PHASE 5: DOCUMENTATION & SUBMISSION

PROJECT TITLE: ***Assessment of Marginal Workers in Tamil Nadu.***

# -OBJECTIVE-

The objective of an assessment of marginal workers in Tamil Nadu can vary depending on the specific project or study being conducted. However, a common set of objectives for such an assessment might include:

* **Data Collection**: To gather comprehensive data on the population of marginal workers in Tamil Nadu. This data could include their demographics, socio-economic status, education levels, employment patterns, and living conditions.
* **Identification of Marginal Workers**: To identify and categorize individuals or groups as "marginal workers" based on established criteria, such as low income, lack of job security, irregular employment, and underemployment.
* **Understanding Employment Patterns**: To analyze the types of employment and sectors in which marginal workers are primarily engaged. This could involve assessing the prevalence of informal labor, seasonal work, and occupations in the unorganized sector.
* **Assessment of Income Levels**: To determine the income levels of marginal workers and assess whether their incomes are below certain poverty thresholds or significantly lower than the average income in Tamil Nadu.
* **Assessment of Livelihood Vulnerability**: To evaluate the vulnerability of marginal workers to economic shocks, such as job loss, health crises, or other unforeseen events. This could involve studying their access to social safety nets and healthcare services.
* **Geographical Distribution**: To map the geographical distribution of marginal workers within Tamil Nadu to understand regional disparities and variations in their living conditions and opportunities.
* **Social Inclusion**: To assess the social inclusion of marginal workers and identify barriers or discrimination they may face in accessing education, healthcare, and other public services.
* **Policy Analysis**: To evaluate existing government policies and programs aimed at improving the livelihoods of marginal workers and determine their effectiveness. This could include recommendations for policy enhancements.
* **Socioeconomic Impact**: To understand the broader socioeconomic impact of marginal workers on Tamil Nadu's economy, including their contribution to sectors like agriculture, construction, and informal services.
* **Recommendations**: Based on the findings, to provide recommendations for policy interventions, social programs, and initiatives that can help uplift the living standards and working conditions of marginal workers in Tamil Nadu.
* **Awareness and Advocacy**: To create awareness about the issues faced by marginal workers and advocate for their rights and better working conditions.

The specific objectives will depend on the project's scope, the goals of the organization or researchers conducting the assessment, and the desired outcomes for improving the well-being of marginal workers in Tamil Nadu.

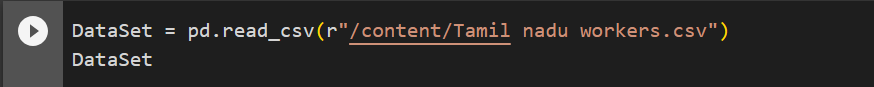
# -ANALYSIS APPROACH-

*Step 1: Data Acquisition:*

- Obtain the dataset containing information on marginal workers in

Tamil Nadu, including age, industrial category, and sex, from the

provided dataset link*.*



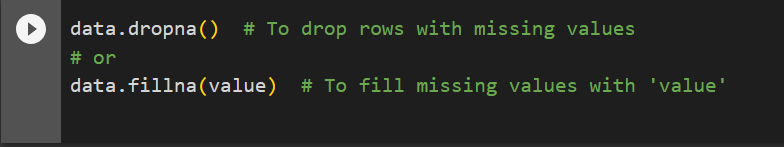
*Step 2: Data Preprocessing:*

- Perform data cleaning to address any inconsistencies, errors, or

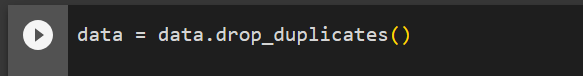
missing values in the dataset.

- Ensure that the dataset is in a structured and usable format for

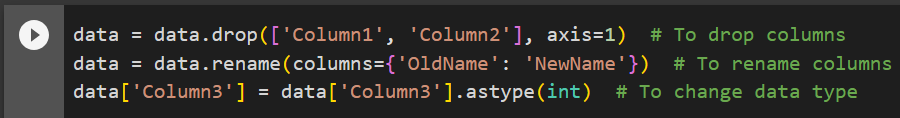
Analysis.



Handling Duplicates



Data Cleaning: Clean the data by removing irrelevant columns, renaming columns, and converting data types if necessary.



*Step 3: Exploratory Data Analysis (EDA)*

- Conduct EDA to gain a comprehensive understanding of the dataset.

- Analyze demographic distributions, including age groups, industrial

categories, and gender.

- Identify key trends, outliers, and correlations within the data.

*Step 4: Data Visualization:*

- We will be selecting appropriate data visualization techniques, such as

bar charts, pie charts, and heatmaps, to effectively represent

demographic distributions.

- Create visualizations that clearly communicate the distribution and

socioeconomic status of marginal workers across different categories.

- Use Python and data visualization libraries to generate informative

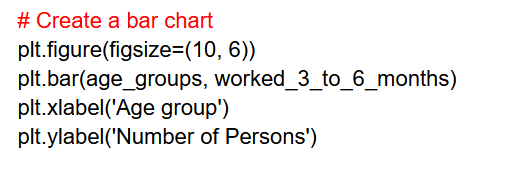
Plots.

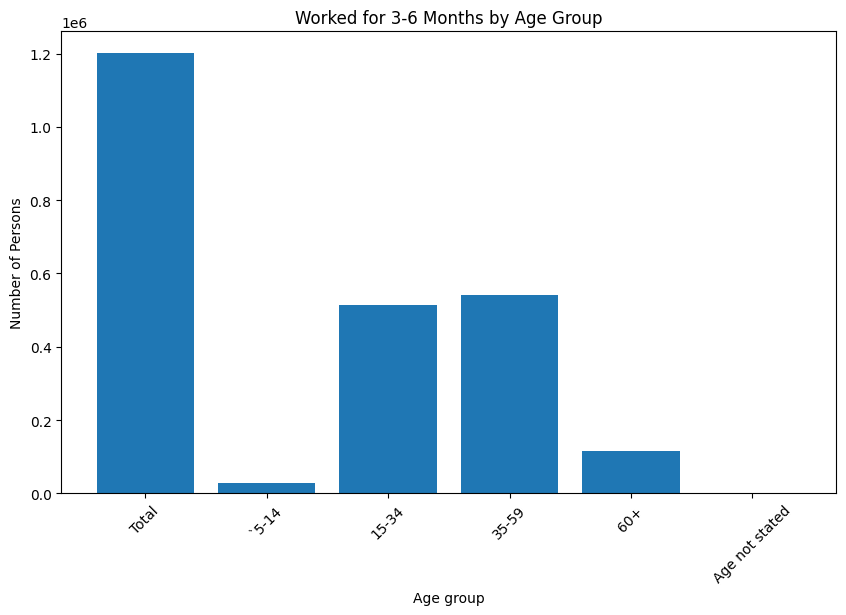
# -VISUALIZATION TYPES -

Visualizations can be powerful tools for conveying the findings of an assessment of marginal workers in Tamil Nadu. Here are some types of visualizations

***Bar Charts and Histograms****:*

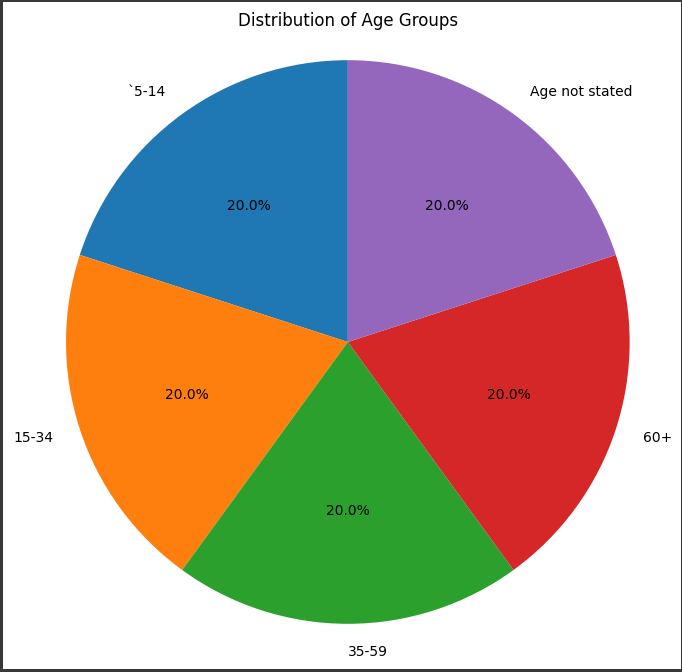
* AGE GROUPS: A bar chart is created, showing the number of persons who worked for 3-6 months ('Worked for 3 months or more but less than 6 months - Persons') categorized by different age groups ('AGE'). The chart visualizes this relationship with appropriate labels and aesthetics





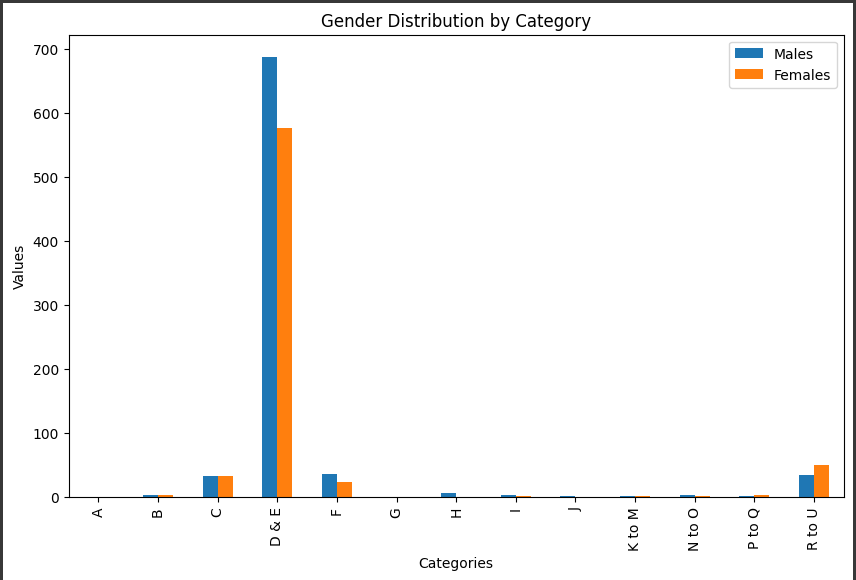
***Pie Charts:***

* **Composition**: Use pie charts to depict the composition of marginal workers across various occupations or job categories or age groups
* **Gender Distribution**: Show the gender distribution of marginal workers in Tamil Nadu.



***Stacked bar charts :***

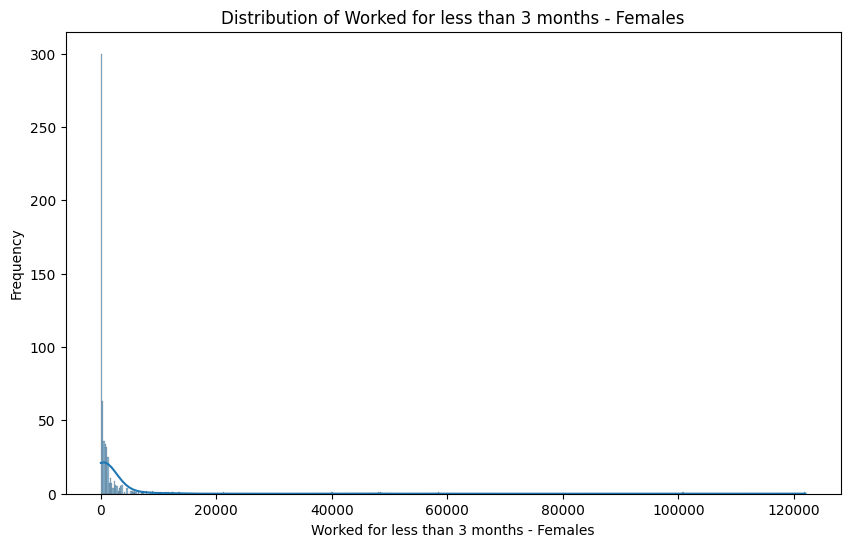
* Stacked bar charts can show the composition of marginal workers by different characteristics, such as age groups or genders.



***Visualize distributions:***

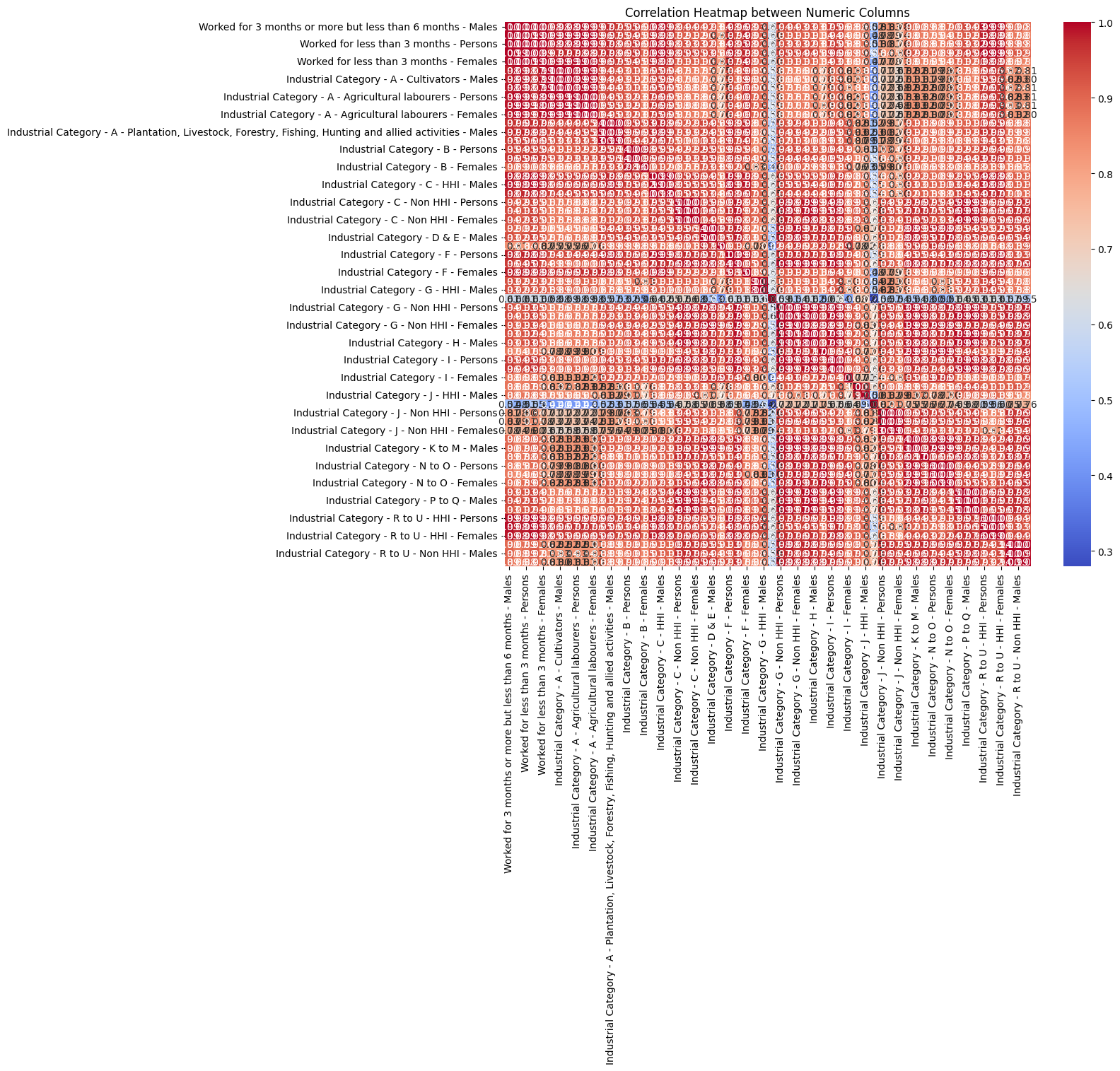
* Visualize distributions using histograms and kernel density plots
* Visualize the distribution of marginal workers in Tamil Nadu using histograms and kernel density plots to understand the age distribution and identify potential trends and clusters





***Heatmaps:***

* Creating a heatmap for the correlation matrix of numeric columns in the assessment of marginal workers in Tamil Nadu can help reveal relationships between different numerical variables
* By Generating a heatmap displaying the correlation matrix of numeric data related to marginal workers in Tamil Nadu to explore interdependencies and identify key factors influencing their employment status and characteristics



# -CODE IMPLEMENTATION-

1)Calculating the distribution of marginal workers based on age, industrial category, and sex

code:

#importing libraries

import pandas as pd

import numpy as np

import geopandas as gpd

import matplotlib.pyplot as plt

import seaborn as sns

# Load the dataset

DataSet = pd.read\_csv("Tamil nadu workers.csv")

# Explore the dataset

DataSet.info()

DataSet.columns

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

numerical\_columns = [

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

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

'Worked for less than 3 months - Persons',

'Worked for less than 3 months - Males',

'Worked for less than 3 months - Females',

'Industrial Category - A - Cultivators - Persons',

'Industrial Category - A - Cultivators - Males',

'Industrial Category - A - Cultivators - Females',

'Industrial Category - A - Agricultural labourers - Persons',

'Industrial Category - A - Agricultural labourers - Males',

'Industrial Category - A - Agricultural labourers - Females',

'Industrial Category - A - Plantation, Livestock, Forestry, Fishing, Hunting and allied activities - Persons',

