

All Saints Sr. Sec. School

Ajmer

A Project Report

On

TOPIC

Analysis Of State Wise Adolescent And Youth Population of
India

For

AISSE EXAMINATION (2025-26)

[AS A PART OF THE INFORMATICS PRACTICES COURSE (065)]

SUBMITTED BY:

Darshil Bulchandani

Under Guidance of :

MRS. NEHA SONI

PGT (I.P)

CERTIFICATE

This is to certify that **Mr./Ms. Darshil Bulchandani** of **Class XII Commerce/Humanities/Science** of **All Saints Sr. Sec. School, Ajmer**, has completed his/her Informatics Practices Project File under my guidance. He/She has taken proper care and shown utmost sincerity in completing this Project.

I certify that this project is up to my expectations and as per the guidelines issued by CBSE.

(Signature)

Mrs. Neha Soni

(P.G.T. IP)

(Signature)

Mrs. Jaya Kumar

(Principal)

(Signature)

External Examiner

ACKNOWLEDGMENT

I hereby take this opportunity to thank all the people who helped me in completion of the project without whose support this would have been an impossible task for me. All the teachers and friends were always ready to help me in solving difficult problems.

I would like to thank our Principal Mrs. Jaya Kumar for encouraging me from time to time and for providing me all the facilities and resources needed to carry out my project.

I would like to convey my heartfelt thanks to Mrs. Neha Soni, my IP teacher who always gave me valuable suggestion and guidance for completion of the project. She helped me to understand and remember important detail of the project. My project has been a success only because of her guidance

Darshil Bulchandani

Class XII Commerce

INDEX

| S.No. | CONTENTS |
|-------|------------------------------------|
| | PYTHON INTRODUCTION AND FEATURES |
| | CSV AND MATPLOTLIB INTRODUCTION |
| | OBJECTIVES OF THE PROJECT |
| | PROPOSED SYSTEM |
| | HARDWARE AND SOFTWARE REQUIREMENTS |
| | SOURCE CODE |
| | OUTPUT |
| | BIBILOGRAPHY |

PYTHON – INTRODUCTION & FEATURES

Introduction:

Python is a high-level, interpreted programming language known for its simplicity, versatility, and readability. It was created by Guido van Rossum and released in 1991. Python supports multiple programming paradigms, including object-oriented, functional, and procedural programming.

Python provides a strong foundation for students and professionals to analyze and visualize data efficiently. Its ability to integrate with other tools and handle large datasets makes it a preferred choice for data analytics projects.

Features of Python:

Easy to Learn and Readable: Python's syntax is similar to English, which makes it beginner-friendly.

Interpreted Language: Python executes code line by line, making debugging simple.

Extensive Library Support: Libraries like pandas, matplotlib, numpy, etc.

Cross-Platform Compatibility: Python works on multiple operating systems like Windows, macOS, and Linux.

Open Source: Python is free to use, modify, and distribute.

Object-Oriented and Functional Programming: It supports both paradigms, increasing flexibility.

Dynamic Typing: There's no need to declare variable types explicitly.

Strong Community Support: Python has one of the largest online developer communities.

In this project, Python serves as the core language for reading, processing, analyzing, and visualizing the dataset of India's adolescent and youth population across various states and regions.

CSV & MATPLOTLIB – INTRODUCTION

Introduction to CSV (Comma Separated Values):

A CSV file is one of the most widely used formats for storing tabular data. It contains data separated by commas, where each line represents a new record and each value is separated by a comma. CSV files are simple, lightweight, and easily readable by both humans and computers.

In Python, the pandas library provides built-in methods like `read_csv()` and `to_csv()` to read and write CSV files effortlessly.

In this project, a CSV file named “PCA_AY_2011_Revised.csv” was used. It contains government data released under the National Data Sharing and Accessibility Policy (NDSAP), detailing the adolescent and youth population of India, categorized by age group, gender, area type (rural/urban), and state.

The dataset was read using the pandas function:

```
pd.read_csv("PCA_AY_2011_Revised.csv")
```

and then filtered and analyzed based on selected attributes

Introduction to Matplotlib:

Matplotlib is one of Python’s most popular data visualization libraries. It helps represent complex numerical data through different types of charts and graphs such as line graphs, bar charts, pie charts, and more. By visualizing data, one can understand trends, patterns, and comparisons more clearly.

