**IBM NAANMUTHALVAN**

**PHASE 3 PROJECT**

**TEAM MEMBERS:**

**A.AHAMED IBRAHIM(820421205002)(samadhasanalthauf@gmail.com)**

**J.JAYACHANDAR(820421205033)(bugal890@gmail.com)**

**K.HARI KRISHNA(820421205027)(harikrishna22032004@gmail.com)**

**S.BALAMURUGAN(820421205013)(balamurugan610001@gmail.com)**

**DOMAIN NAME :**

**DATA ANALYTICS WITH IBM COGNOS**

**PROJECT NAME:**

**ASSESSMENT OF TN MARGINAL WORKERS –A SOCIO-ECONOMIC ANALYSIS.**

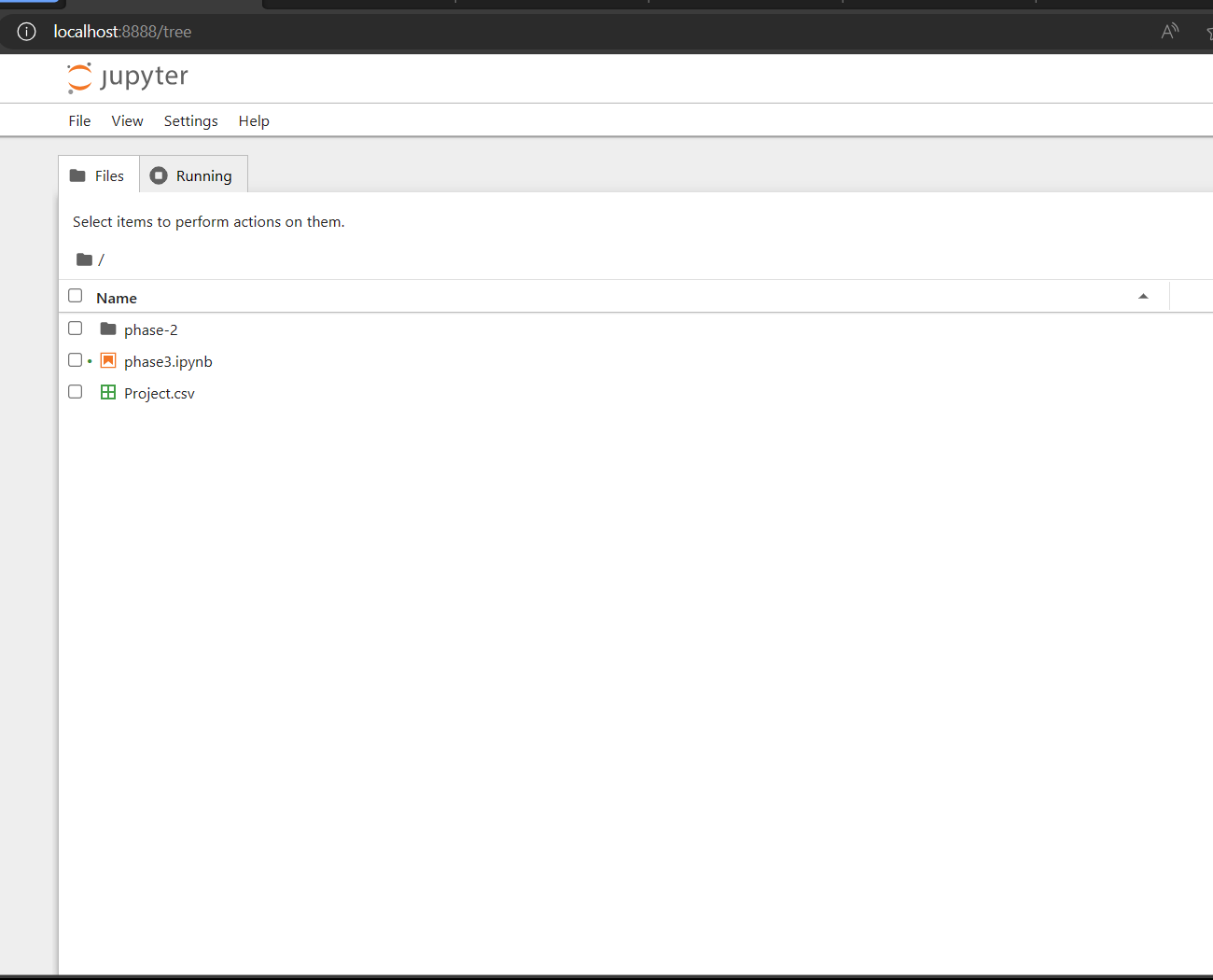
To load and pre-process a dataset using Python and libraries like pandas, you need to follow these steps. Let's assume you have a CSV file as your dataset.

1. Import the Required Libraries: First, we need to import the necessary libraries. In this case, we'll use pandas for data manipulation.
2. Load the Dataset: Use the pd.read\_csv() function to load our dataset. If you have a different type of dataset (e.g., Excel, JSON, SQL), pandas provides functions to read those as well.
3. Explore the Dataset: After loading the dataset, it's a good practice to take a quick look at its contents to understand its structure.
4. Data Preprocessing: Depending on the dataset and our project's needs, we might need to pre-process the data. Common preprocessing tasks include handling missing values, encoding categorical variables, scaling or normalizing numerical features, and more.

Here are some examples of data preprocessing:

Handling Missing Values: If our dataset has missing values, you can use methods like dropna() to handle them.

1. Encoding Categorical Variables: If our dataset contains categorical variables, we might want to encode them into numerical values using techniques like one-hot encoding.
2. Scaling/Normalizing Numerical Features: For machine learning models, it's often a good practice to scale or normalize numerical features.



**Program:**

import pandas as pd

import numpy as np

df=pd.read\_csv("D:\\pythonproject\Project.csv")

head=df.head()

print(head)

**output:**

Table Code State Code District Code Area Name Total/ Rural/ Urban \

0 B0806SC `33 `000 State - TAMIL NADU Total

1 B0806SC `33 `000 State - TAMIL NADU Total

2 B0806SC `33 `000 State - TAMIL NADU Total

3 B0806SC `33 `000 State - TAMIL NADU Total

4 B0806SC `33 `000 State - TAMIL NADU Total

Age group Worked for 3 months or more but less than 6 months - Persons \

0 Total 1200828

1 `5-14 27791

2 15-34 514340

3 35-59 542581

4 60+ 115103

Worked for 3 months or more but less than 6 months - Males \

0 589003

1 14125

2 259560

3 251957

4 62833

Worked for 3 months or more but less than 6 months - Females \

0 611825

1 13666

2 254780

3 290624

4 52270

Worked for less than 3 months - Persons ... \

0 221386 ...

1 2447 ...

2 92423 ...

3 99202 ...

4 27165 ...

Industrial Category - N to O - Females \

0 3565

1 11

2 1754

3 1619

4 175

Industrial Category - P to Q - Persons \

0 11080

1 122

2 7536

3 3205

4 211

Industrial Category - P to Q - Males \

0 4019

1 71

2 2718

3 1131

4 93

Industrial Category - P to Q - Females \

0 7061

1 51

2 4818

3 2074

4 118

Industrial Category - R to U - HHI - Persons \

0 16833

1 427

2 8346

3 6591

4 1457

Industrial Category - R to U - HHI - Males \

0 4266

1 169

2 2127

3 1487

4 483

Industrial Category - R to U - HHI - Females \

0 12567

1 258

2 6219

3 5104

4 974

Industrial Category - R to U - Non HHI - Persons \

0 122088

1 19305

2 68929

3 26498

4 7065

Industrial Category - R to U - Non HHI - Males \

0 55801

1 9774

2 32803

3 9675

4 3394

Industrial Category - R to U - Non HHI - Females

0 66287

1 9531

2 36126

3 16823

4 3671

[5 rows x 69 columns]

**program:**

print(df.isnull().sum())

**output:**

Table Code 0

State Code 0

District Code 0

Area Name 0

Total/ Rural/ Urban 0

..

Industrial Category - R to U - HHI - Males 0

Industrial Category - R to U - HHI - Females 0

Industrial Category - R to U - Non HHI - Persons 0

Industrial Category - R to U - Non HHI - Males 0

Industrial Category - R to U - Non HHI - Females 0

Length: 69, dtype: int64

**program:**

new\_data=data.drop\_duplicates()

print(new\_data.head())

**output:**

Table Code State Code District Code Area Name Total/ Rural/ Urban \

0 B0806SC `33 `000 State - TAMIL NADU Total

1 B0806SC `33 `000 State - TAMIL NADU Total

2 B0806SC `33 `000 State - TAMIL NADU Total

3 B0806SC `33 `000 State - TAMIL NADU Total

4 B0806SC `33 `000 State - TAMIL NADU Total

Age group Worked for 3 months or more but less than 6 months - Persons \

0 Total 1200828

1 `5-14 27791

2 15-34 514340

3 35-59 542581

4 60+ 115103

Worked for 3 months or more but less than 6 months - Males \

0 589003

1 14125

2 259560

3 251957

4 62833

Worked for 3 months or more but less than 6 months - Females \

0 611825

1 13666

2 254780

3 290624

4 52270

Worked for less than 3 months - Persons ... \

0 221386 ...

1 2447 ...

2 92423 ...

3 99202 ...

4 27165 ...

Industrial Category - N to O - Females \

0 3565

1 11

2 1754

3 1619

4 175

Industrial Category - P to Q - Persons \

0 11080

1 122

2 7536

3 3205

4 211

Industrial Category - P to Q - Males \

0 4019

1 71

2 2718

3 1131

4 93

Industrial Category - P to Q - Females \

0 7061

1 51

2 4818

3 2074

4 118

Industrial Category - R to U - HHI - Persons \

0 16833

1 427

2 8346

3 6591

4 1457

Industrial Category - R to U - HHI - Males \

0 4266

1 169

2 2127

3 1487

4 483

Industrial Category - R to U - HHI - Females \

0 12567

1 258

2 6219

3 5104

4 974

Industrial Category - R to U - Non HHI - Persons \

0 122088

1 19305

2 68929

3 26498

4 7065

Industrial Category - R to U - Non HHI - Males \

0 55801

1 9774

2 32803

3 9675

4 3394

Industrial Category - R to U - Non HHI - Females

0 66287

1 9531

2 36126

3 16823

4 3671

[5 rows x 69 columns]

**Program:**

sel**=**df[['Age group','Industrial Category - N to O - Females','Industrial Category - P to Q - Persons']]

print(sel)

**output**:

Age group Industrial Category - N to O - Females \

0 Total 3565

1 `5-14 11

2 15-34 1754

3 35-59 1619

4 60+ 175

.. ... ...

589 `5-14 0

590 15-34 20

591 35-59 33

592 60+ 0

593 Age not stated 0

Industrial Category - P to Q - Persons

0 11080

1 122

2 7536

3 3205

4 211

.. ...

589 0

590 44

591 35

592 3

593 0

[594 rows x 3 columns]

**Program:**

sel.to\_csv('preprocessed\_data.csv',index=False)

p\_df=pd.read\_csv("D:\\pythonproject\preprocessed\_data.csv")

print(p\_df.head())

**output:**

Age group Industrial Category - N to O - Females \

0 Total 3565

1 `5-14 11

2 15-34 1754

3 35-59 1619

4 60+ 175

Industrial Category - P to Q - Persons

0 11080

1 122

2 7536

3 3205

4 211

**Program:**

print(p\_df.columns)

**output:**

Index(['Age group', 'Industrial Category - N to O - Females', 'Industrial Category - P to Q - Persons'],dtype='object')

**Program:**

data=df.dropna()

print(data)

**output:**

Table Code State Code District Code Area Name \

0 B0806SC `33 `000 State - TAMIL NADU

1 B0806SC `33 `000 State - TAMIL NADU

2 B0806SC `33 `000 State - TAMIL NADU

3 B0806SC `33 `000 State - TAMIL NADU

4 B0806SC `33 `000 State - TAMIL NADU

.. ... ... ... ...

589 B0806SC `33 `633 District - Tiruppur

590 B0806SC `33 `633 District - Tiruppur

591 B0806SC `33 `633 District - Tiruppur

592 B0806SC `33 `633 District - Tiruppur

593 B0806SC `33 `633 District - Tiruppur

Total/ Rural/ Urban Age group \

0 Total Total

1 Total `5-14

2 Total 15-34

3 Total 35-59

4 Total 60+

.. ... ...

589 Urban `5-14

590 Urban 15-34

591 Urban 35-59

592 Urban 60+

593 Urban Age not stated

Worked for 3 months or more but less than 6 months - Persons \

0 1200828

1 27791

2 514340

3 542581

4 115103

.. ...

589 272

590 3285

591 3672

592 696

593 2

Worked for 3 months or more but less than 6 months - Males \

0 589003

1 14125

2 259560

3 251957

4 62833

.. ...

589 129

590 1654

591 1769

592 399

593 1

Worked for 3 months or more but less than 6 months - Females \

0 611825

1 13666

2 254780

3 290624

4 52270

.. ...

589 143

590 1631

591 1903

592 297

593 1

Worked for less than 3 months - Persons ... \

0 221386 ...

1 2447 ...

2 92423 ...

3 99202 ...

4 27165 ...

.. ... ...

589 18 ...

590 473 ...

591 522 ...

592 111 ...

593 0 ...

Industrial Category - N to O - Females \

0 3565

1 11

2 1754

3 1619

4 175

.. ...

589 0

590 20

591 33

592 0

593 0

Industrial Category - P to Q - Persons \

0 11080

1 122

2 7536

3 3205

4 211

.. ...

589 0

590 44

591 35

592 3

593 0

Industrial Category - P to Q - Males \

0 4019

1 71

2 2718

3 1131

4 93

.. ...

589 0

590 15

591 12

592 0

593 0

Industrial Category - P to Q - Females \

0 7061

1 51

2 4818

3 2074

4 118

.. ...

589 0

590 29

591 23

592 3

593 0

Industrial Category - R to U - HHI - Persons \

0 16833

1 427

2 8346

3 6591

4 1457

.. ...

589 0

590 62

591 36

592 10

593 0

Industrial Category - R to U - HHI - Males \

0 4266

1 169

2 2127

3 1487

4 483

.. ...

589 0

590 6

591 9

592 3

593 0

Industrial Category - R to U - HHI - Females \

0 12567

1 258

2 6219

3 5104

4 974

.. ...

589 0

590 56

591 27

592 7

593 0

Industrial Category - R to U - Non HHI - Persons \

0 122088

1 19305

2 68929

3 26498

4 7065

.. ...

589 228

590 675

591 279

592 81

593 0

Industrial Category - R to U - Non HHI - Males \

0 55801

1 9774

2 32803

3 9675

4 3394

.. ...

589 104

590 247

591 103

592 35

593 0

Industrial Category - R to U - Non HHI - Females

0 66287

1 9531

2 36126

3 16823

4 3671

.. ...

589 124

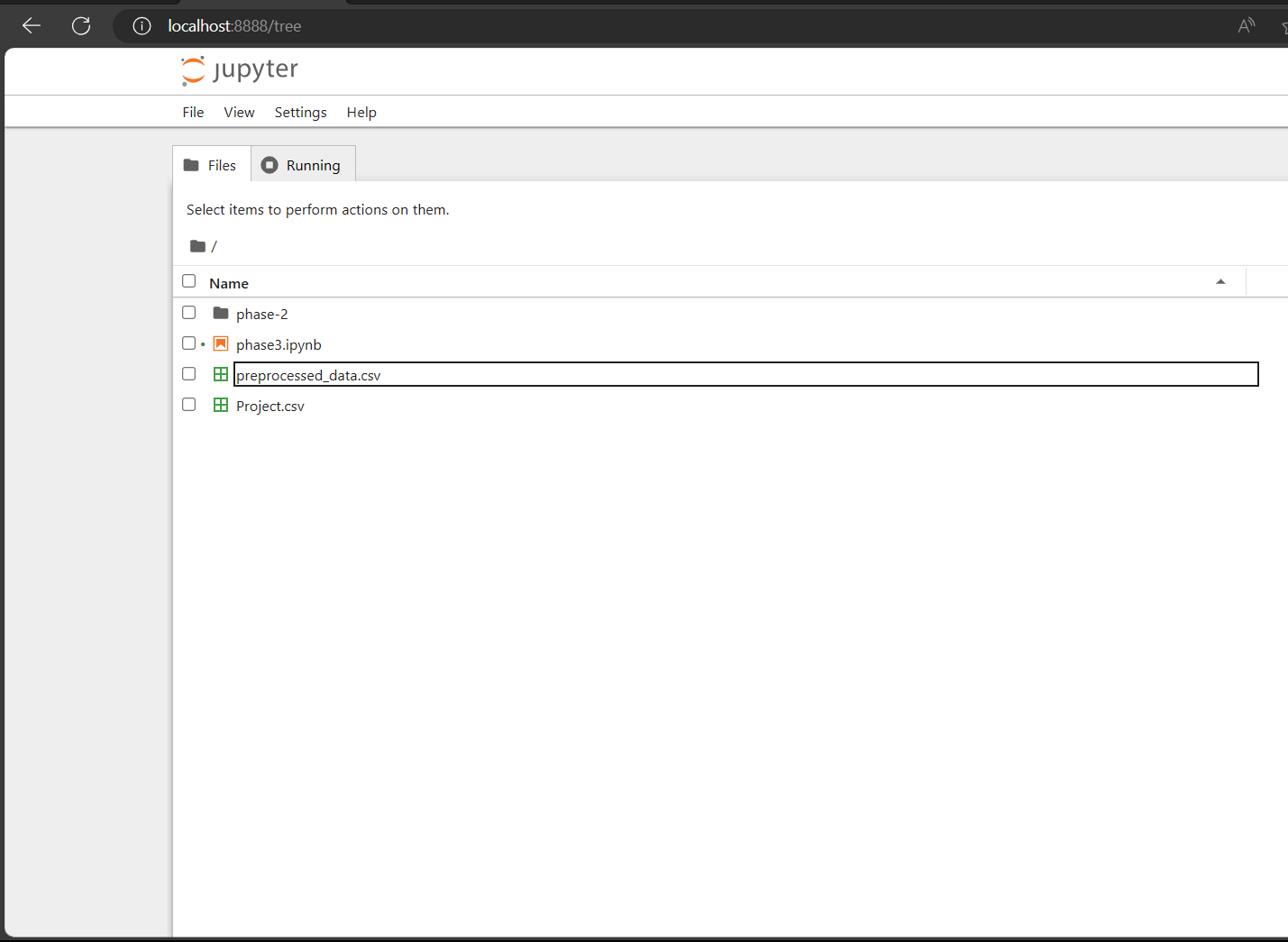
590 428

591 176

592 46

593 0

[594 rows x 69 columns]



**Conclusion:**

In conclusion, in this part of the project, we successfully loaded and preprocessed the dataset using Python and the pandaslibrary. We began by importing the necessary libraries and loading the dataset into a pandasDataFrame. We also explored the dataset's structure to gain insights into its contents, checked for missing values, and listed the column names.

Preprocessing steps were discussed but can be further customized based on the specific requirements of the analysis, such as handling missing values, encoding categorical variables, scaling numerical features, and other data cleaning and transformation tasks.