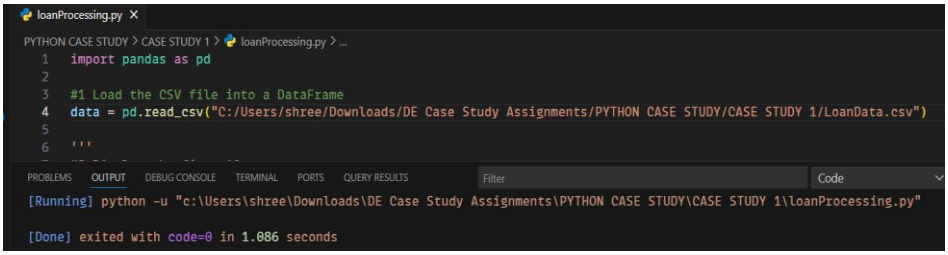
Data Processing with Pandas case study

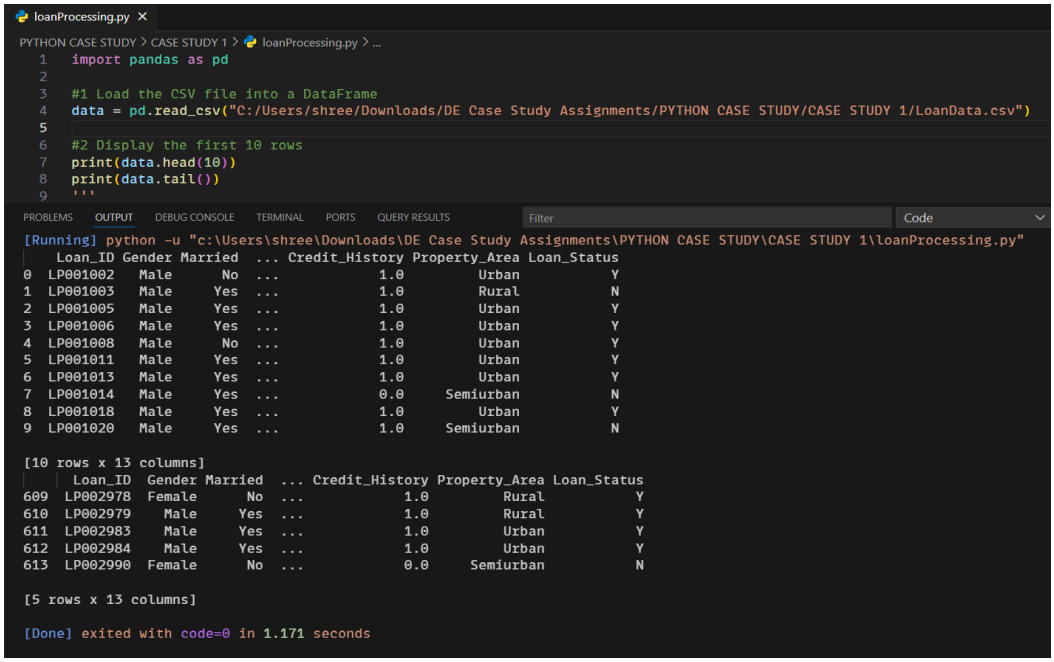
1. Loading Data in Pandas DataFrame



```
loanProcessing.py
PYTHON CASE STUDY > CASE STUDY 1 > loanProcessing.py > ...
1  import pandas as pd
2
3  #1 Load the CSV file into a DataFrame
4  data = pd.read_csv("C:/Users/shree/Downloads/DE Case Study Assignments/PYTHON CASE STUDY/CASE STUDY 1/LoanData.csv")
5
6  '''
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  QUERY RESULTS   Filter   Code
[Running] python -u "c:\Users\shree\Downloads\DE Case Study Assignments\PYTHON CASE STUDY\CASE STUDY 1\loanProcessing.py"

[Done] exited with code=0 in 1.086 seconds
```

2. Printing rows of the Data



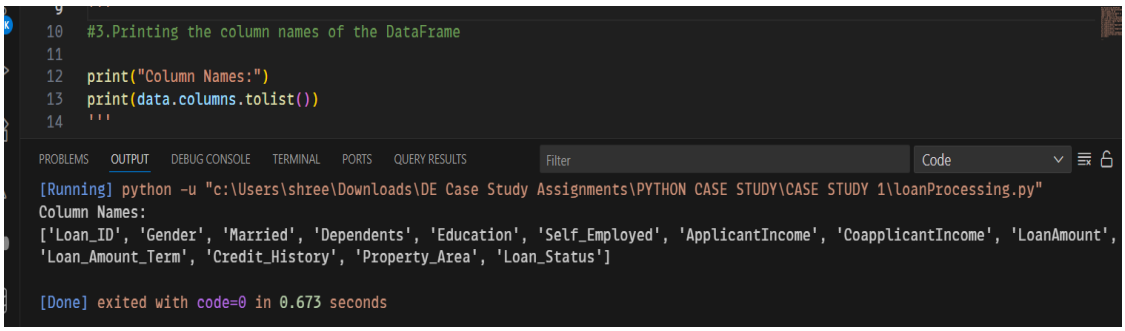
```
loanProcessing.py
PYTHON CASE STUDY > CASE STUDY 1 > loanProcessing.py > ...
1  import pandas as pd
2
3  #1 Load the CSV file into a DataFrame
4  data = pd.read_csv("C:/Users/shree/Downloads/DE Case Study Assignments/PYTHON CASE STUDY/CASE STUDY 1/LoanData.csv")
5
6  #2 Display the first 10 rows
7  print(data.head(10))
8  print(data.tail())
9  '''
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  QUERY RESULTS   Filter   Code
[Running] python -u "c:\Users\shree\Downloads\DE Case Study Assignments\PYTHON CASE STUDY\CASE STUDY 1\loanProcessing.py"
    Loan_ID Gender Married  ... Credit_History Property_Area Loan_Status
0  LP001002   Male      No  ...            1.0         Urban           Y
1  LP001003   Male     Yes  ...            1.0         Rural           N
2  LP001005   Male     Yes  ...            1.0         Urban           Y
3  LP001006   Male     Yes  ...            1.0         Urban           Y
4  LP001008   Male      No  ...            1.0         Urban           Y
5  LP001011   Male     Yes  ...            1.0         Urban           Y
6  LP001013   Male     Yes  ...            1.0         Urban           Y
7  LP001014   Male     Yes  ...            0.0     Semiurban           N
8  LP001018   Male     Yes  ...            1.0         Urban           Y
9  LP001020   Male     Yes  ...            1.0     Semiurban           N

[10 rows x 13 columns]
     Loan_ID  Gender Married  ... Credit_History Property_Area Loan_Status
609  LP002978  Female      No  ...            1.0         Rural           Y
610  LP002979    Male     Yes  ...            1.0         Rural           Y
611  LP002983    Male     Yes  ...            1.0         Urban           Y
612  LP002984    Male     Yes  ...            1.0         Urban           Y
613  LP002990  Female      No  ...            0.0     Semiurban           N

[5 rows x 13 columns]

[Done] exited with code=0 in 1.171 seconds
```

3. Printing the column names of the DataFrame



```
10  #3.Printing the column names of the DataFrame
11
12  print("Column Names:")
13  print(data.columns.tolist())
14  '''
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  QUERY RESULTS   Filter   Code
[Running] python -u "c:\Users\shree\Downloads\DE Case Study Assignments\PYTHON CASE STUDY\CASE STUDY 1\loanProcessing.py"
Column Names:
['Loan_ID', 'Gender', 'Married', 'Dependents', 'Education', 'Self_Employed', 'ApplicantIncome', 'CoapplicantIncome', 'LoanAmount',
'Loan_Amount_Term', 'Credit_History', 'Property_Area', 'Loan_Status']

[Done] exited with code=0 in 0.673 seconds
```

4. Summary of Data Frame

```
#4.Summary of Data Frame

print("\nSummary of data frame:")
print(data.info())
'''
```

```
Summary of data frame:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 614 entries, 0 to 613
Data columns (total 13 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   Loan_ID            614 non-null    object
 1   Gender             601 non-null    object
 2   Married            611 non-null    object
 3   Dependents         599 non-null    object
 4   Education          614 non-null    object
 5   Self_Employed      582 non-null    object
 6   ApplicantIncome    614 non-null    int64
 7   CoapplicantIncome  614 non-null    float64
 8   LoanAmount         592 non-null    float64
 9   Loan_Amount_Term   600 non-null    float64
 10  Credit_History     564 non-null    float64
 11  Property_Area      614 non-null    object
 12  Loan_Status        614 non-null    object
dtypes: float64(4), int64(1), object(8)
memory usage: 62.5+ KB
None

[Done] exited with code=0 in 0.667 seconds
```

5. Descriptive Statistical Measures of a Data Frame

```
#5.Descriptive Statistical Measures of a DataFrame
print("\nDescriptive Statistical Measures of a DataFrame:")
print(data.describe())
'''
```

```
[Running] python -u "c:\Users\shree\Downloads\DE Case Study Assignments\PYTHON CASE STUDY

Descriptive Statistical Measures of a DataFrame:
       ApplicantIncome  CoapplicantIncome  ...  Loan_Amount_Term  Credit_History
count       614.000000         614.000000  ...         600.00000      564.000000
mean       5403.459283        1621.245798  ...         342.00000        0.842199
std        6109.041673        2926.248369  ...          65.12041        0.364878
min         150.000000           0.000000  ...          12.00000        0.000000
25%        2877.500000           0.000000  ...         360.00000        1.000000
50%        3812.500000        1188.500000  ...         360.00000        1.000000
75%        5795.000000        2297.250000  ...         360.00000        1.000000
max       81000.000000       41667.000000  ...         480.00000        1.000000

[8 rows x 5 columns]

[Done] exited with code=0 in 0.673 seconds
```

6. Merge Data Frames

```
#8.Merge Data Frames
# Creating anotherDataFrame for merge
new_data = pd.DataFrame({
    'Loan_ID': data['Loan_ID'],
    'Status': ['Verified']*len(data)
})
# Merge on Loan_ID
merged_data = pd.merge(data, new_data, on='Loan_ID')
print(merged_data.head(3))
```

```
[Running] python -u "c:\Users\shree\Downloads\DE Case Study Assignments\
    Loan_ID Gender Married  ... Property_Area Loan_Status    Status
0  LP001002   Male      No  ...         Urban           Y  Verified
1  LP001003   Male     Yes  ...         Rural           N  Verified
2  LP001005   Male     Yes  ...         Urban           Y  Verified

[3 rows x 14 columns]

[Done] exited with code=0 in 0.66 seconds
```

7. Sorting DataFrame values

```
#7. Sorting DataFrame values
# Sort by ApplicantIncome in descending order
sorted_data = data.sort_values(by='ApplicantIncome', ascending=False)
print(sorted_data[['ApplicantIncome', 'LoanAmount']].head())

'''
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS QUERY RESULTS

```
[Running] python -u "c:\Users\shree\Downloads\DE Case Study Assignments\PYTHON
     ApplicantIncome  LoanAmount
409            81000       360.0
333            63337       490.0
171            51763       700.0
155            39999       600.0
185            39147       120.0

[Done] exited with code=0 in 0.754 seconds
```

8. Apply Function

```
#9.Apply Function
# Define a function to categorize loan risk
def loan_risk(row):
    if row['Credit_History'] == 0 and row['LoanAmount'] > 150:
        return 'High Risk'
    elif row['Credit_History'] == 1 and row['LoanAmount'] <= 150:
        return 'Low Risk'
    else:
        return 'Moderate Risk'
# Apply the function row-wise
data['LoanRisk'] = data.apply(loan_risk, axis=1)
# Display result
print(data[['Credit_History', 'LoanAmount', 'LoanRisk']].head())
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS QUERY RESULTS

```
[Running] python -u "c:\Users\shree\Downloads\DE Case Study Assignments\PYTHON CASE STUDY\
   Credit_History  LoanAmount       LoanRisk
0             1.0         NaN  Moderate Risk
1             1.0       128.0       Low Risk
2             1.0        66.0       Low Risk
3             1.0       120.0       Low Risk
4             1.0       141.0       Low Risk

[Done] exited with code=0 in 0.662 seconds
```

9. Visualizing Data Frame

```
#11.Visualizing DataFrame
data.plot(x='ApplicantIncome', y='CoapplicantIncome', kind= 'hist')
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
data = pd.read_csv("C:/Users/shree/Downloads/DE Case Study Assignments/PYTHON CASE STUDY/CASE STUDY 1/LoanData.csv")
plt.style.use('fivethirtyeight')
plt.rcParams['figure.figsize'] = (15,5)
plt.subplot(1,3,1)
sns.boxplot(data['ApplicantIncome'])
plt.subplot(1,3,2)
sns.boxplot(data['CoapplicantIncome'])
plt.subplot(1,3,3)
sns.boxplot(data['LoanAmount'])
plt.suptitle('Outliers Detection')
plt.show()
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS QUERY

```
[Running] python -u "c:\Users\shree\Downloads\
```