In this project, Matplotlib was used to create bar charts and line graphs for visualizing:

Population distribution by age group, Rural vs. urban sex ratio, Literacy rate and illiteracy trends, Scheduled caste and tribe population by age

Through the use of CSV files for data handling and Matplotlib for visualization, the project demonstrates how Python can be used to perform meaningful population analysis and present it in a graphical form that’s easy to interpret.

OBJECTIVE OF THE PROJECT

The main objective of this project, “Analysis of State-Wise Adolescent and Youth Population of India,” is to analyze and understand the demographic distribution of adolescents and youth across various states and union territories of India using Python programming. India has one of the largest youth populations in the world. The adolescent and youth age group (10–24 years) plays a crucial role in shaping the country’s social and economic development. Analyzing this data helps in identifying patterns related to population distribution, literacy levels, gender ratio, and social categories such as Scheduled Castes (SC) and Scheduled Tribes (ST).

This project aims to use data analysis and visualization techniques to extract meaningful insights from a real government dataset provided under the National Data Sharing and Accessibility Policy (NDSAP).

Specific objectives include:

To import and organize real population data using Python and the Pandas library.

To perform data manipulation operations such as filtering, sorting, and extracting specific information.

To calculate and compare important demographic indicators such as total population, gender ratio, and literacy rate.

To visualize the analyzed data through charts and graphs using Matplotlib for better understanding.

To identify key trends, such as which state has the highest number of Scheduled Tribes or which region shows literacy challenges among youth.

To demonstrate how Python can be effectively used for data analysis and representation of government datasets.

Through this project, an effort has been made to bridge the gap between raw data and meaningful conclusions, while also strengthening programming and analytical skills using Python.

PROPOSED SYSTEM (WHY WE HAVE CHOSEN THIS PROJECT)

Proposed System:

The proposed system is a Python-based data analysis program that reads, processes, and visualizes the state-wise adolescent and youth population data of India. The system uses Python libraries such as pandas for handling data and matplotlib for creating visual representations like line charts and bar graphs.

The system allows the user to:

Read the dataset from a CSV file.

Perform various types of data manipulations such as adding or removing records.

Conduct analytical operations like finding top and bottom records, total population counts, and literacy comparisons.

Generate visual graphs for better understanding of trends across states and age groups.

Reason for Choosing This Project:

This project was chosen because it connects real-world data with practical data analysis using Python. Population studies are essential for planning in sectors such as education, employment, and public health. Understanding youth and adolescent demographics can help policymakers and researchers identify key challenges and opportunities in India's development.

By choosing this topic, we aimed to:

Work with authentic government data released under the National Data Sharing and Accessibility Policy (NDSAP).

Learn how to handle large datasets efficiently using Python and CSV files.

Gain experience in data visualization, turning numbers into meaningful insights.

HARDWARE AND SOFTWARE REQUIREMENTS

Hardware Requirements:

To execute and test this Python-based project, a basic computer configuration is sufficient. The project does not require high-end hardware as it mainly involves reading CSV files, performing data analysis, and generating graphs.

The minimum hardware specifications are as follows:

Component Minimum Requirement

Processor Intel Core i3 or equivalent

RAM 4 GB (8 GB recommended for smooth performance)

Hard Disk 500 MB of free storage space

Display 1024 × 768 resolution or higher

Keyboard & Mouse Standard input devices

Internet Connection Optional (only required for downloading libraries or dataset)

Software Requirements:

The following software components are used to build and run this project:

Operating System Windows 10 or later / Linux / macOS

Python Version Python 3.8 or above

IDE / Environment Jupyter Notebook (Anaconda Distribution)

Libraries Used pandas, numpy, matplotlib, sys

Dataset Format CSV (Comma Separated Values)

Source of Data National Data Sharing and Accessibility Policy (NDSAP) portal – Government of India

Untitled.ipynb PCA_AY_2011_Revised.csv

Delimiter: .

