Reading CSV Files

In [4]:
```
import pandas as pd #Dataframe operations
import numpy as np   #Math operations
import matplotlib.pyplot as plt #Diagram/plots
import seaborn as sns #Diagram/plots
```

In [ ]:
```
# data set name is visa dataset
# read csv file: Comma seperated value
# extension: .csv
# you can read this using pandas package
# read excel file
# extension: .xlsx
```

In [5]:
```
path=r'C:\Users\aramaiah.ASUAD\Naresh_IT\MyDataScience\Data_Files\Visadataset.c
pd.read_csv(path)
```

Out[5]:

	case_id	continent	education_of_employee	has_job_experience	requires_job_training
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
4	EZYV05	Africa	Master's	Y	N
...	...	...	...	...	...
25475	EZYV25476	Asia	Bachelor's	Y	Y
25476	EZYV25477	Asia	High School	Y	N
25477	EZYV25478	Asia	Master's	Y	N
25478	EZYV25479	Asia	Master's	Y	Y
25479	EZYV25480	Asia	Bachelor's	Y	N

25480 rows × 12 columns

In [10]:
```
path =r'C:\Users\aramaiah.ASUAD\Naresh_IT\MyDataScience\Data_Files\bank.csv'
pd.read_csv(path,sep=';',header=None)
```

Out[10]:

	0	1	2	3	4	5	6	7	8	9	10
0	age	job	marital	education	default	balance	housing	loan	contact	day	month
1	30	unemployed	married	primary	no	1787	no	no	cellular	19	oct
2	33	services	married	secondary	no	4789	yes	yes	cellular	11	may
3	35	management	single	tertiary	no	1350	yes	no	cellular	16	apr
4	30	management	married	tertiary	no	1476	yes	yes	unknown	3	jun
...	...	...	...	...	...	...	...	...	...	...	...
4517	33	services	married	secondary	no	-333	yes	no	cellular	30	jul
4518	57	self-employed	married	tertiary	yes	-3313	yes	yes	unknown	9	may
4519	57	technician	married	secondary	no	295	no	no	cellular	19	aug
4520	28	blue-collar	married	secondary	no	1137	no	no	cellular	6	feb
4521	44	entrepreneur	single	tertiary	no	1136	yes	yes	cellular	3	apr

4522 rows × 17 columns

In [8]:
```
path=r'C:\Users\aramaiah.ASUAD\Naresh_IT\MyDataScience\Data_Files\sample2.csv'
pd.read_csv(path)
```

Out[8]:

	Unnamed: 0	Col1	Col2
0	0	Ram	25
1	1	Sita	22
2	2	Laxman	20

In [ ]:
```
path=r'Salary_Data'
```

Create Dataframes using List

```
<!-- provide data here -->
```

In [11]:
```
name=['Sriya','Navya','Divya','Aishwarya']
age=[30,48,78,80]
name,age
```

Out[11]:
```
(['Sriya', 'Navya', 'Divya', 'Aishwarya'], [30, 48, 78, 80])
```

**Step 1: Create the data frame

In [12]:
```
pd.DataFrame() #make the dataframe
```

Out[12]:

—

In [ ]:
```
- Step 2:
```

In [13]:
```
pd.DataFrame(zip(name,age))
```

Out[13]:

	0	1
0	Sriya	30
1	Navya	48
2	Divya	78
3	Aishwarya	80

In [14]:
```
data=zip(name,age)
pd.DataFrame(data,columns=['Name','Age'])
```

Out[14]:

	Name	Age
0	Sriya	30
1	Navya	48
2	Divya	78
3	Aishwarya	80

In [20]:
```
data=zip(name,age)
cols=['Name','Age']
ind=['A','B','C','D']
df=pd.DataFrame(data,columns=cols,index=ind)
df
```

Out[20]:

	Name	Age
A	Sriya	30
B	Navya	48
C	Divya	78
D	Aishwarya	80

- if you want to add a new column
- df['new column']
- you need to have a list which is having some elements
- that elemts need to equal the number of rows
- city_names=['Hyd','Blr','Chen','Nel']
- df['city']=city_names

In [22]:
```
city_names=['Hyd','Blr','Chen','Nel']
df['city']=city_names
df
```

Out[22]:

	Name	Age	city
A	Sriya	30	Hyd
B	Navya	48	Blr
C	Divya	78	Chen
D	Aishwarya	80	Nel

Step 6 : Update the existing coulmn

Update the existing Coulumn withnewnames

In [25]:
```
df['Name']=['Swamy', 'Vaddi','Shiro','Atiya']
df
```

Out[25]:

	Name	Age	city	Names
A	Swamy	30	Hyd	Swamy
B	Vaddi	48	Blr	Vaddi
C	Shiro	78	Chen	Shiro
D	Atiya	80	Nel	Atiya

Step 8 **Drop the column**

- in order to drop the coulmn we need to take 3 parameters
- we need to use drop method
- it takes 3 parameters
- drop coulmn or row
- mention the coulmn name
- provide the axis
  - axis=1 referance as coulmn
  - axis=0 referance of row -you want to create a new dataframe or
- you want to overwrite the existing data frame
  - inplace = True

In [35]:
```
df.drop('city',#column name

        axis=1,#column
        inplace=True # overwrite the same )
df
```

Out[35]:

	Name	Age
A	Swamy	30
B	Vaddi	48
C	Shiro	78
D	Atiya	80

- Step 9: to save a file

In [40]:
```
# To save a file
df.to_csv('output.csv',index=False)
# while saving index is considered as extra column
# while saving to avoid the above problem we need to give index=False
df.to_excel('output.xlsx')
```

- Step 10: remove the index

In [ ]:
```
df.to_csv('output.csv',index=False)
# while saving index is considered as extra column
# while saving to avoid the above problem we need to give index=False
```

In [39]:
```
family=['Aishwarya','Ramaiah','Kumar','Usha','Anushka']
age=[25,85,56,45,85]

pd.DataFrame()
```

Out[39]:

—

- Create Data Frames using Dictionary

In [42]:
```
d1={'name':['Ramesh','Suresh','Sathish'],'age':[30,35,40]}
pd.DataFrame(d1)
```

Out[42]:

	name	age
0	Ramesh	30
1	Suresh	35
2	Sathish	40

In [ ]:
```
# no need to use zip and coumn names
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: