', name='continent')
```

In [22]:

```
# Generalised
lables=visa_df['continent'].uniq
ue()
certified_count=[]
denied_count=[]
for i in lables:
 con1=visa_df['continent']==i

con2=visa_df['case_status']=='Ce
rtified'
```

```
con3=visa_df['case_status']=='De
nied'

certified_count.append(len(visa_
df[con1&con2]))
denied_count.append(len(visa_df[
con1&con3]))

pd.DataFrame(zip(lables,certifie
d_count,denied_count),
columns=['continent','certified'
,'denied'])
```

Out[22]:

	continent	certified	denied
0	Asia	11012	5849
1	Africa	397	154
2	North America	2037	1255
3	Europe	2957	775
4	South America	493	359
5	Oceania	122	70

In [23]:

```
pd.DataFrame(zip(lables,certified_count,de
nied_count),
columns=['continent','certified','denied']
```

```
).set_index('contine
```

Out[23]:

```
certified denied continent

Asia 11012 5849

Africa 397 154

North America 2037 1255

Europe 2957 775

South America 493 359

Oceania 122 70
```

����. ����������������

In [28]:

```
col1=visa_df['conti
nent']
col2=visa_df['case_
status']
result1=pd.crosstab
(col1,col2) result1
```

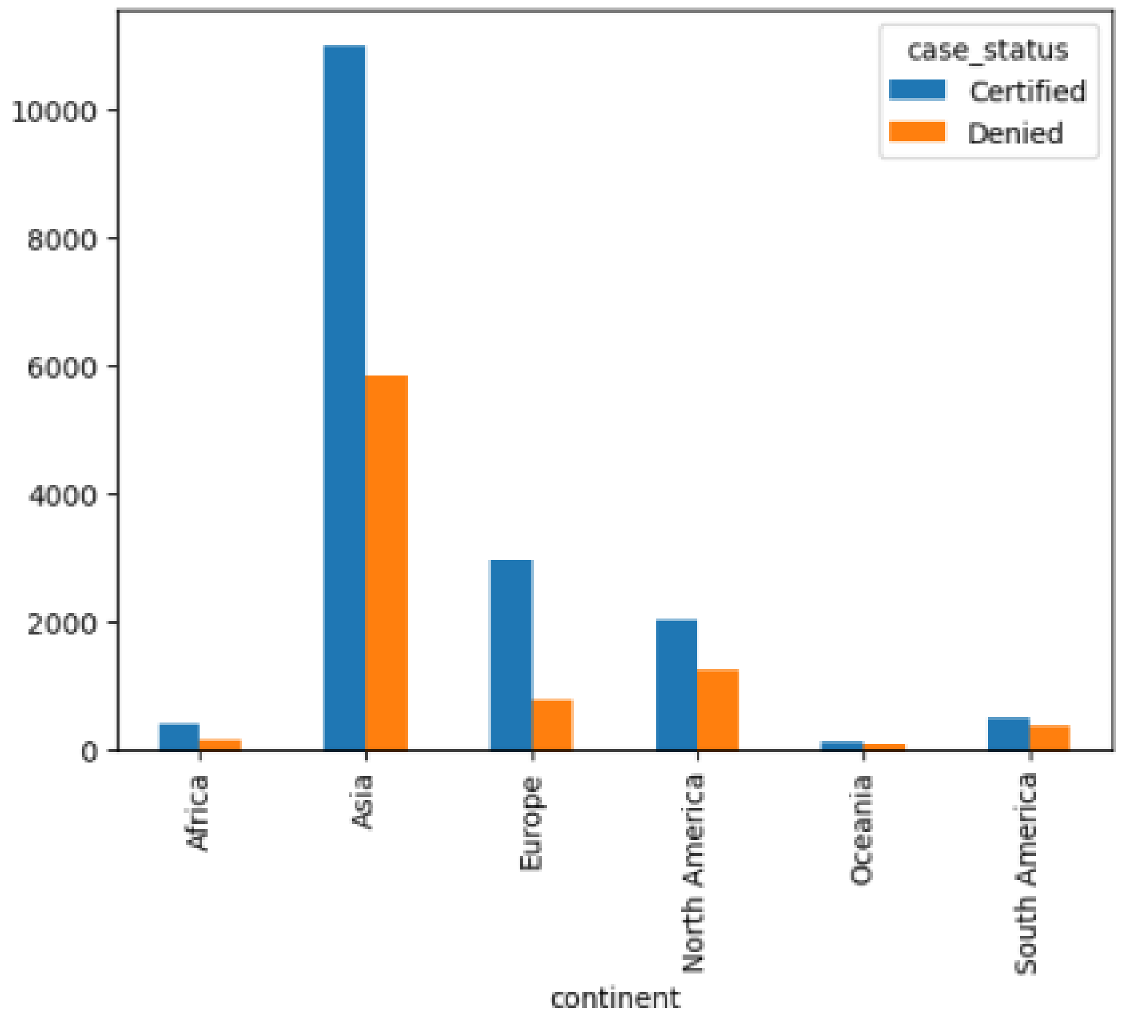
Out[28]:

```
case_status Certified Denied

continent

Africa 397 154

Asia 11012 5849

Europe 2957 775

North America 2037 1255

Oceania 122 70

South America 493 359
```

In [29]:

```
result1.plot(kin
d='bar')
```

```
Out[29]: <Axes: xlabel='continent'>
```



```
In [36]:
#Continent
#Education
#Case status
col1=visa_df['continent
']
col2=visa_df['case_stat
us']
col3=visa_df['education
_of_employee']
col=[col2,col3] #
values
result2=pd.crosstab(col
1,col) result2
```

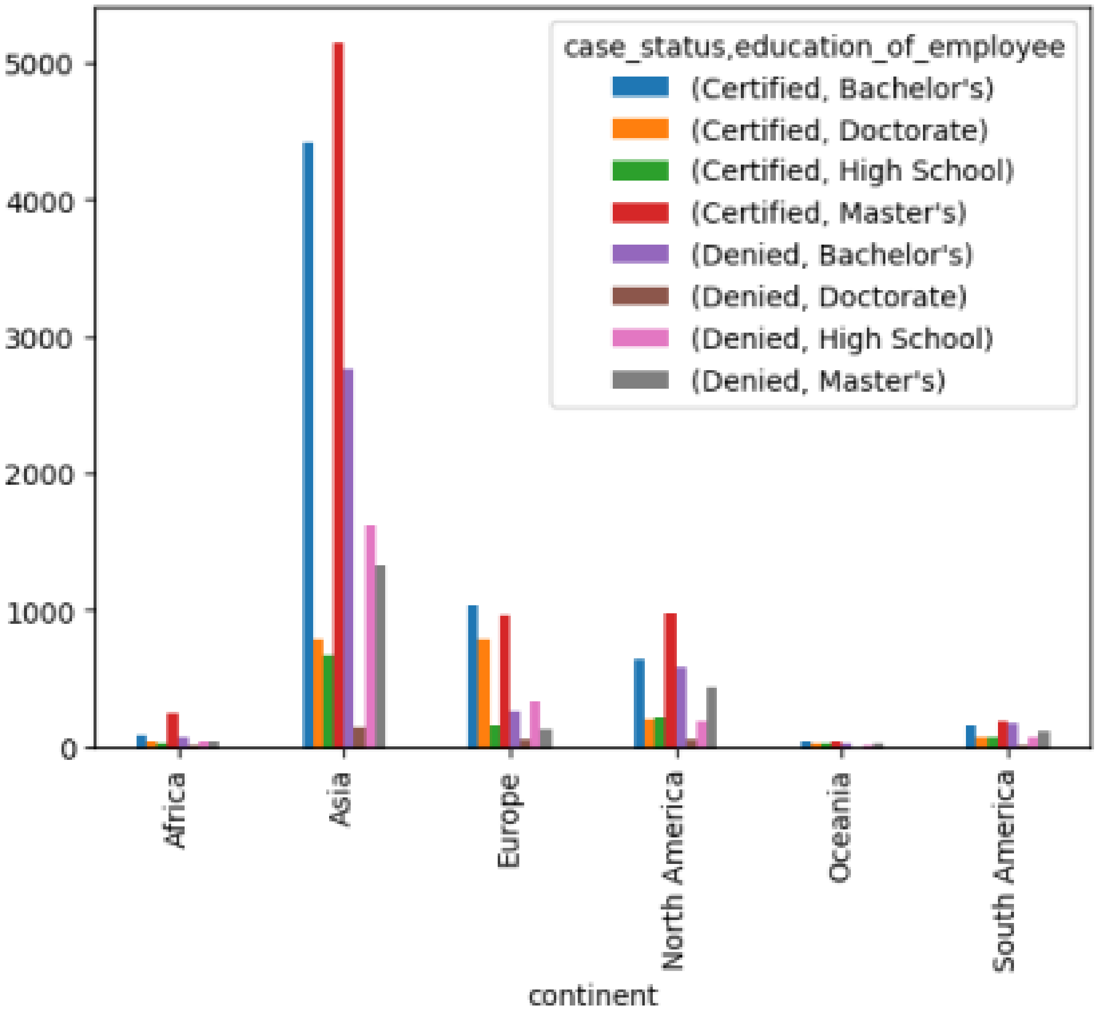
```
Out[36]: case_status Certified