| | Area Name | Total/ Rural/ Urban | Adolescent and youth categories | Total Population - Persons | Total Population - Males | Total Population - Females | Scheduled Caste - Persons | Scheduled Caste - Males | Scheduled Caste - Females | Scheduled Tribe - |
|----|-------------------------|---------------------|---------------------------------|----------------------------|--------------------------|----------------------------|---------------------------|-------------------------|---------------------------|-------------------|
| 19 | JAMMU & KASHMIR (01) | Total | All Ages | 12541302 | 6640662 | 5900640 | 924991 | 486232 | 438759 | 1 |
| 20 | JAMMU & KASHMIR (01) | Total | 10-14 | 1413853 | 744855 | 668998 | 101117 | 53702 | 47415 | |
| 21 | JAMMU & KASHMIR (01) | Total | 15-19 | 1237462 | 640362 | 597100 | 97563 | 51780 | 45783 | |
| 22 | JAMMU & KASHMIR (01) | Total | 20-24 | 1160913 | 603578 | 557335 | 92420 | 47508 | 44912 | |
| 23 | JAMMU & KASHMIR (01) | Total | Adolescent (10-19) | 2651315 | 1385217 | 1266098 | 198680 | 105482 | 93198 | |
| 24 | JAMMU & KASHMIR (01) | Total | Youth (15-24) | 2398375 | 1243940 | 1154435 | 189983 | 99288 | 90695 | |
| 25 | JAMMU & KASHMIR (01) | Rural | All Ages | 9108060 | 4774477 | 4333583 | 751026 | 392981 | 358045 | 1 |
| 26 | JAMMU & KASHMIR (01) | Rural | 10-14 | 1097236 | 575511 | 521725 | 84272 | 44630 | 39642 | |
| 27 | JAMMU & KASHMIR (01) | Rural | 15-19 | 926108 | 477098 | 449010 | 79402 | 42053 | 37349 | |
| 28 | JAMMU & KASHMIR (01) | Rural | 20-24 | 827361 | 423292 | 404069 | 73987 | 37897 | 36090 | |
| 29 | JAMMU & KASHMIR (01) | Rural | Adolescent (10-19) | 2023344 | 1052609 | 970735 | 163674 | 86683 | 76991 | |
| 30 | JAMMU & KASHMIR (01) | Rural | Youth (15-24) | 1753469 | 900390 | 853079 | 153389 | 79950 | 73439 | |
| 31 | JAMMU & KASHMIR (01) | Urban | All Ages | 3432342 | 1866185 | 1567057 | 1739865 | 93251 | 80714 | |
| 32 | JAMMU & KASHMIR (01) | Urban | 10-14 | 316617 | 169344 | 147273 | 18845 | 9072 | 7773 | |
| 33 | JAMMU & KASHMIR (01) | Urban | 15-19 | 311354 | 163264 | 148090 | 18161 | 9727 | 8434 | |
| 34 | JAMMU & KASHMIR (01) | Urban | 20-24 | 333552 | 180286 | 153266 | 18433 | 9611 | 8822 | |
| 35 | JAMMU & KASHMIR (01) | Urban | Adolescent (10-19) | 627971 | 332608 | 295363 | 35006 | 18799 | 16207 | |
| 36 | JAMMU & KASHMIR (01) | Urban | Youth (15-24) | 644906 | 343550 | 301356 | 36594 | 19338 | 17256 | |
| 37 | District - Kupwara (01) | Total | All Ages | 870354 | 474190 | 395164 | 1048 | 1046 | 2 | |
| 38 | District - Kupwara (01) | Total | 10-14 | 118673 | 62428 | 56245 | 4 | 3 | 1 | |
| 39 | District - Kupwara (01) | Total | 15-19 | 81849 | 41873 | 39976 | 2 | 2 | 0 | |
| 40 | District - Kupwara (01) | Total | 20-24 | 70692 | 38505 | 32187 | 192 | 191 | 1 | |
| 41 | District - Kupwara (01) | Total | Adolescent (10-19) | 200522 | 104301 | 96221 | 6 | 5 | 1 | |
| 42 | District - Kupwara (01) | Total | Youth (15-24) | 152541 | 80378 | 72163 | 194 | 193 | 1 | |
| 43 | District - Kupwara (01) | Rural | All Ages | 765625 | 412038 | 353587 | 872 | 870 | 2 | |
| 44 | District - Kupwara (01) | Rural | 10-14 | 103802 | 54249 | 49553 | 4 | 3 | 1 | |
| 45 | District - Kupwara (01) | Rural | 15-19 | 72665 | 36874 | 35791 | 1 | 1 | 0 | |
| 46 | District - Kupwara (01) | Rural | 20-24 | 62591 | 33731 | 28880 | 173 | 172 | 1 | |
| 47 | District - Kupwara (01) | Rural | Adolescent (10-19) | 176467 | 91123 | 85344 | 5 | 4 | 1 | |
| 48 | District - Kupwara (01) | Rural | Youth (15-24) | 135256 | 70605 | 64651 | 174 | 173 | 1 | |
| 49 | District - Kupwara (01) | Urban | All Ages | 104729 | 62152 | 42577 | 176 | 176 | 0 | |
| 50 | District - Kupwara (01) | Urban | 10-14 | 14871 | 8179 | 6692 | 0 | 0 | 0 | |
| 51 | District - Kupwara (01) | Urban | 15-19 | 9184 | 4999 | 4185 | 1 | 1 | 0 | |
| 52 | District - Kupwara (01) | Urban | 20-24 | 8101 | 4774 | 3327 | 19 | 19 | 0 | |
| 53 | District - Kupwara (01) | Urban | Adolescent (10-19) | 24055 | 13178 | 10877 | 1 | 1 | 0 | |
| 54 | District - Kupwara (01) | Urban | Youth (15-24) | 17285 | 9773 | 7512 | 20 | 20 | 0 | |

DATASET OVERVIEW – CAPTURED FROM EXCEL WORKBOOK

SOURCECODE

```
import pandas as pd  
import numpy as np  
import matplotlib.pyplot as plt  
import sys
```

```
main= """\n \n Main Menu  
1) To write CSV to DataFrame  
2) To manipulate data  
3) To analyse Data  
4) To visualise Data  
5) Exit"""
```

```
analyse= """\n \n -----Analysis Data Menu -----  
1) All Data  
2) Top Records  
3) Bottom Records  
4) Total population, total male population and total female population  
aged 10 to 24 in India  
5) Compare the sex ratio of urban areas and rural areas.  
6) Which state is home to the maximum no. of ST in India? Which state  
has the minimum no. of ST in India?  
7) Which State or Union Territory in India has the maximum number  
of illiterates in the youth ages?  
8) Return to Main Menu  
"""
```

```
visualize= """\n \n -----Visualisation Data Menu-----  
1) Line Chart(Agricultural Labourers age wise in a state)  
2) Line Chart(ST Males age wise in a state)  
3) Line Chart(Rural illiterates age wise in a state)  
4) Bar Chart (Total population, total male population and total  
female population aged 10 to 24 in India)  
5) Bar Chart (Sex ratio of urban areas and rural areas.)  
6) Bar Chart (No. of SCs age wise in a state)  
7) Bar Chart (No. of female illiterates age wise in a state)  
8) Return to Main Menu
```

"""

manipulate=""\n \n -----Manipulation Data Menu-----

- 1) To append a record
- 2) To remove a record"""

#Data Source OGD PCA_AY_2011_Revised

#Selected columns are chosen

```
df=pd.read_csv("PCA_AY_2011_Revised.csv",
    usecols=["State Code","District Code", "Area Name","Total/ Rural/ Urban",
    "Adolescent and youth categories","Total Population - Persons",
    "Total Population - Males", "Total Population - Females",
    "Scheduled Caste - Persons","Scheduled Tribe - Persons",
    "Scheduled Tribe - Males", "Illiterates - Total - Persons",
    "Illiterates - Total - Males","Illiterates - Total - Females",
    "Main Worker - Agricultural labourers - Persons"])
```

def main_menu():

print(main)

ch=int(input("Enter choice"))

if ch==1:

print(df)

main_menu()

elif ch==2:

manipulate_menu()

elif ch==3:

analyse_data()

elif ch==4:

visualise_data()

elif ch==5:

sys.exit()

def analyse_data():

print(analyse)

ch=int(input("Enter choice"))

if ch==1:

print(df)

elif ch==2:

x=int(input("How many records from top do you want?"))

```

    print(df.head(x))
elif ch==3:
    x=int(input("How many records from bottom do you want?"))
    print(df.tail(x))
elif ch==4:
    df1=df.loc[(df['Area Name'] == 'INDIA'),
    'Area Name':'Total Population - Females']
    df1.columns= ['Area', 'Class', 'Category', 'TotalPop', 'MalePop',
    'FemalePop']
    d=df1.loc[1:3]
    d=d.drop(["Area","Class"],axis=1)
    print(d)
elif ch==5:
    df1=df.loc[(df["Area Name"] == "INDIA"),
    ["Total/ Rural/ Urban","Adolescent and youth categories",
    "Total Population - Males","Total Population - Females"]]
    df1.columns=["Type","Category", "MalePop","FemalePop"]
    d=df1.loc[[6,12]]
    d["Sex Ratio"]=d["FemalePop"]/d["MalePop"] * 1000
    d=d.drop(["Category","MalePop","FemalePop"],axis=1)
    print(d)
elif ch==6:
    df1=df.loc[(df['Area Name'].str.contains('State'))&
    (df["Adolescent and youth categories"]== "All Ages")&
    (df["Total/ Rural/ Urban"]=="Total")
    ,["Area Name","Adolescent and youth categories",
    "Scheduled Tribe - Persons"]]
    print("State with maximum number of ST is:")
    mx=int(df1.max(numeric_only=True).values)
    dfmax=df1.loc[(df1["Scheduled Tribe - Persons"]==mx),
    ["Area Name","Scheduled Tribe - Persons"]]
    print(dfmax)
    print("States with minimum number of ST are:")
    mn=int(df1.min(numeric_only=True).values)
    dfmin=df1.loc[(df1["Scheduled Tribe - Persons"]==mn),
    ["Area Name","Scheduled Tribe - Persons"]]
    print(dfmin)
elif ch==7:
    df1=df.loc[(df['Area Name'].str.contains('State'))&
    (df["Adolescent and youth categories"]== "Youth (15-24)")&

```

```

(df["Total/ Rural/ Urban"]=="Total"),
["Area Name","Illiterates - Total - Persons"]]
print("""State or UT which has maximum number of illiterates
in the youth ages is:"""")
mx=int(df1.max(numeric_only=True).values)
dfmax=df1.loc[(df1["Illiterates - Total - Persons"]==mx),
["Area Name","Illiterates - Total - Persons"]]
print(dfmax)
elif ch==8:
    main_menu()
main_menu()

def visualise_data():
    print(visualize)
    ch=int(input("Enter choice"))
    if ch==1:
        df1=df.loc[(df['Area Name'].str.contains('State'))
&(df["Total/ Rural/ Urban"]=="Total"),
["Area Name","Adolescent and youth categories",
"Main Worker - Agricultural labourers - Persons"]]
        b=input("Which state graph do you want?")
        df2=df1.loc[(df1["Area Name"].str.contains(b.upper())),["Area Name",
        "Adolescent and youth categories","Main Worker - Agricultural labourers -
Persons"]]
        print(df2)
        plt.plot(df2["Adolescent and youth categories"],df2["Main Worker -
Agricultural labourers - Persons"]
,color="Gold")
        plt.xlabel("Category(age)")
        plt.ylabel("No. of Agricultural Labourers")
        plt.title("No of agricultural workers age wise")
        plt.grid(True)
        plt.show()
    elif ch==2:
        df1=df.loc[(df['Area Name'].str.contains('State'))
&(df["Total/ Rural/ Urban"]=="Total"),
["Area Name","Adolescent and youth categories",
"Scheduled Tribe - Males"]]
        b=input("Which state graph do you want?")
        df2=df1.loc[(df1["Area Name"].str.contains(b.upper())),["Area Name",

```

```

    "Adolescent and youth categories","Scheduled Tribe - Males"])
print(df2)
plt.plot(df2["Adolescent and youth categories"],df2["Scheduled Tribe -
Males"])
,color="orange")
plt.xlabel("Category(age)")
plt.ylabel("No. of ST Males")
plt.title("No of ST Males age wise")
plt.grid(True)
plt.show()
elif ch==3:
    df1=df.loc[(df['Area Name'].str.contains('State'))
&(df["Total/ Rural/ Urban"]=="Rural"),
    ["Area Name","Adolescent and youth categories",
    "Illiterates - Total - Persons"]]
    b=input("Which state graph do you want?")
    df2=df1.loc[(df1["Area Name"].str.contains(b.upper()),["Area Name",
    "Adolescent and youth categories","Illiterates - Total - Persons"])]
    print(df2)
    plt.plot(df2["Adolescent and youth categories"],df2["Illiterates - Total -
Persons"])
    plt.xlabel("Category(age)")
    plt.ylabel("No. of rural illiterates")
    plt.title("No of rural illiterates age wise")
    plt.grid(True)
    plt.show()
elif ch==4:
    df1=df.loc[(df['Area Name'] == 'INDIA'),
    'Area Name':'Total Population - Females']
    df1.columns= ['Area', 'Class', 'Category', 'TotalPop', 'MalePop',
    'FemalePop']
    d=df1.loc[1:3]
    d=d.drop(["Area","Class"],axis=1)
    d.index=["10-14","15-19","20-24"]
    d.plot(kind="bar",title="Total Population Age wise in youth")
    plt.xlabel("Age category")
    plt.ylabel("Population")
    plt.show()
elif ch==5:
    df1=df.loc[(df["Area Name"] == "INDIA"),