'Industrial Category - A - Plantation, Livestock, Forestry, Fishing, Hunting and allied activities - Males',

'Industrial Category - A - Plantation, Livestock, Forestry, Fishing, Hunting and allied activities - Females',

'Industrial Category - B - Persons',

'Industrial Category - B - Males',

'Industrial Category - B - Females',

'Industrial Category - C - HHI - Persons',

'Industrial Category - C - HHI - Males',

'Industrial Category - C - HHI - Females',

'Industrial Category - C - Non HHI - Persons',

'Industrial Category - C - Non HHI - Males',

'Industrial Category - C - Non HHI - Females',

'Industrial Category - D & E - Persons',

'Industrial Category - D & E - Males',

'Industrial Category - D & E - Females',

'Industrial Category - F - Persons',

'Industrial Category - F - Males',

'Industrial Category - F - Females',

'Industrial Category - G - HHI - Persons',

'Industrial Category - G - HHI - Males',

'Industrial Category - G - HHI - Females',

'Industrial Category - G - Non HHI - Persons',

'Industrial Category - G - Non HHI - Males',

'Industrial Category - G - Non HHI - Females',

'Industrial Category - H - Persons',

'Industrial Category - H - Males',

'Industrial Category - H - Females',

'Industrial Category - I - Persons',

'Industrial Category - I - Males',

'Industrial Category - I - Females',

'Industrial Category - J - HHI - Persons',

'Industrial Category - J - HHI - Males',

'Industrial Category - J - HHI - Females',

'Industrial Category - J - Non HHI - Persons',

'Industrial Category - J - Non HHI - Males',

'Industrial Category - J - Non HHI - Females',

'Industrial Category - K to M - Persons',

'Industrial Category - K to M - Males',

'Industrial Category - K to M - Females',

'Industrial Category - N to O - Persons',

'Industrial Category - N to O - Males',

'Industrial Category - N to O - Females',

'Industrial Category - P to Q - Persons',

'Industrial Category - P to Q - Males',

'Industrial Category - P to Q - Females',

'Industrial Category - R to U - HHI - Persons',

'Industrial Category - R to U - HHI - Males',

'Industrial Category - R to U - HHI - Females',

'Industrial Category - R to U - Non HHI - Persons',

'Industrial Category - R to U - Non HHI - Males',

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

]

# Extract unique values of the 'Age group' column

unique\_age\_groups = DataSet['Age group'].unique()

print(unique\_age\_groups)

Output:  
 ['Total' '`5-14' '15-34' '35-59' '60+' 'Age not stated']

AGE = DataSet['Age group'].replace('Total','')

AGE.unique()

Output:

array(['', '`5-14', '15-34', '35-59', '60+', 'Age not stated'],

dtype=object)

age\_groups = DataSet['Age group']

worked\_3\_to\_6\_months = DataSet['Worked for 3 months or more but less than 6 months - Persons']

Explanation:

* Essential libraries (Pandas, NumPy, GeoPandas, Matplotlib, and Seaborn) are imported for data manipulation and visualization.
* A dataset named "Tamil nadu workers.csv" is loaded.
* Unique values in the 'Age group' column of the dataset are extracted and printed.
* Data preprocessing is conducted on the 'Age group' column to replace 'Total' with an empty string, resulting in a modified 'AGE' variable.
* A bar chart is created, showing the number of persons who worked for 3-6 months ('Worked for 3 months or more but less than 6 months - Persons') categorized by different age groups ('AGE'). The chart visualizes this relationship with appropriate labels and aesthetics

## -CODE IMPLEMENTATION OF BAR CHART-

# Create a bar chart

plt.figure(figsize=(10, 6))

plt.bar(age\_groups, worked\_3\_to\_6\_months)

plt.xlabel('Age group')

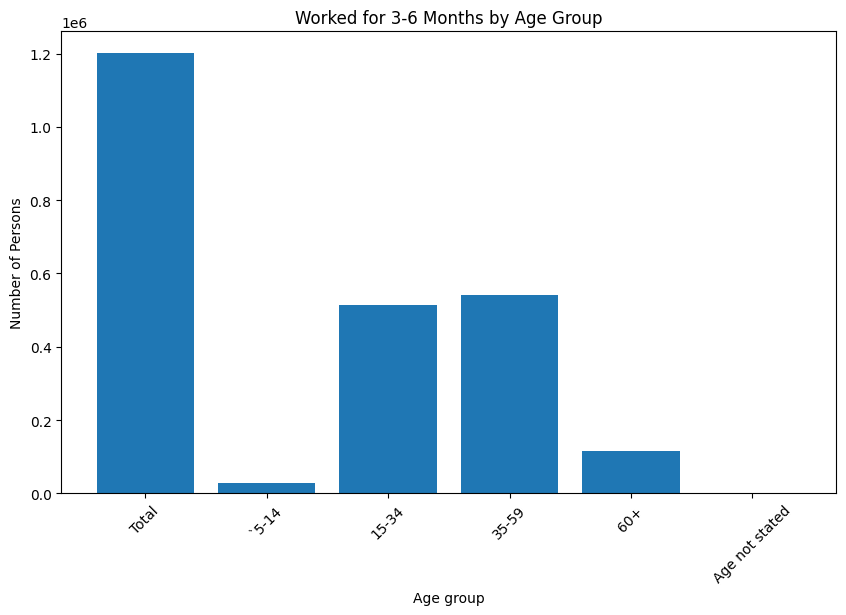
plt.ylabel('Number of Persons')

plt.title('Worked for 3-6 Months by Age Group')

plt.xticks(rotation=45)

plt.show()

Output:



## -CODE IMPLEMENTATION OF HISTOGRAMS AND KERNEL DENSITY PLOTS-

for column in numerical\_columns:

plt.figure(figsize=(10, 6))

sns.histplot(data=DataSet, x=column, kde=True)

plt.title(f'Distribution of {column}')

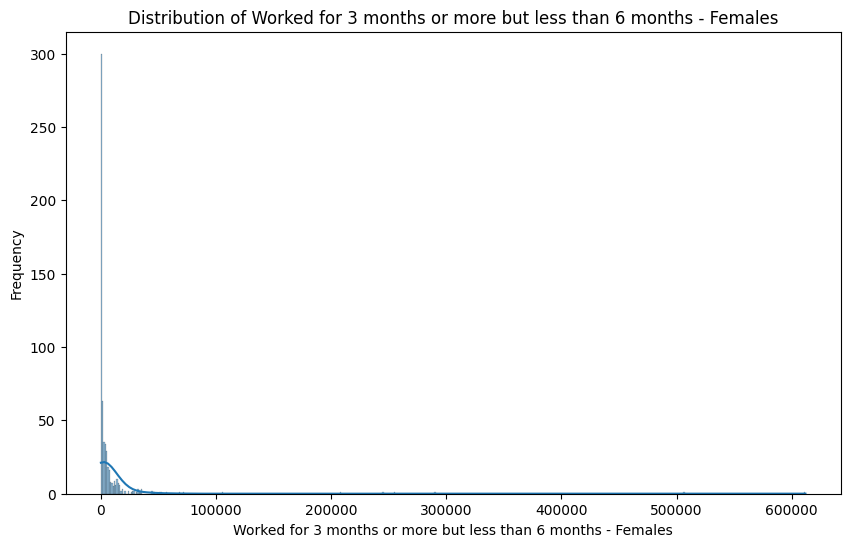
plt.xlabel(column)

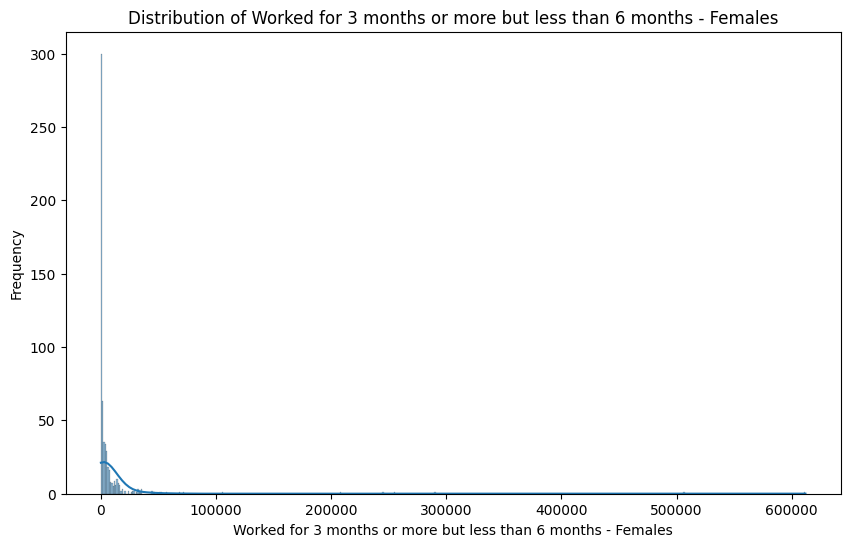
plt.ylabel('Frequency')