   education_of_employee Bachelor's Doctorate High

                                        School Master's Bachelor's Doctorate Hig
                                                                             Scho

            continent

                                                    Africa 81 43 23 250 62 11 4

                                           Asia 4407 780 676 5149 2761 143 161

                                             Europe 1040 788 162 967 259 58 32

North America 641 207 210 979 584 51 19 Oceania 38 19 19 46 28 3 1

                                     South America 160 75 74 184 173 14 6
```

In [37]:

```python
result2.plot(kind='bar')
```

Out[37]:

```
<Axes: xlabel='continent'>
```

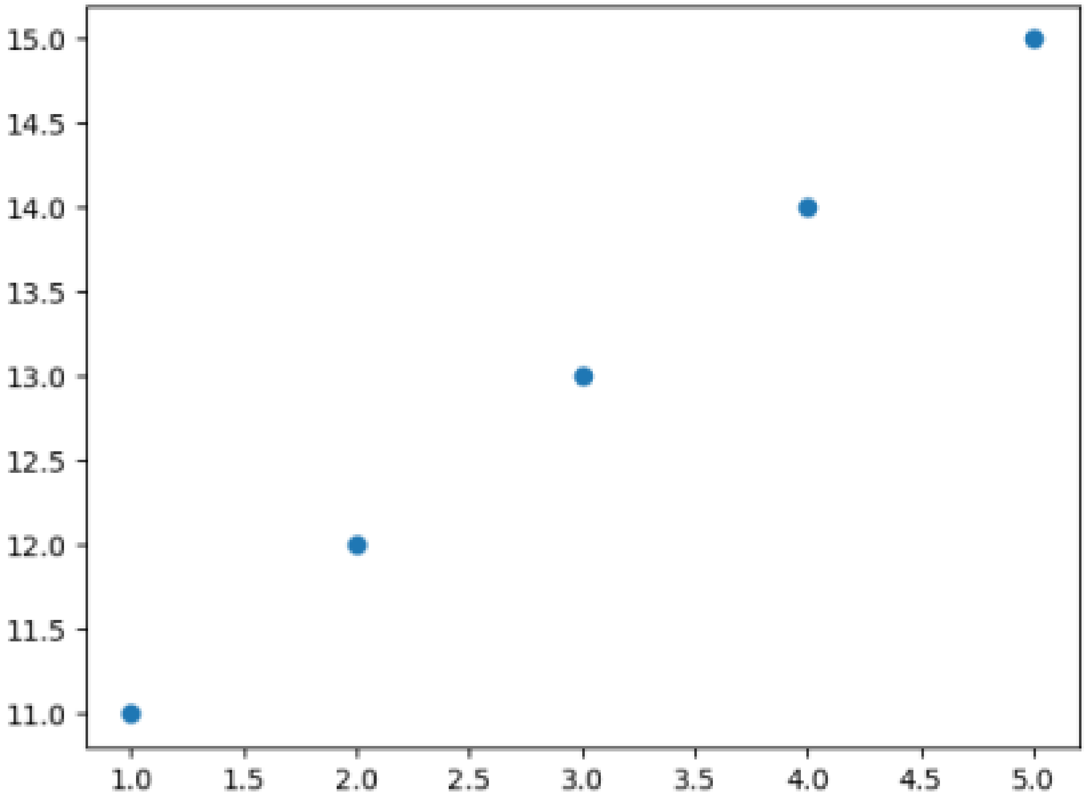


Numerical vs Numerical

������. ��������������

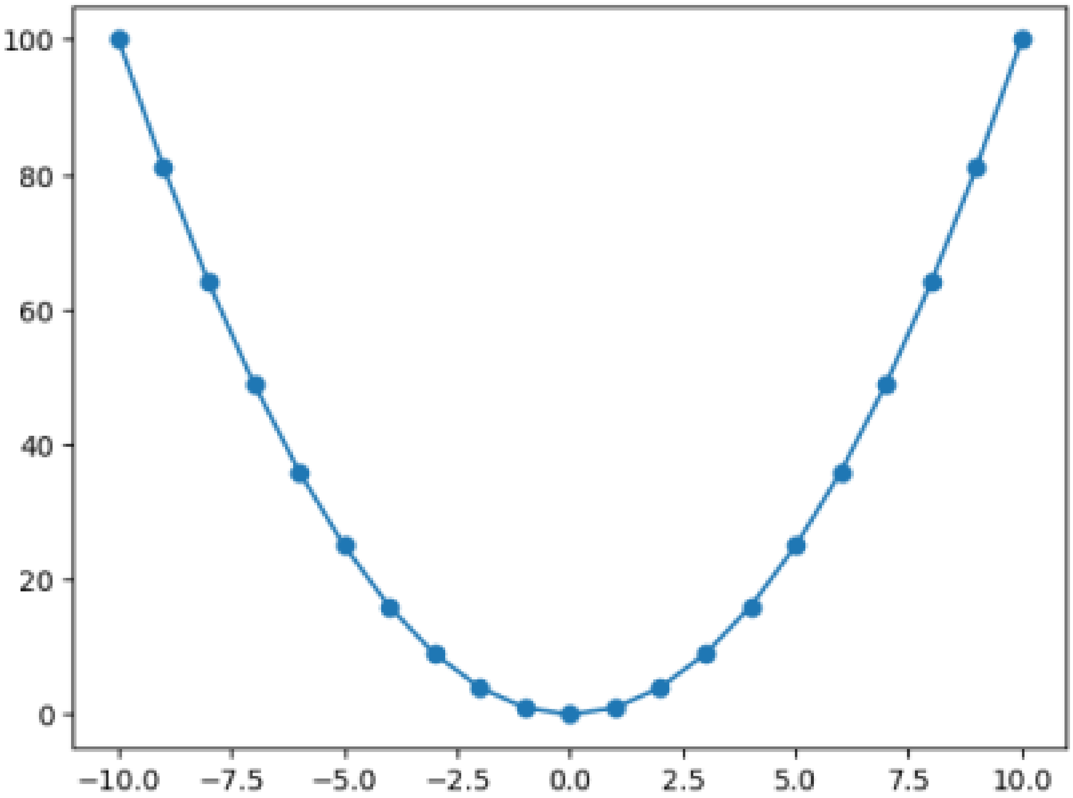
```
In [38]:
x=[1,2,3,4,5]
y=[11,12,13,14,15]
#(1,11),(2,12),(3,13),
(4,14),(5,15)
plt.scatter(x,y)
```

```
Out[38]: <matplotlib.collections.PathCollection at 0x127d5643750>
```



```
                         y=[i*i for i in x]
In [39]:                 x
x=[i for i in
range(-10,11)]

Out[39]: [-10, -9, -8, -7, -6, -5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 1
         0]
                                             [40]:
                                             y
                                       In

