In [5]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

Read the data

In [6]:

```
path =r'C:\Users\aramaiah.ASUAD\Naresh_IT\MyDataScience\Data_Files\Visadataset.
visa_df=pd.read_csv(path)
visa_df.head(3)
```

Out[6]:

	case_id	continent	education_of_employee	has_job_experience	requires_job_training	no_of_e
0	EZYV01	Asia	High School	N	N	
1	EZYV02	Asia	Master's	Y	N	
2	EZYV03	Asia	Bachelor's	N	Y	

Reading a specific Column

In [7]:

```
visa_df['continent'] #series type
```

Out[7]:

```
0          Asia
1          Asia
2          Asia
3          Asia
4        Africa
          ...  
25475      Asia
25476      Asia
25477      Asia
25478      Asia
25479      Asia
Name: continent, Length: 25480, dtype: object
```

In [9]:
```
visa_df[['continent']] #data frame
```

Out[9]:

	continent
0	Asia
1	Asia
2	Asia
3	Asia
4	Africa
...	...
25475	Asia
25476	Asia
25477	Asia
25478	Asia
25479	Asia

25480 rows × 1 columns

In [10]:
```
visa_df['continent']
```

Out[10]:
```
0          Asia
1          Asia
2          Asia
3          Asia
4        Africa
          ...  
25475      Asia
25476      Asia
25477      Asia
25478      Asia
25479      Asia
Name: continent, Length: 25480, dtype: object
```

In [11]:
```
visa_df.continent
```

Out[11]:
```
0          Asia
1          Asia
2          Asia
3          Asia
4        Africa
          ...  
25475      Asia
25476      Asia
25477      Asia
25478      Asia
25479      Asia
Name: continent, Length: 25480, dtype: object
```

In [14]:
```
visa_df[['continent']] #df
```

Out[14]:

	continent
0	Asia
1	Asia
2	Asia
3	Asia
4	Africa
...	...
25475	Asia
25476	Asia
25477	Asia
25478	Asia
25479	Asia

25480 rows × 1 columns

In [15]:
```
visa_df.columns
```

Out[15]:
```
Index(['case_id', 'continent', 'education_of_employee', 'has_job_experience',
       'requires_job_training', 'no_of_employees', 'yr_of_estab',
       'region_of_employment', 'prevailing_wage', 'unit_of_wage',
       'full_time_position', 'case_status'],
      dtype='object')
```

In [16]:

```
cols=['continent','education_of_employee']
visa_df[cols]
```

Out[16]:

	continent	education_of_employee
0	Asia	High School
1	Asia	Master's
2	Asia	Bachelor's
3	Asia	Bachelor's
4	Africa	Master's
...	...	...
25475	Asia	Bachelor's
25476	Asia	High School
25477	Asia	Master's
25478	Asia	Master's
25479	Asia	Bachelor's

25480 rows × 2 columns

In [17]:

```
visa_df[cols]
```

Out[17]:

	continent	education_of_employee
0	Asia	High School
1	Asia	Master's
2	Asia	Bachelor's
3	Asia	Bachelor's
4	Africa	Master's
...	...	...
25475	Asia	Bachelor's
25476	Asia	High School
25477	Asia	Master's
25478	Asia	Master's
25479	Asia	Bachelor's

25480 rows × 2 columns

In [20]:
```
visa_df.values
# list of all te samples
# list of all the observations
# list of all the tuples
```

Out[20]:
```
array([['EZYV01', 'Asia', 'High School', ..., 'Hour', 'Y', 'Denied'],
       ['EZYV02', 'Asia', "Master's", ..., 'Year', 'Y', 'Certified'],
       ['EZYV03', 'Asia', "Bachelor's", ..., 'Year', 'Y', 'Denied'],
       ...,
       ['EZYV25478', 'Asia', "Master's", ..., 'Year', 'N', 'Certified'],
       ['EZYV25479', 'Asia', "Master's", ..., 'Year', 'Y', 'Certified'],
       ['EZYV25480', 'Asia', "Bachelor's", ..., 'Year', 'Y', 'Certified']],
      dtype=object)
```

In [ ]:
```
# if i give list ==== df
# if i give df === list
```

continent

In [148]:
```
l1=[1,2,3]
l2=['a','b','c']
l=[l1,l2]
l

# pd.DataFrame(l).values
# pd.DataFrame(l)
pd.DataFrame(l).keys()
# l1=continent_vc.keys()
```

Out[148]:
```
RangeIndex(start=0, stop=3, step=1)
```

In [22]:
```
pd.DataFrame(l)
```

Out[22]:

	0	1	2
0	1	2	3
1	a	b	c

In [23]:
```
col=['continent']
visa_df[col]
```

Out[23]:

	continent
0	Asia
1	Asia
2	Asia
3	Asia
4	Africa
...	...
25475	Asia
25476	Asia
25477	Asia
25478	Asia
25479	Asia

25480 rows × 1 columns

unique

In [24]:
```
# how many unique labels are there
visa_df['continent'].unique()
```

Out[24]:
```
array(['Asia', 'Africa', 'North America', 'Europe', 'South America',
       'Oceania'], dtype=object)
```

In [26]:
```
# python basic logics
l1=['a','a','b','c'] #['a','b','c']
set(l1)
```

Out[26]:
```
{'a', 'b', 'c'}
```

In [28]:
```
set(visa_df['continent'].values)
```

Out[28]:
```
{'Africa', 'Asia', 'Europe', 'North America', 'Oceania', 'South America'}
```

nunique

In [30]:
```
visa_df['continent'].nunique()
```

Out[30]:
```
6
```

> in the continent column only 7 elements id repeated

```
{'Africa', 'Asia', 'Europe', 'North America', 'Oceania', 'South America'}
```

out of total bservaions how many asia observations are there?

In [32]:
```
con=visa_df['continent']=='Asia'#true or false
visa_df[con]
```

Out[32]:

	case_id	continent	education_of_employee	has_job_experience	requires_job_training	no_o
0	EZYV01	Asia	High School	N	N	
1	EZYV02	Asia	Master's	Y	N	
2	EZYV03	Asia	Bachelor's	N	Y	
3	EZYV04	Asia	Bachelor's	N	N	
5	EZYV06	Asia	Master's	Y	N	
...	...	...	...	...	...	
475	EZYV25476	Asia	Bachelor's	Y	Y	
476	EZYV25477	Asia	High School	Y	N	
477	EZYV25478	Asia	Master's	Y	N	
478	EZYV25479	Asia	Master's	Y	Y	
479	EZYV25480	Asia	Bachelor's	Y	N	

361 rows × 12 columns

In [149]:
```
con=visa_df['continent']=='Africa'#true or false
visa_df[con]
```

Out[149]:

	case_id	continent	education_of_employee	has_job_experience	requires_job_training
4	EZYV05	Africa	Master's	Y	N
18	EZYV19	Africa	Master's	Y	N
74	EZYV75	Africa	Master's	Y	N
194	EZYV195	Africa	Master's	Y	N
242	EZYV243	Africa	Bachelor's	N	Y
...	...	...	...	...	...
25385	EZYV25386	Africa	Doctorate	Y	N
25408	EZYV25409	Africa	Master's	Y	Y
25443	EZYV25444	Africa	Bachelor's	N	N
25446	EZYV25447	Africa	Master's	N	Y
25474	EZYV25475	Africa	Doctorate	N	N

551 rows × 12 columns

Frequnecy Table

In [44]:

```
visa_df # Total data frame
visa_df['continent']#specific column
visa_df['continent']=='Asia' #specific labels
##############################################################################

len(visa_df[visa_df['continent']=='Asia'])

##############################################################################
unique_labels=visa_df['continent'].unique()
count=[]
for i in unique_labels:
    con=visa_df['continent']==i  #true or false
#     print(i,":",len(visa_df[con]))
    count.append(len(visa_df[con]))

##############################################################################
continent_df=pd.DataFrame(zip(unique_labels,count),columns=['Continent','Count'
##############################################################################
continent_df.to_csv('continent_df.csv',index=False)
```

In [ ]:

```
visa_df # Total data frame
visa_df['continent']#specific column
visa_df['continent']=='Asia' #specific labels
```

In [51]:

```
len(visa_df[visa_df['continent']=='Asia'])
```

Out[51]:

```
16861
```

In [52]:

```
continent_df
```

Out[52]:

	Continent	Count
0	Asia	16861
1	Africa	551
2	North America	3292
3	Europe	3732
4	South America	852
5	Oceania	192

$value - counts$

In [59]:
```
continent_vc=visa_df['continent'].value_counts()#series
continent_vc
```

Out[59]:
```
continent
Asia             16861
Europe            3732
North America     3292
South America      852
Africa             551
Oceania            192
Name: count, dtype: int64
```

In [ ]:
```
visa_df
visa_df['continent']
visa_df['continent'].unique()
visa_df['continent'].nunique()
visa_df['continent'].value_counts()
```

In [60]:
```
continent_vc.keys()
```

Out[60]:
```
Index(['Asia', 'Europe', 'North America', 'South America', 'Africa',
       'Oceania'],
      dtype='object', name='continent')
```

In [63]:
```
continent_vc.values
```

Out[63]:
```
array([16861,  3732,  3292,   852,   551,   192], dtype=int64)
```

In [66]:
```
continent_vc=visa_df['continent'].value_counts()#series
continent_vc
l1=continent_vc.keys()
l2=continent_vc.values
continent_vc_df=pd.DataFrame(zip(l1,l2),columns=['continent','count'])
print(continent_vc_df)
```

```
       continent  count
0           Asia  16861
1         Europe   3732
2  North America   3292
3  South America    852
4         Africa    551
5        Oceania    192
```

Bar Chart

- in order to draw a bar chart
- we require one categorcal coulmn
- we require one nummerical column
- package : matplotlip
- dataframe:continent_vc_df

In [69]:

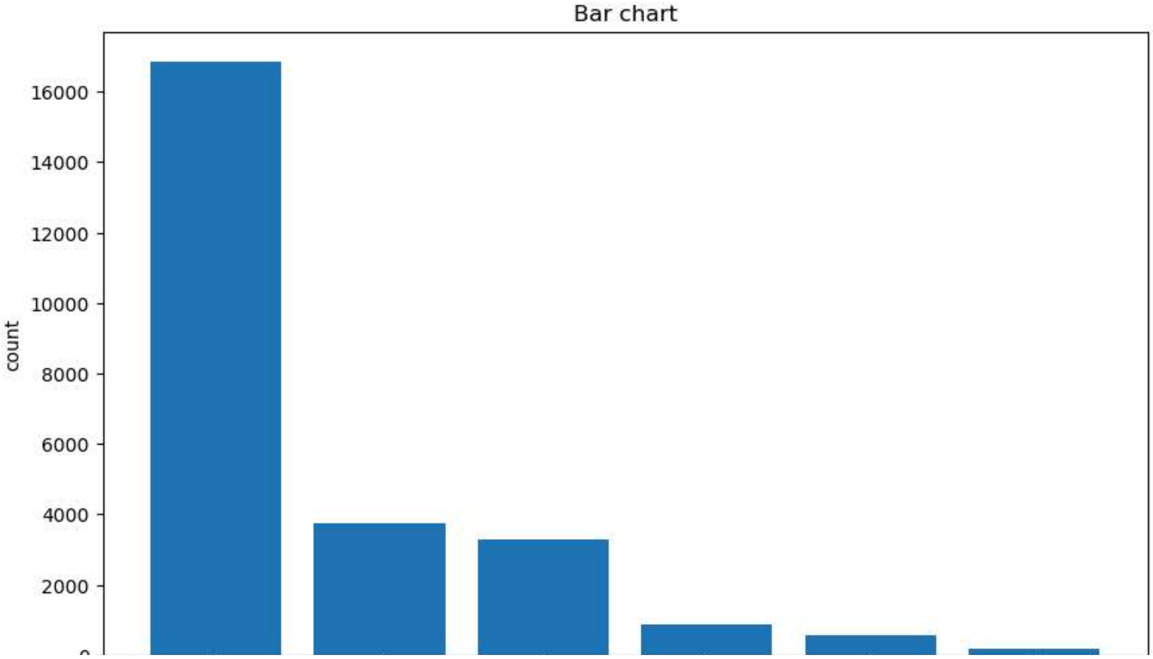
```
# plt.bar(<cat>,<nmer>,<data>)
continent_vc_df
```

Out[69]:

	continent	count
0	Asia	16861
1	Europe	3732
2	North America	3292
3	South America	852
4	Africa	551
5	Oceania	192

In [152]:

```
plt.figure(figsize=(10,6))#to increase the plot size
plt.bar('continent','count',data=continent_vc_df)
plt.xlabel('continent')#xaxis name
plt.ylabel('count')#yaxis name
plt.title('Bar chart')#title of the chart
plt.savefig('continent_bar.jpg')
plt.show()
```



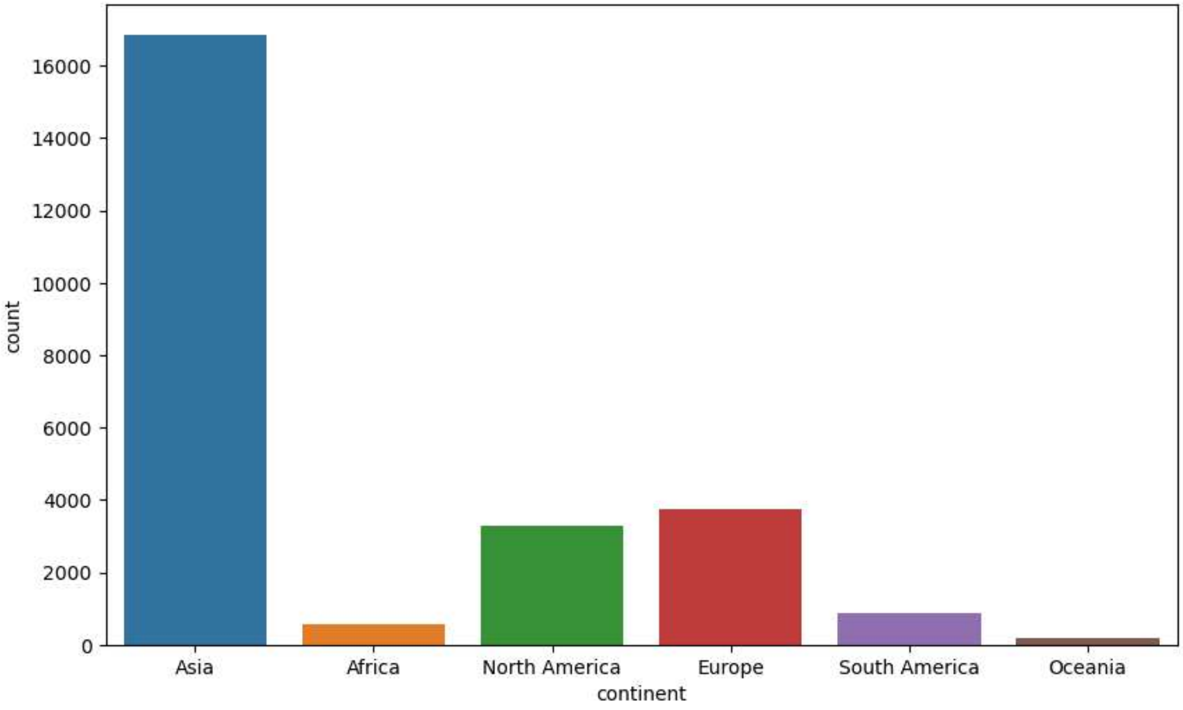
- we read the data
- we read categorical coulmn
- we made frequency table by using value count
- we plot the bar chart using matplotlib
- but matplotlib required 3 arguments
- xlabel :categorical coulmn(width)
- y-label:numerical coulmn(height)
- data(frequency table name)

COUNT PLOT

-count plot can be used bt seaborn package

- it requres only **entire data frame** and **categorical coulmns**`
- entire dataframe name : **Visadf**
- categorical column name: **content**
- in which order you want to plot

In [154]:
```
plt.figure(figsize=(10,6))
sns.countplot(data=visa_df,x='continent')
plt.show()
```



In [105]:
```
#  perform the same analayiss on education employee
#  show me the plots in whatsaap
#  take a screenshot and post in the group


l11=continent_vc1.keys()
```

In [131]:

```
education_vc1=visa_df['education_of_employee'].value_counts()#series
l11=education_vc1.keys()
l22=education_vc1.values

education_vc_df=pd.DataFrame(zip(l11,l22),columns=['Education','Student Count']
education_vc_df
# plt.figure(figsize=(8,5))#to increase the plot size
# plt.bar('Grade','Student-count',data=education_vc_df)
# plt.xlabel('Grade')#xaxis name
# plt.ylabel('Student-count')#yaxis name
# plt.title('Bar chart')#title of the chart
# # plt.savefig('Education_bar.jpg')
# # plt.show()
```

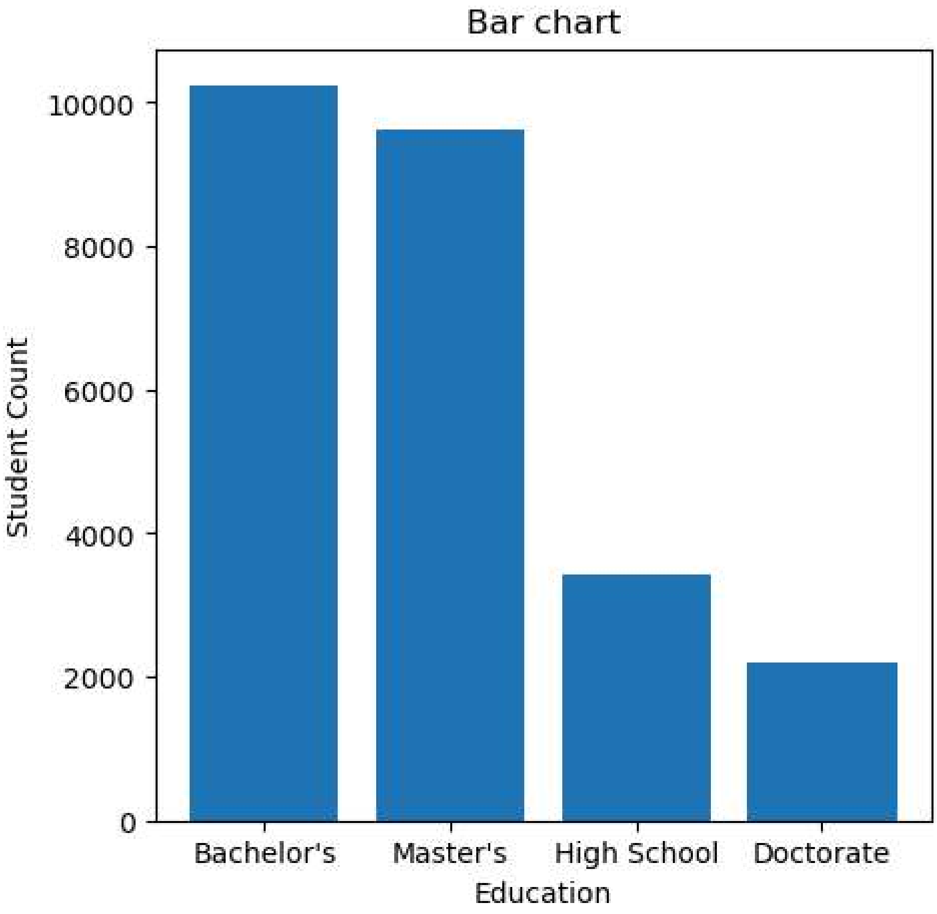
Out[131]:

	Education	Student Count
0	Bachelor's	10234
1	Master's	9634
2	High School	3420
3	Doctorate	2192

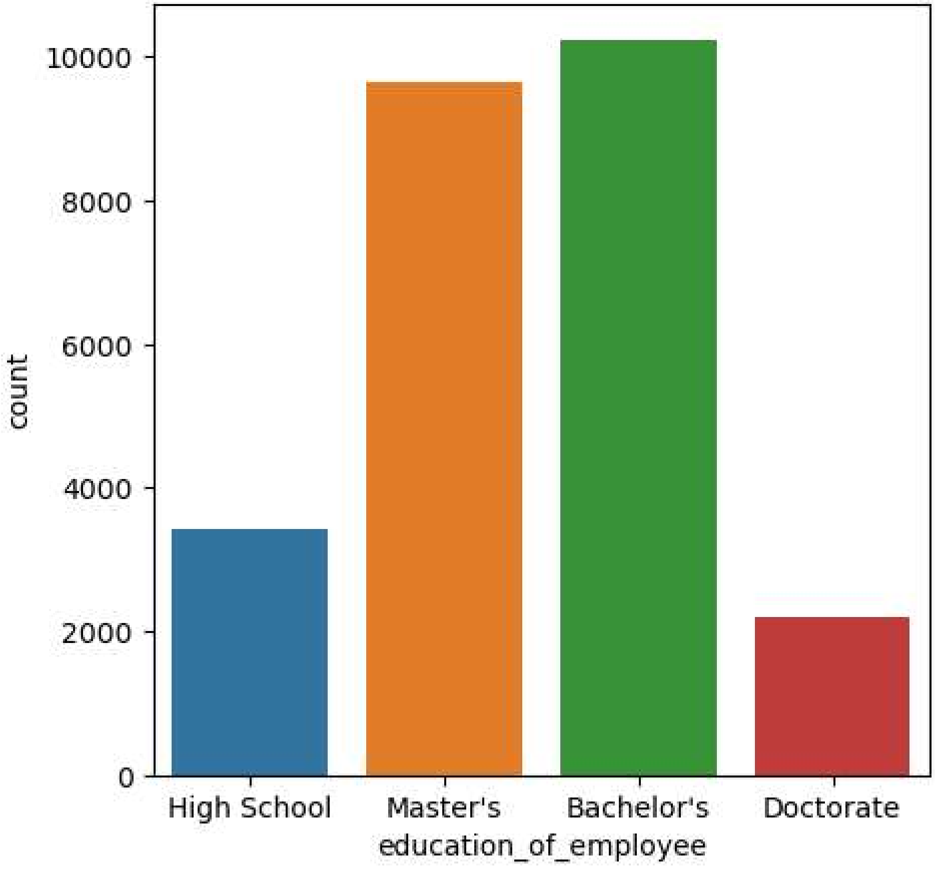
In [161]:

```
# plt.bar(<cat>,<nmer>plt.figure(figsize=(5,5)),<data>)
plt.figure(figsize=(5,5))
plt.bar('Education','Student Count',data=education_vc_df)

# plt.figure(figsize=(5,5))
plt.xlabel('Education')#xaxis name
plt.ylabel('Student Count')#yaxis name
plt.title('Bar chart')#title of the chart
plt.savefig('Education_bar.jpg')
plt.show()
```



In [167]:
```
# education_vc1=visa_df['education_of_employee'].value_counts()#series
# l11=education_vc1.keys()
# l22=education_vc1.values
# education_vc_df=pd.DataFrame(zip(l11,l22),columns=['Education','Student Count
# education_vc_df
plt.figure(figsize=(5,5))
sns.countplot(data=visa_df,x='education_of_employee')
plt.show()
```



In [ ]:
```
```

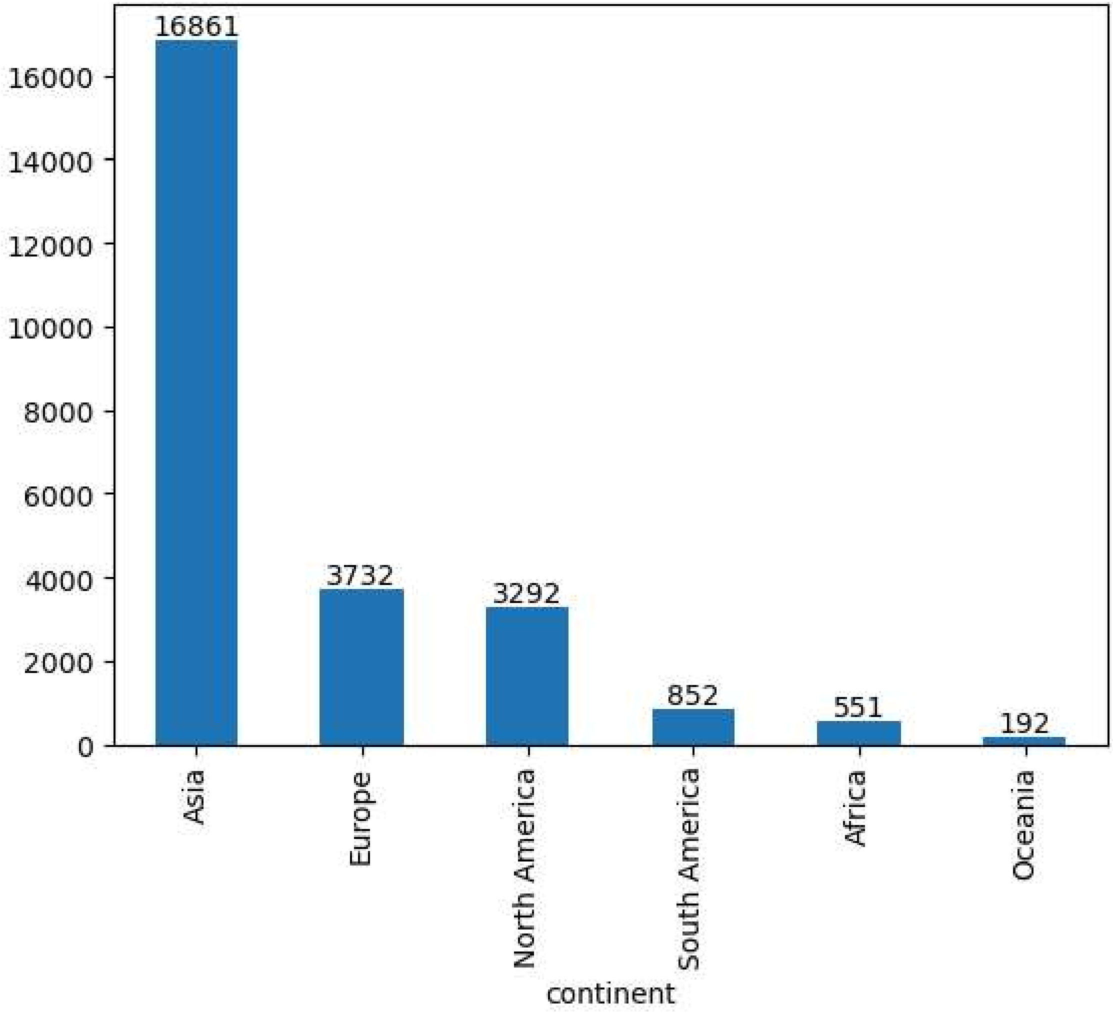
In [ ]:
```
```

In [ ]:
```
```

In [ ]:
```
method 3
```

- we created the a freqency table: matplotlib
- we created bar chart using seaborn
- main data frame
- column name
- by using value counts

```
In [160]: values=visa_df['continent'].value_counts()
          ax=values.plot(kind='bar')
          ax.bar_label(ax.containers[0])
          plt.show()
```



RElatve frequncy

```
In [112]: visa_df['continent'].value_counts(normalize=True)

Out[112]: continent
          Asia             0.661735
          Europe           0.146468
          North America    0.129199
          South America    0.033438
          Africa           0.021625
          Oceania          0.007535
          Name: proportion, dtype: float64
```

Pie- Chart

- x is data in the frm of list
- labels also in form of list

- will take value count help without normalizing
- pie chart will automatically convert values to percentages

In [117]:

```
values=visa_df['continent'].value_counts().keys()
values=visa_df['continent'].value_counts().values
values
```

Out[117]:

```
array([16861,  3732,  3292,   852,   551,   192], dtype=int64)
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: