***Assessment of marginal workers in Tamil Nadu.***

**INTRODUCTION:**

**Loading the dataset involves reading the data into our Python environment so that we can analyze and manipulate it. Preprocessing steps are essential for cleaning the data, handling missing values, and preparing it for analysis.**

1. **Import Libraries**

First, import the necessary libraries, pandas for data manipulation and any other libraries you might need for your specific preprocessing tasks.

Import pandas as pd

# import other libraries as needed

1. **Load the Dataset:**

Use the `pd.read\_csv()` function (or similar functions for other formats like Excel, JSON, etc.) to load your dataset into a pandas DataFrame.

# Replace ‘your\_dataset.csv’ with the actual file path or URL of your dataset

Df = pd.read\_csv(‘your\_dataset.csv’)

1. **Data Exploration:**

Perform some basic exploratory data analysis (EDA) to understand the structure of your data. You can use functions like `head()`, `info()`, and `describe()` to get a glimpse of the data.

# Display the first few rows of the dataset

Print(df.head())

# Get information about the dataset (data types, non-null counts, etc.)

Print(df.info())

# Generate summary statistics

Print(df.describe())

1. **Data Preprocessing**:

Depending on your project and the dataset, you might need to preprocess the data. This can include handling missing values, converting data types, normalizing data, or encoding categorical variables.

# Handle missing values (drop or fill missing values)

Df.dropna(inplace=True) # Example: drop rows with missing values

# Convert data types if necessary (e.g., convert object columns to numeric)

# df[‘column\_name’] = pd.to\_numeric(df[‘column\_name’], errors=’coerce’)

# Perform other preprocessing tasks as needed (normalization, encoding, etc.)

Here’s an example Python program to load and preprocess a dataset using the pandas library:

**PROGRAM:**

# Import the pandas library for data manipulation

Import pandas as pd

# Load the dataset into a pandas DataFrame

# Replace ‘your\_dataset.csv’ with the actual file path or URL of your dataset

Df = pd.read\_csv(‘your\_dataset.csv’)

# Display the first few rows of the dataset

Print(“First 5 rows of the dataset:”)

Print(df.head())

# Check for missing values and handle them (drop or fill missing values)

# For example, drop rows with missing values

Df.dropna(inplace=True)

# Convert data types if necessary (e.g., convert object columns to numeric)

# For example, convert a column named ‘column\_name’ to numeric

# df[‘column\_name’] = pd.to\_numeric(df[‘column\_name’], errors=’coerce’)

# Perform other preprocessing tasks as needed (normalization, encoding, etc.)

# Display the preprocessed data

Print(“Preprocessed dataset:”)

Print(df.head())

In this program, replace `’your\_dataset.csv’` with the actual path or URL of your dataset file. The program reads the dataset into a pandas DataFrame, displays the first few rows of the original dataset, handles missing values by dropping rows with missing values (you can choose a different method based on your requirements), and performs other preprocessing tasks as needed.

**CONCLUSION:**

**At this point, you have successfully loaded and preprocessed your dataset, making it ready for further analysis or modeling .**