Out[40]: [100, 81, 64, 49, 36, 25, 16, 9, 4, 1, 0, 1, 4, 9, 16, 25, 36, 49, 64, 81,
         100]
```

```
                x,y)
In [44]:        plt.plot(x,y
plt.scatter( )

Out[44]: [<matplotlib.lines.Line2D at 0x127d5a68690>]
```



Scatter plots for only numerical analysis

Scatter plots provides an idea , both variables are related or not related

Postivie relation

Increase in the curve

Negative relation

Decrease in the curve

No realtion

Neither increase nor Decrease

In [48]:

```
dtypes=dict(visa_df.dtypes)
num=[i for i in dtypes if dtypes[i]!='O'] num
```

Out[48]:

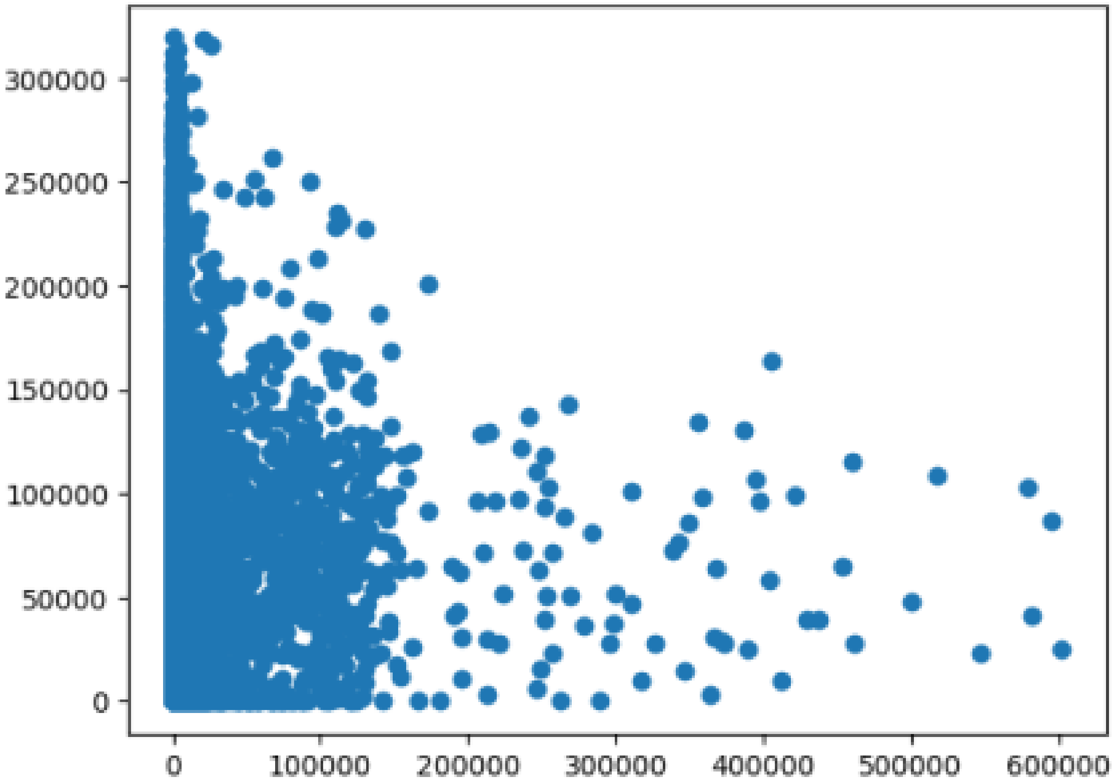
```
['no_of_employees', 'yr_of_estab', 'prevailing_wage']
```

In [49]:

```
col1=visa_df['no_of_employees']
col2=visa_df['prevailing_wage']
plt.scatter(col1,col2)
```

Out[49]:

```
<matplotlib.collections.PathCollection at 0x127d862f3d0>
```



```
In [51]:
#Covariance-matrix

#How many numerical
variables are there : 3
# no_employee yr wage
#no_employee var cov cov

#yr cov var cov #age cov

cov var
```

������������
���������� –
������������
������������

Denoted with r
r range from -1 to 1
postive relation range = (0,1]
negative relation range = [-1,0)
no relation = 0

```
visa_df.corr(numeric_only=True) #
applicable for yo need to see numeric_on

# in the data frame we have both cat and
numerical column # correlation applicable
for only numerical column
# Explicitly mention numeric= True

# If people has pandas old version
# they dont have numeric_only argument
# for them visa_df.corr() works
```

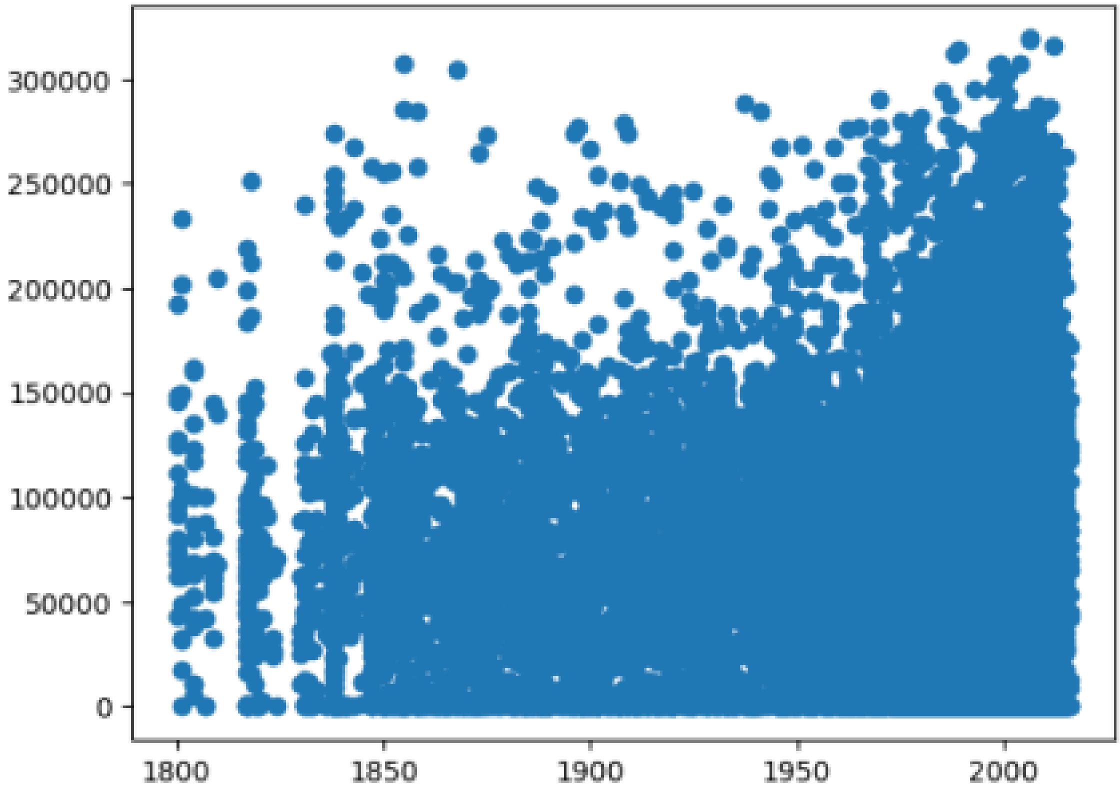
```
In [55]:
��������()
```

```
Out[55]: no_of_employees yr_of_estab prevailing_wage no_of_employees

1.000000 -0.017770 -0.009523 yr_of_estab -0.017770 1.000000 0.012342

prevailing_wage -0.009523 0.012342 1.000000
```

```
In [53]: pd.__version__ # double underscore

Out[53]: '2.0.3'
```

```
In [ ]: #pip unisntall pandas #pip install pandas==2.0.3
```

```
In [56]: plt.scatter(visa_df['yr_of_estab'],
         visa_df['prevailing_wage'])

Out[56]: <matplotlib.collections.PathCollection at 0x127da424610>
```


300000
250000
200000
150000
100000
50000
0
1800
1850
1900
1950
2000

In [ ]: