CODING CHALLENGE – 3 (Python)

TRAINEE NAME: Swathi Baskaran

Joins:

```
In [1]: import pandas as pd

        #Joins
        df1 = pd.DataFrame({
            'ID': [1,2,3],
            'Name':["Alice","Charlie","Jack"]
        })

        df2 = pd.DataFrame({
            'ID': [2,3,4],
            'Marks': [78, 45, 97]
        })
```

1. Inner Join

```
In [3]: innerJoin = pd.merge(df1, df2, on = 'ID', how = 'inner')
        print(innerJoin)
```

```
   ID     Name  Marks
0   2  Charlie     78
1   3     Jack     45
```

2. Left Join

```
In [5]: leftJoin = pd.merge(df1, df2, on = 'ID', how = 'left')
        print(leftJoin)
```

```
   ID     Name  Marks
0   1    Alice    NaN
1   2  Charlie   78.0
2   3     Jack   45.0
```

3. Right Join

```
In [6]: rightJoin = pd.merge(df1, df2, on = 'ID', how = 'right')
        print(rightJoin)
```

```
   ID     Name  Marks
0   2  Charlie     78
1   3     Jack     45
2   4      NaN     97
```

4. Outer Join

```
In [7]: outerJoin = pd.merge(df1, df2, on = 'ID', how = 'outer')
        print(outerJoin)
```

```
   ID     Name  Marks
0   1    Alice    NaN
1   2  Charlie   78.0
2   3     Jack   45.0
3   4      NaN   97.0
```

Data Cleaning:

```
In [10]: # Data Cleaning
         df = pd.read_csv("D:\Victus Laptop\Downloads\Hexaware\Data Engineering Training\Drive Material\Python\LoanData (1).csv")
         df.head()
```

Out[10]:

	Loan_ID	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History
0	LP001002	Male	No	0	Graduate	No	5849	0.0	NaN	360.0	1.0
1	LP001003	Male	Yes	1	Graduate	No	4583	1508.0	128.0	360.0	1.0
2	LP001005	Male	Yes	0	Graduate	Yes	3000	0.0	66.0	360.0	1.0
3	LP001006	Male	Yes	0	Not Graduate	No	2583	2358.0	120.0	360.0	1.0
4	LP001008	Male	No	0	Graduate	No	6000	0.0	141.0	360.0	1.0

```
In [11]: df.dropna(inplace = True)

In [12]: df.fillna(0, inplace = True)

In [13]: df.drop_duplicates(inplace=True)

In [16]: df.isnull().sum()
```

```
Out[16]: Loan_ID              0
         Gender               0
         Married              0
         Dependents           0
         Education            0
         Self_Employed        0
         ApplicantIncome      0
         CoapplicantIncome    0
         LoanAmount           0
         Loan_Amount_Term     0
         Credit_History       0
         Property_Area        0
         Loan_Status          0
         dtype: int64
```

After Cleaning:

In [17]:
```
df.head()
```

Out[17]:

	Loan_ID	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History
1	LP001003	Male	Yes	1	Graduate	No	4583	1508.0	128.0	360.0	1.0
2	LP001005	Male	Yes	0	Graduate	Yes	3000	0.0	66.0	360.0	1.0
3	LP001006	Male	Yes	0	Not Graduate	No	2583	2358.0	120.0	360.0	1.0
4	LP001008	Male	No	0	Graduate	No	6000	0.0	141.0	360.0	1.0
5	LP001011	Male	Yes	2	Graduate	Yes	5417	4196.0	267.0	360.0	1.0

In [18]:
```
df.describe()
```

Out[18]:

	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History
count	480.000000	480.000000	480.000000	480.000000	480.000000
mean	5364.231250	1581.093583	144.735417	342.050000	0.854167
std	5668.251251	2617.692267	80.508164	65.212401	0.353307
min	150.000000	0.000000	9.000000	36.000000	0.000000
25%	2898.750000	0.000000	100.000000	360.000000	1.000000
50%	3859.000000	1084.500000	128.000000	360.000000	1.000000
75%	5852.500000	2253.250000	170.000000	360.000000	1.000000
max	81000.000000	33837.000000	600.000000	480.000000	1.000000