

1. Import all the required Python Libraries.

```
import pandas as pd
```

3. Load the dataset into pandas dataframe

```
df = pd.read_csv('Social_Network_Ads.csv')
```

df

	User ID	Gender	Age	EstimatedSalary	Purchased
0	15624510	Male	19.0	19000.0	0
1	15810944	Male	35.0	20000.0	0
2	15668575	Female	26.0	43000.0	0
3	15603246	Female	27.0	57000.0	0
4	15804002	Male	19.0	76000.0	0
...
395	15691863	Female	46.0	41000.0	1
396	15706071	Male	51.0	23000.0	1
397	15654296	Female	50.0	20000.0	1
398	15755018	Male	36.0	33000.0	0
399	15594041	Female	49.0	36000.0	1

400 rows x 5 columns

Saving...

statistics. Provide variable descriptions *

```
df.describe()
```

	User ID	Age	EstimatedSalary	Purchased
count	4.000000e+02	400.000000	400.000000	400.000000
mean	1.569154e+07	37.655000	69742.500000	0.357500
std	7.165832e+04	10.482877	34096.960282	0.479864
min	1.556669e+07	18.000000	15000.000000	0.000000
25%	1.562676e+07	29.750000	43000.000000	0.000000
50%	1.569434e+07	37.000000	70000.000000	0.000000
75%	1.575036e+07	46.000000	88000.000000	1.000000
max	1.581524e+07	60.000000	150000.000000	1.000000

```
df.describe
```

<bound method NDFrame.describe of					User ID	Gender	Age	EstimatedSalary	Purchased
0	15624510	Male	19.0	19000.0	0				
1	15810944	Male	35.0	20000.0	0				
2	15668575	Female	26.0	43000.0	0				
3	15603246	Female	27.0	57000.0	0				
4	15804002	Male	19.0	76000.0	0				
..			
395	15691863	Female	46.0	41000.0	1				
396	15706071	Male	51.0	23000.0	1				
397	15654296	Female	50.0	20000.0	1				
398	15755018	Male	36.0	33000.0	0				
399	15594041	Female	49.0	36000.0	1				

[400 rows x 5 columns]>

4. Data Preprocessing: check for missing values in the data using pandas isnull()

```
df.isnull()
```

```
User ID Gender Age EstimatedSalary Purchased
0 False False False False False
1 False False False False False
2 False False False False False
3 False False False False False
4 False False False False False
... ..
395 False False False False False
396 False False False False False
397 False False False False False
398 False False False False False
399 False False False False False

400 rows x 5 columns

df.isnull().sum()

User ID      0
Gender       0
Age          0
EstimatedSalary 0
Purchased    0
dtype: int64

student = pd.read_csv('student2.csv')

student
```

	roll	name	class	marks	age
0	1	anil	TE	56.77	22.0
1	2	aniket	BE	76.88	19.0
2	3	ajinkya	TE	69.66	NaN
3	4	asha	NaN	63.28	20.0
4	5	ayesha	BE	49.55	20.0
5	6	amar	BE	NaN	19.0
6	7	amita	BE	NaN	23.0
7	8	amol	TE	56.75	20.0
8	9	anmol	BE	78.66	21.0

```
student.isnull()

roll name class marks age
0 False False False False False
1 False False False False False
2 False False False False False
3 False False False False True
4 False False True False False
5 False False False False False
6 False False False True False
7 False False False True False
8 False False False False False
9 False False False False False

student.isnull().sum()

roll      0
name      0
class     1
```

```
marks    2
age       1
dtype: int64
```

4. Check the dimensions of the data frame

```
df.shape

(400, 5)
```

4. Data Formatting and Data Normalization: Summarize the types of variables by checking the data types (i.e., character, numeric, integer, factor, and logical) of the variables in the data set.

```
df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 400 entries, 0 to 399
Data columns (total 5 columns):
#   Column                Non-Null Count  Dtype  
---  -
0   User ID               400 non-null   int64  
1   Gender                400 non-null   object  
2   Age                   400 non-null   float64 
3   EstimatedSalary       400 non-null   float64 
4   Purchased             400 non-null   int64  
dtypes: float64(2), int64(2), object(1)
memory usage: 15.8+ KB
```

```
df.dtypes

User ID      int64
Gender      object
Age         float64
EstimatedSalary float64
Purchased   int64
dtype: object
```

Saving... X

```
dtype('float64')
```

5. Turn categorical variables into quantitative variables in Python.

Label Encoding

```
from sklearn.preprocessing import LabelEncoder

le = LabelEncoder()

df['Gender'] = le.fit_transform(df['Gender'])
```

df

	User ID	Gender	Age	EstimatedSalary	Purchased
0	15624510	1	19.0	19000.0	0
1	15810944	1	35.0	20000.0	0
2	15668575	0	26.0	43000.0	0
3	15603246	0	27.0	57000.0	0
4	15804002	1	19.0	76000.0	0
...
395	15691863	0	46.0	41000.0	1
396	15706071	1	51.0	23000.0	1
397	15654296	0	50.0	20000.0	1
398	15755018	1	36.0	33000.0	0
399	15594041	0	49.0	36000.0	1

400 rows × 5 columns



```
df.dtypes['Gender']

dtype('int64')

df['Gender'] = le.inverse_transform(df['Gender'])

df.head()
```

	User ID	Gender	Age	EstimatedSalary	Purchased	
0	15624510	Male	19.0	19000.0	0	
1	15810944	Male	35.0	20000.0	0	
2	15668575	Female	26.0	43000.0	0	
3	15603246	Female	27.0	57000.0	0	
4	15804002	Male	19.0	76000.0	0	

one-hot Encoding

```
df = pd.read_csv('Social_Network_Ads.csv')
pd.get_dummies(df)
```

	User ID	Age	EstimatedSalary	Purchased	Gender_Female	Gender_Male	
0	15624510	19.0	19000.0	0	0	1	
1	15810944	35.0	20000.0	0	0	1	
2	15668575	26.0	43000.0	0	1	0	
3	15603246	27.0	57000.0	0	1	0	
4	15804002	19.0	76000.0	0	0	1	
...	
			1000.0	1	1	0	
			3000.0	1	0	1	
397	15654296	50.0	20000.0	1	1	0	
398	15755018	36.0	33000.0	0	0	1	
399	15594041	49.0	36000.0	1	1	0	

400 rows × 6 columns