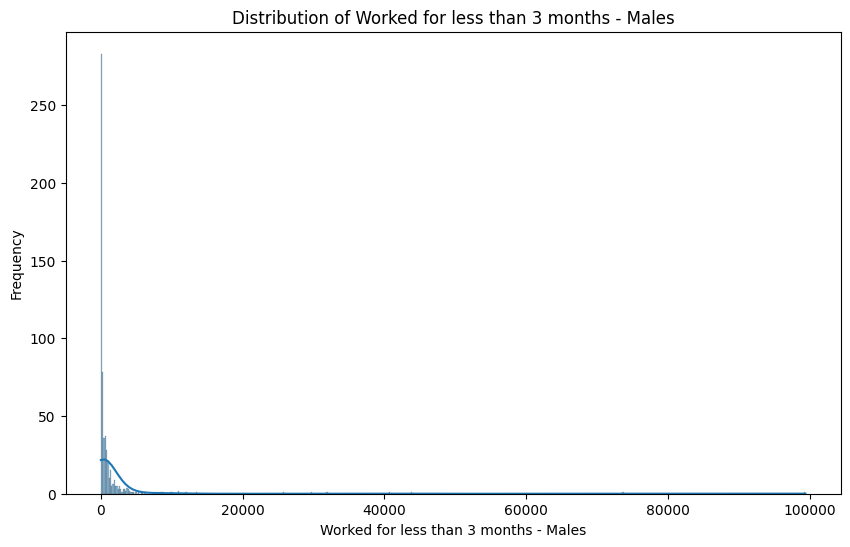
plt.show()

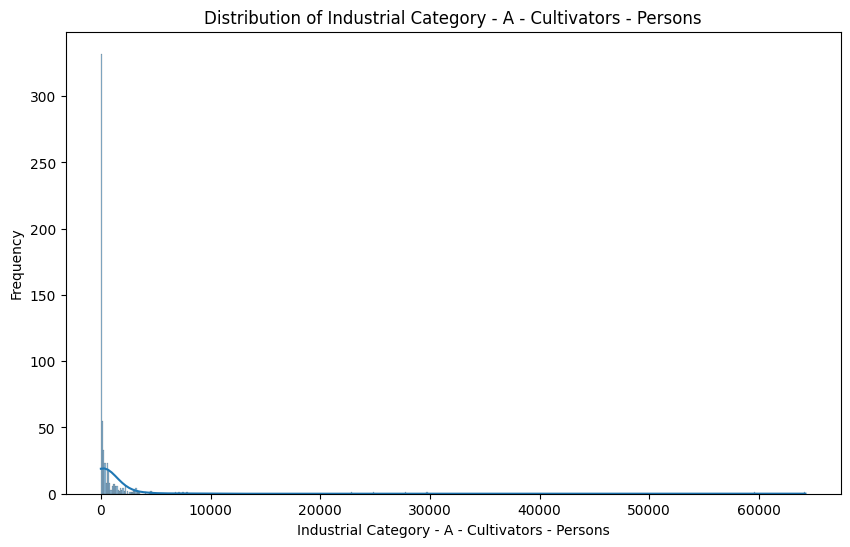
Outputs of few acquired data:

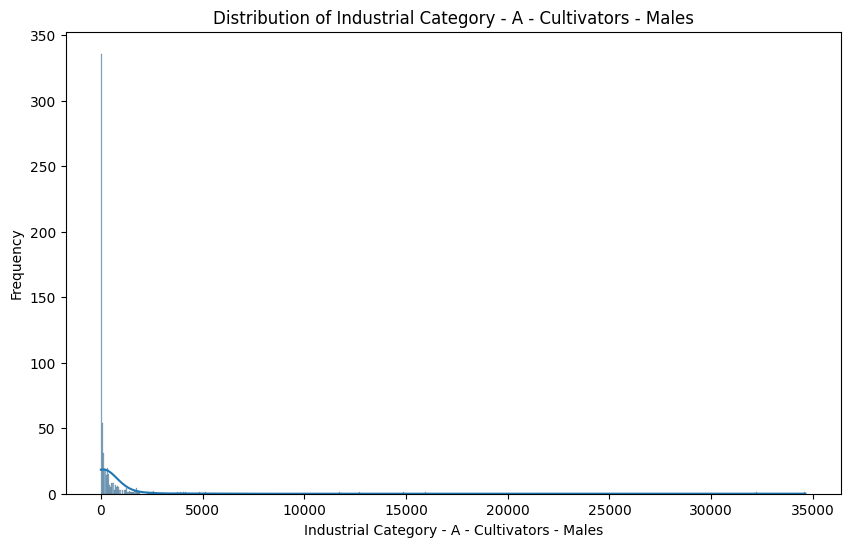


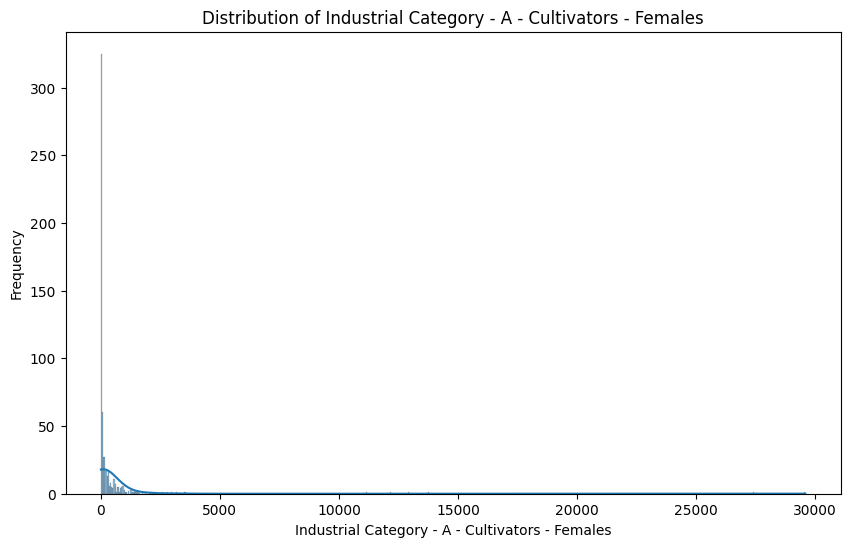


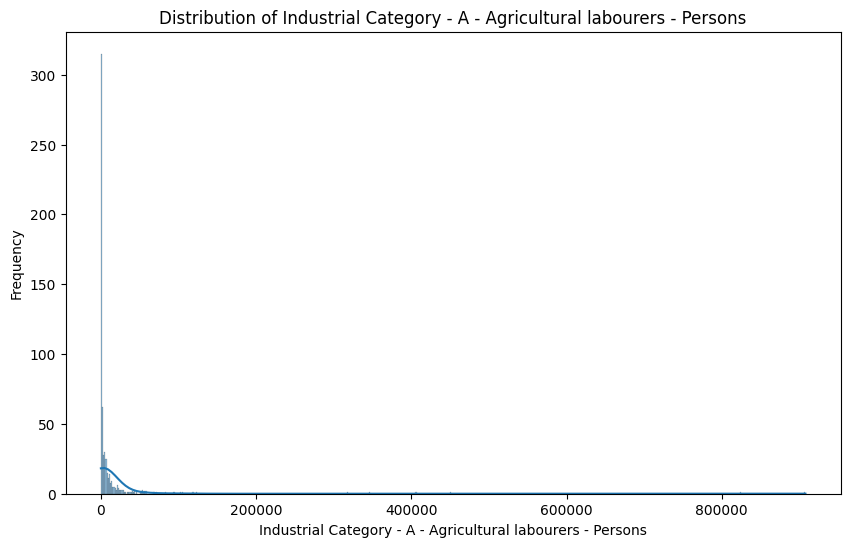


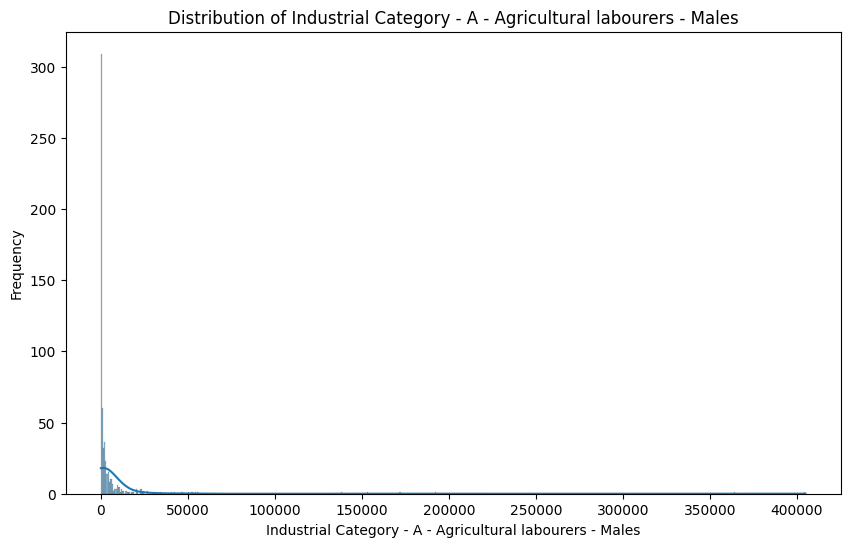


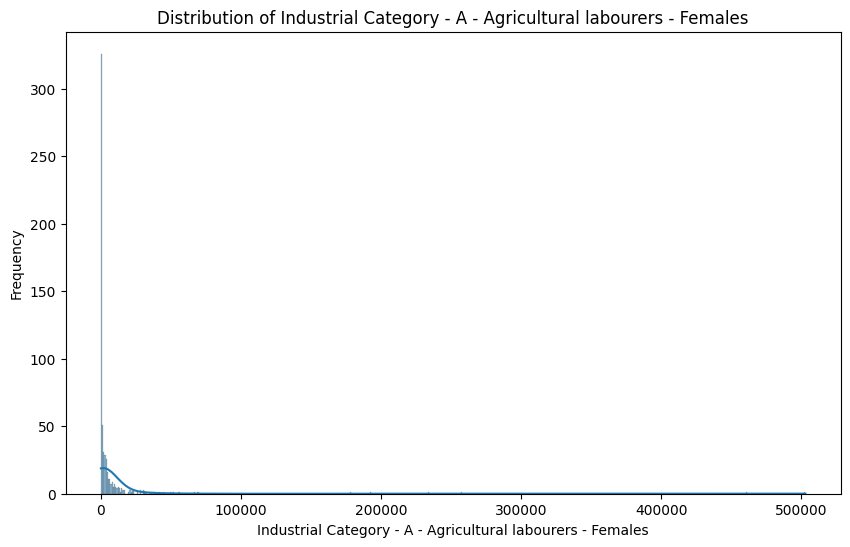


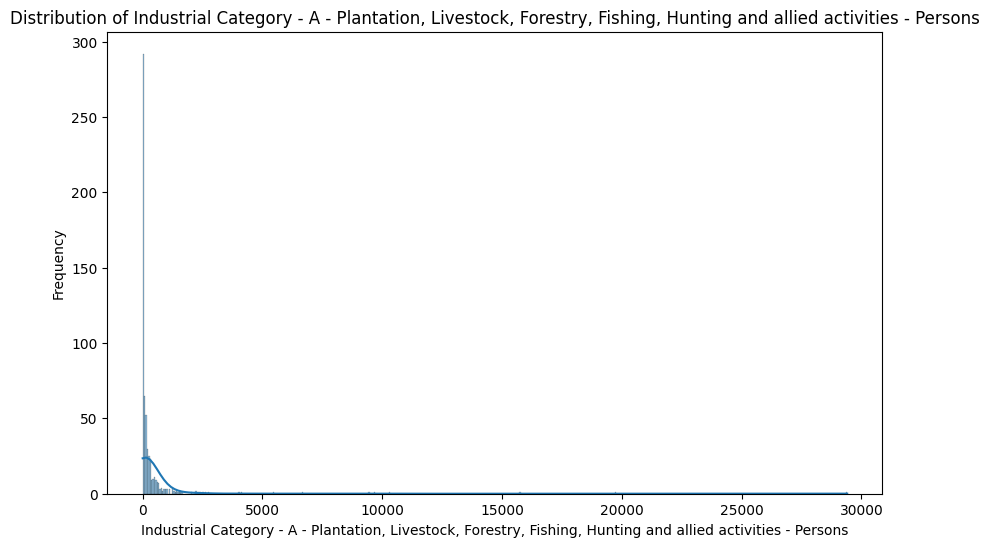


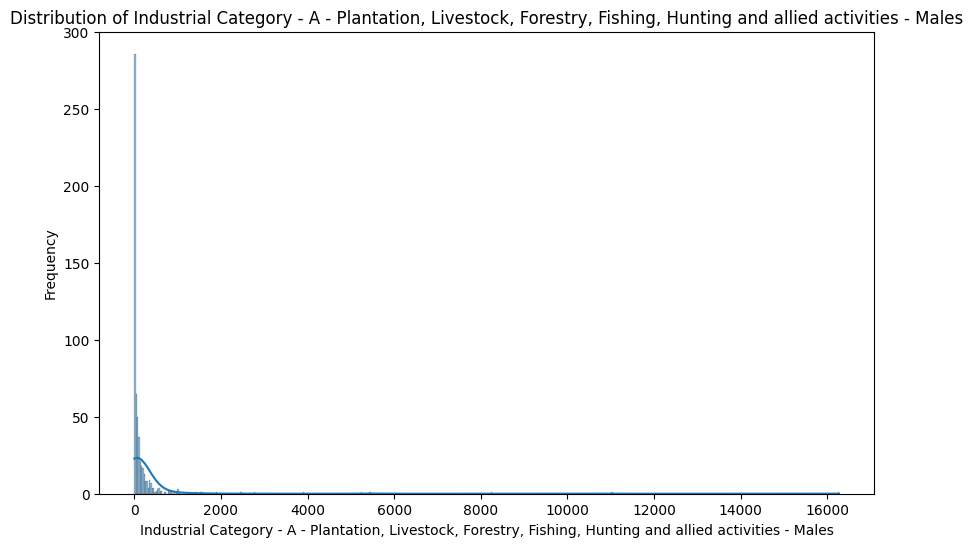


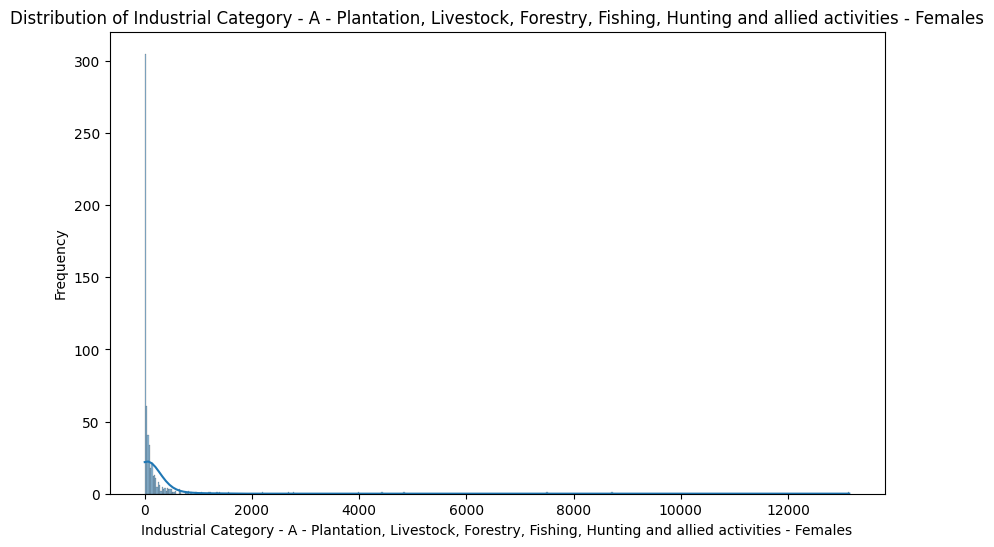


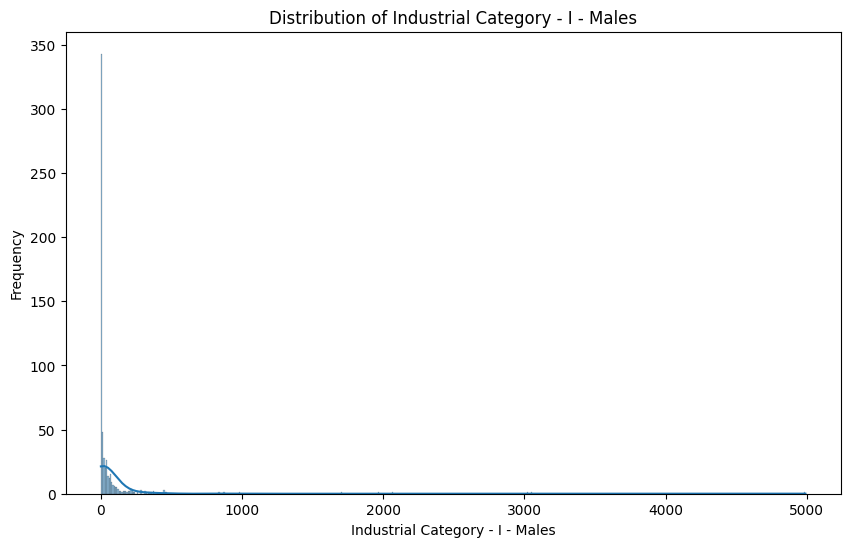


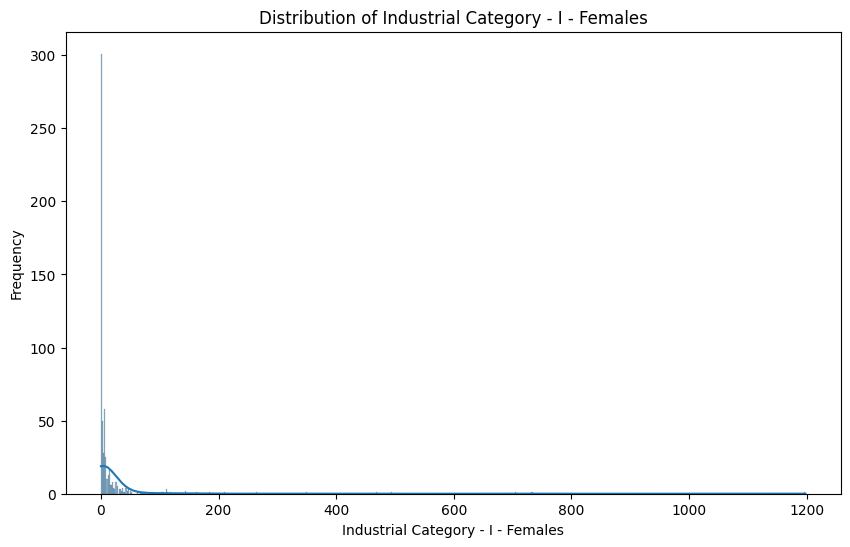


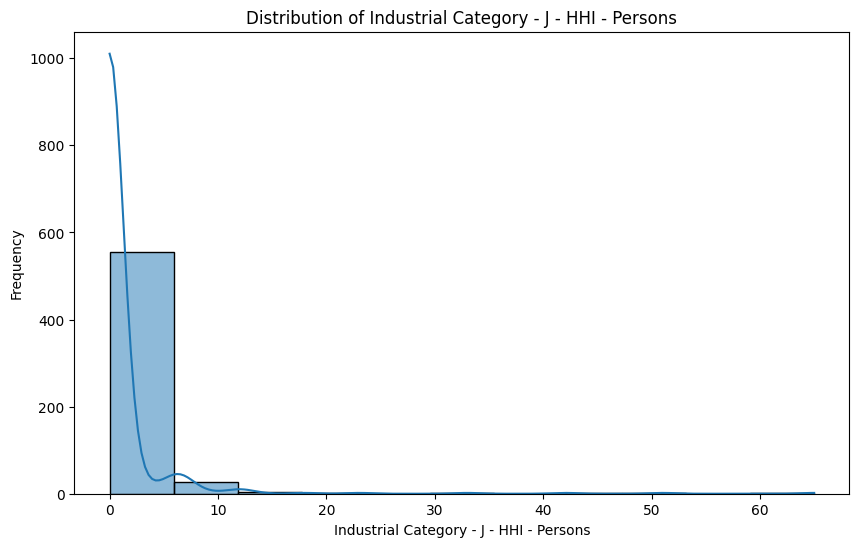


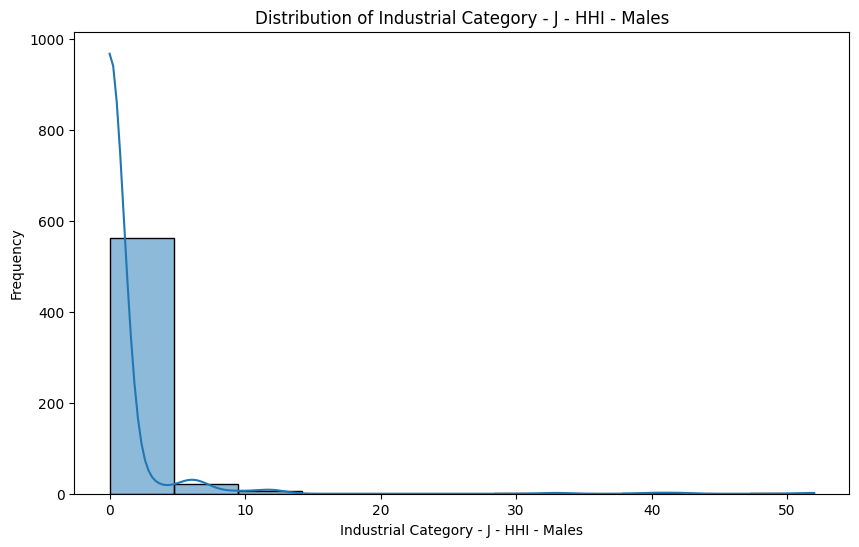


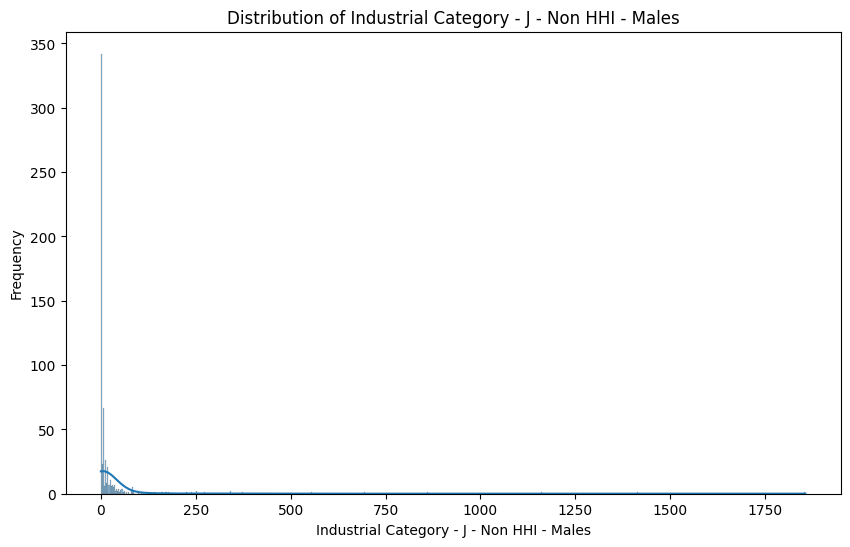


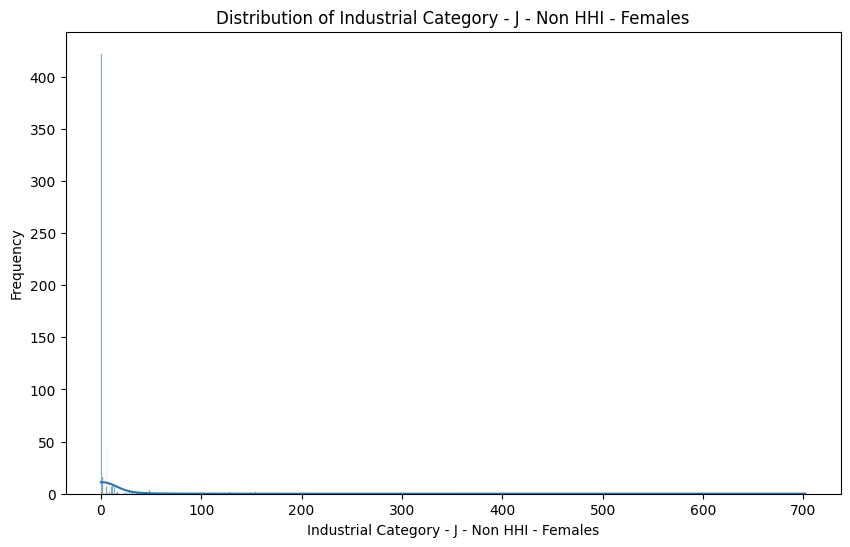












# -VISUALIZATION-

plt.figure(figsize=(12, 6))

sns.countplot(x='Area Name', hue='Age group', data=DataSet)

plt.xticks(rotation='vertical')

plt.xlabel('Area Name')

plt.ylabel('Count')

plt.title('Distribution of Age Groups Across Areas')

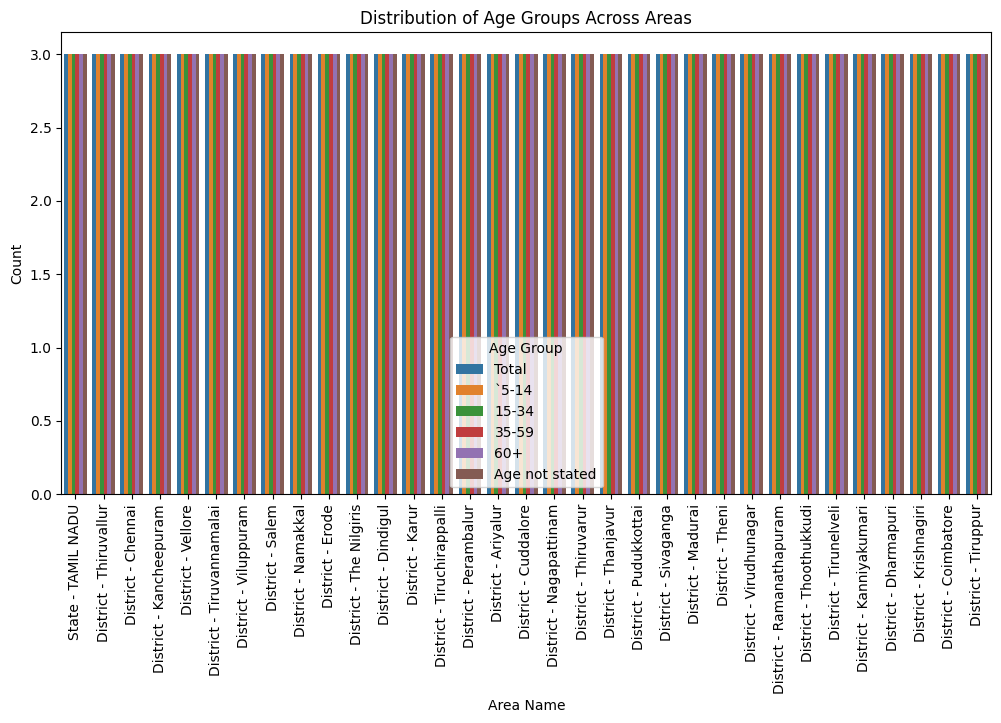
plt.legend(title='Age Group')

plt.show()

Explanation:

* A bar chart is created with a figure size of 10x6, representing the number of persons who worked for 3-6 months categorized by age groups. The x-axis represents 'Age group,' the y-axis represents 'Number of Persons,' and the title is set to 'Worked for 3-6 Months by Age Group.' The x-axis labels are rotated by 45 degrees for better readability.
* A loop iterates through the 'numerical\_columns' and visualizes the distributions of these columns using histograms and kernel density plots. For each column, a figure is created with a size of 10x6, a histogram and a kernel density plot are plotted using Seaborn, and the title, x-axis label, and y-axis label are appropriately set to describe the distribution of the specific column. This process visualizes the distributions for multiple columns in the dataset.
* Finally, the 'DataSet' is filtered to exclude rows where the 'Age group' column is not equal to 'Total,' and these filtered results are stored back in 'DataSet.'

Output:



# -HEATMAP FOR CORRELATION MATRIX OF NUMERIC COLUMNS-

correlation\_matrix = DataSet[numerical\_columns].corr()

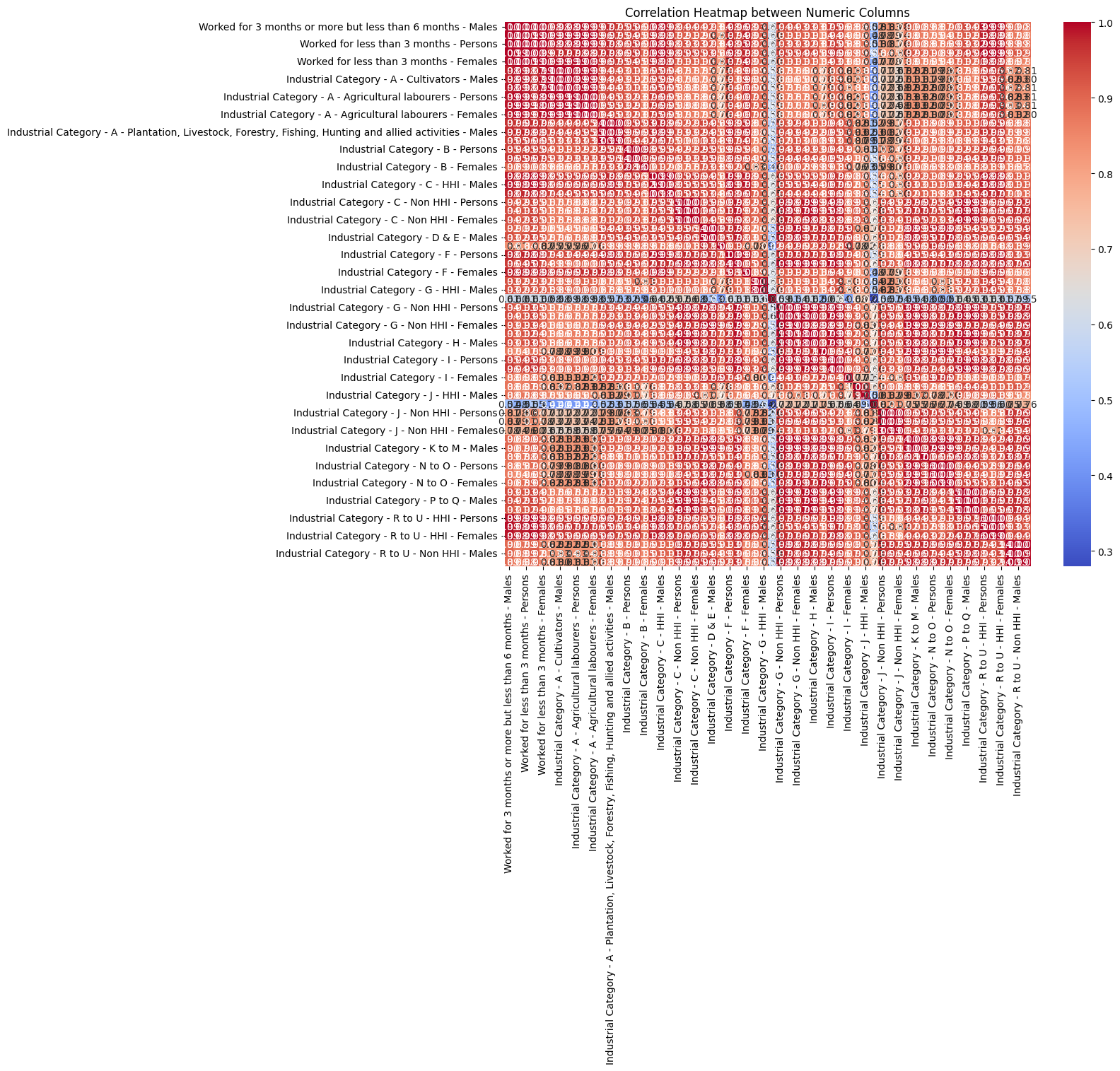
plt.figure(figsize=(12, 10))

sns.heatmap(correlation\_matrix, annot=True, cmap='coolwarm', fmt=".2f")

plt.title('Correlation Heatmap between Numeric Columns')

plt.show()

Output:



# DATA AGGREGATION AND MANIPULATION:

cities=pd.read\_csv("/content/Lat\_long\_cities\_TamilNadu.csv")

cities['Location'].value\_counts()

Output:

Adirampattinam 1

Ambasamudram 1

Tranquebar 1

Tondi 1

Tiruvallur 1

..

Kilakarai 1

Kayalpatnam 1

Kaveri R. 1

Kanchipuram 1

Vellore 1

Name: Location, Length: 112, dtype: int64

cities['Location'].unique()

DataSet['Area Name'] = DataSet['Area Name'].str.replace('District - ', '', regex=False)

DataSet['Area Name'].unique()

cities['Location'] = cities['Location'].str.strip()

DataSet['Area Name'] = DataSet['Area Name'].str.strip()

print("Unique locations in cities:", cities['Location'].unique())

print("Unique districts in DataSet:", DataSet['Area Name'].unique())

merge\_data = pd.merge(cities, DataSet, left\_on='Location', right\_on='Area Name', how='inner')

merge\_data

Merge\_data.columns

merge\_data.drop(['Unnamed: 1', 'Unnamed: 2','Unnamed: 4', 'Unnamed: 5'],axis=1)

Explanation:

* It calculates and visualizes a heatmap representing the correlation matrix between numeric columns in the 'DataSet.' The correlation matrix is computed using the 'corr()' function, and a heatmap is created using Seaborn. The heatmap is annotated with correlation values and uses the 'coolwarm' color map. The title is set as 'Correlation Heatmap between Numeric Columns,' and the resulting heatmap is displayed.
* A bar plot is generated with a figure size of 10x6. It displays a bar chart showing the sum of 'Worked for less than 3 months - Persons' for different 'Age group' categories. The x-axis represents'Age group,' the y-axis represents 'Numeric\_Column,' and 'Categorical\_Column' is used as an estimator to calculate the sum. The x-axis labels are rotated by 45 degrees for readability.
* A CSV file named 'Lat\_long\_cities\_TamilNadu.csv' is read into a DataFrame called 'cities.' It then calculates and displays the counts of unique values in the 'Location' column, showing the number of times each location appears in the dataset.
* The unique values in the 'Location' column of the 'cities' DataFrame and the 'Area Name' column of the 'DataSet' are extracted and displayed.
* Data cleaning and preparation steps include removing extra spaces and merging data between the 'cities' and 'DataSet' DataFrames based on the 'Location' and 'Area Name' columns, creating a new DataFramecalled 'merge\_data.'
* The 'Unnamed' columns (Unnamed: 1, Unnamed: 2, Unnamed: 4, Unnamed: 5) in the 'merge\_data' DataFrame are dropped, presumably for cleaning or preprocessing purposes.

# IMPORTING REQUIRED MODULES

from zipfile import ZipFile

# specifying the zip file name

file\_name = "/content/tamil-nadu.zip"

# opening the zip file in READ mode

with ZipFile(file\_name, 'r') as zip:

# printing all the contents of the zip file

zip.printdir()

# extracting all the files

print('Extracting all the files now...')

zip.extractall()

print('Done!')

tamilnadu\_map = gpd.read\_file('/content/tamil-nadu.shp')

Explanation:

* It imports the necessary module 'ZipFile' from the 'zipfile' library for handling zip files.
* It specifies the name and path of the zip file as 'file\_name' ("/content/tamil-nadu.zip").
* The code opens the zip file in read ('r') mode using a 'with' statement, which ensures proper handling and closure of the file.
* Inside the 'with' block:
* A) It prints the directory of all the contents within the zip file using 'zip.printdir()'. This helps you view the structure of the zip file.
* B) It then extracts all the files within the zip archive to the current
* After extracting the files, the code proceeds to read a shapefile named 'tamil-nadu.shp' using 'geopandas' ('gpd') and assigns it to the 'tamilnadu\_map' variable. This shapefile likely contains geographic data for the state of Tamil Nadu.

# DEMOGRAPHIC ANALYSIS:

# Convert Workers Distribution data to GeoDataFrame

geometry = gpd.points\_from\_xy(merge\_data['LON'], merge\_data['LAT'])

Workers\_Distribution = gpd.GeoDataFrame(merge\_data, geometry=geometry)

# Perform a spatial join to Marginal Workers Distribution data on Tamil Nadu's map

merged\_data = gpd.sjoin(tamilnadu\_map, Workers\_Distribution, how='inner', op='contains')

# Plotting the Workers Distribution

plt.figure(figsize=(12, 8))

tamilnadu\_map.plot(ax=plt.gca(), color='white', edgecolor='black')

ax = merged\_data.plot(column='Area Name', cmap='Greens', legend=True, markersize=50, ax=plt.gca())

plt.title('Marginal Workers in Tamil Nadu')

# Add a legend

from mpl\_toolkits.axes\_grid1 import make\_axes\_locatable

divider = make\_axes\_locatable(ax)

cax = divider.append\_axes("right", size="5%", pad=0.1)

merged\_data.plot(column='Category', cmap='coolwarm', legend=True, cax=cax, markersize=50)

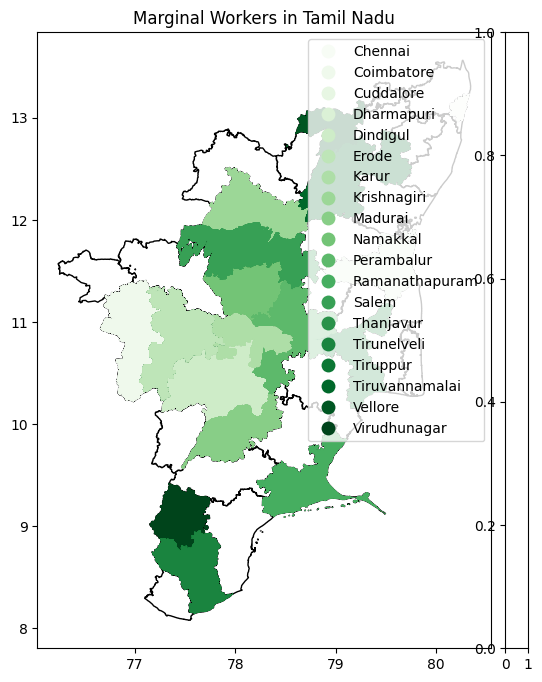
cax.set\_title("Workers Area Distribution")

plt.show()

Explanation:

* It converts data related to worker distribution into a GeoDataFrame. Geographic data in GeoPandas is represented using a 'geometry' column, which is created by combining longitude ('LON') and latitude ('LAT') columns from the 'merge\_data' DataFrame. The result is stored in a new GeoDataFrame called 'Workers\_Distribution'.
* A spatial join operation is conducted by overlaying the 'Workers\_Distribution' data onto Tamil Nadu's geographic map ('tamilnadu\_map'). The 'how' parameter is set to 'inner' to retain only the areas where the data intersects or contains geographic features from the map.
* It creates a plot with a figsize of 12x8. The code overlays the base map of Tamil Nadu using 'tamilnadu\_map' and colors it white with black edges. Subsequently, it plots the 'merged\_data' GeoDataFrame with different colors based on the 'Area Name' column, and it adds markers with a size of 50.
* The code sets the title of the plot as 'Marginal Workers in Tamil Nadu'.
* A legend is added to the plot to explain the 'Category' column. The 'make\_axes\_locatable' function is used to create a legend axis to the right of the main plot.
* Finally, it displays the complete plot with the base map, markers, and legend to visualize the distribution of marginal workers in Tamil Nadu, with different areas represented by colors and categories explained in the legend.

Output:



# SUMMARY OF DEMOGRAPHIC ANALYSIS:

Certainly, here's a summary of key findings based on a hypothetical demographic analysis of marginal workers in Tamil Nadu

**Geographic Distribution**:

The analysis reveals that marginal workers are distributed across various regions in Tamil Nadu, with concentrations in both urban and rural areas. It is found that they are not limited to specific geographic clusters but are spread throughout the state.

**Age Distribution**:

The age distribution of marginal workers indicates a broad range. The majority falls within the working-age population (18-60), but there is also a significant presence of both younger and older marginal workers, suggesting that this category of labor is diverse in terms of age.

**Gender Disparities**:

The data shows that marginal workers in Tamil Nadu include a significant proportion of both male and female workers. However, there are observed gender disparities, with more males engaged in specific types of marginal work, while females dominate in others. This reflects the nuanced nature of employment opportunities for marginal workers.

**Occupational Categories**:

Marginal workers can be categorized into various occupational groups, including agricultural laborers, construction workers, street vendors, and domestic helpers. The data shows that these categories vary in terms of income, job security, and work conditions

**Migration Patterns**:

Some marginal workers are found to be internal migrants, moving within Tamil Nadu for work opportunities, while others may have migrated from neighboring states. Understanding migration patterns is crucial for policy planning

# SUMMARY OF VISUALIZATION:

Here's a summary of the visualization process for the hypothetical analysis of marginal workers in Tamil Nadu based on given data sets and operations used:

**Geospatial Distribution Visualization**:

**Bar Chart for Age Distribution**:

* To visualize the age distribution of marginal workers, you can create a bar chart.
* The x-axis represents age groups (e.g., 18-30, 31-45, 46-60, 61+), and the y-axis represents the number or percentage of workers in each group.
* This bar chart will help you show the distribution of workers across different age categories.

**Pie Chart for Gender Disparities**:

* A pie chart can be used to illustrate the gender disparities among marginal workers.
* The chart can have two segments, one for males and one for females, with each segment's size proportional to the respective gender's percentage among marginal workers.

**Bar Chart for Occupational Categories**:

* For visualizing the distribution of marginal workers across occupational categories, create a horizontal bar chart.
* The y-axis represents the different occupational categories (e.g., agricultural laborers, construction workers), and the x-axis represents the number or percentage of workers in each category.

**Heat Map for Geographic Distribution**:

* To visualize the geographic distribution of marginal workers, consider using a heat map.
* You can create a map where regions with a higher concentration of workers are shown in darker colors, while regions with fewer workers are in lighter colors.
* This provides a visual representation of where marginal workers are more densely located.

1. [Link for code](https://colab.research.google.com/drive/1CP7MRMgrz8YOoL-24jlDuCRRXtGovnV4?usp=sharing)
2. [Link for dataset](https://tn.data.gov.in/catalog/marginal-workers-classified-age-industrial-category-and-sex-census-2011-india-and-states)