```

```

["Total/ Rural/ Urban","Adolescent and youth categories" ,
 "Total Population - Males","Total Population - Females"]]
df1.columns=["Type","Category", "MalePop","FemalePop"]
d=df1.loc[[6,12]]
d["Sex Ratio"]=d["FemalePop"]/d["MalePop"] * 1000
d=d.drop(["Category","MalePop","FemalePop"],axis=1)
d.index=["Rural","Urban"]
d.plot(kind="bar")
plt.xlabel("Area type")
plt.ylabel("Sex Ratio")
plt.show()

elif ch==6:
    df1=df.loc[(df['Area Name'].str.contains('State'))
    &(df["Total/ Rural/ Urban"]=="Total"),
    ["Area Name","Adolescent and youth categories",
    "Scheduled Caste - Persons"]]
    b=input("Which state graph do you want?")
    df2=df1.loc[(df1["Area Name"].str.contains(b.upper()),["Area Name",
    "Adolescent and youth categories","Scheduled Caste - Persons"])]
    df2=df2.drop(["Area Name"],axis=1)
    df2.plot(kind="bar",x="Adolescent and youth
    categories",color="red",edgecolor="gold")
    plt.xlabel("Category(age)")
    plt.ylabel("No. of SCs")
    plt.title("No of SCs age wise")
    plt.show()

elif ch==7:
    df1=df.loc[(df['Area Name'].str.contains('State'))
    &(df["Total/ Rural/ Urban"]=="Total"),
    ["Area Name","Adolescent and youth categories",
    "Illiterates - Total - Females"]]
    b=input("Which state graph do you want?")
    df2=df1.loc[(df1["Area Name"].str.contains(b.upper()),["Area Name",
    "Adolescent and youth categories","Illiterates - Total - Females"])]
    df2=df2.drop(["Area Name"],axis=1)
    print(df2)
    df2.plot(kind="bar",x="Adolescent and youth
    categories",color="green",edgecolor="silver",linewidth=3)
    plt.xlabel("Category(age)")
    plt.ylabel("No. of female illiterates")

```

```

plt.title("No of female illiterates age wise")
plt.show()
elif ch==8:
    main_menu()

main_menu()

def manipulate_menu():
    df=pd.read_csv(r"PCA_AY_2011_Revised.csv",
    usecols=["State Code","District Code", "Area Name","Total/ Rural/ Urban",
    "Adolescent and youth categories","Total Population - Persons",
    "Total Population - Males", "Total Population - Females",
    "Scheduled Caste - Persons","Scheduled Tribe - Persons",
    "Scheduled Tribe - Males", "Illiiterates - Total - Persons",
    "Illiiterates - Total - Males","Illiiterates - Total - Females",
    "Main Worker - Agricultural labourers - Persons"])
    print(manipulate)
    ch=int(input("Enter choice"))
    if ch==1:
        a1=int(input('State Code'))
        a2=int(input('District Code'))
        a3=input('Area Name')
        a4=input('Total/ Rural/ Urban')
        a5=input('Adolescent and youth categories')
        a6=int(input('Total Population - Persons'))
        a7=int(input('Total Population - Males'))
        a8=int(input('Total Population - Females'))
        a9=int(input('Scheduled Caste - Persons'))
        a10=int(input('Scheduled Tribe - Persons'))
        a11=int(input('Scheduled Tribe - Males'))
        a12=int(input('Illiiterates - Total - Persons'))
        a13=int(input('Illiiterates - Total - Males'))
        a14=int(input('Illiiterates - Total - Females'))
        a15=int(input('Main Worker - Agricultural labourers - Persons'))
        l1=[a1,a2,a3,a4,
            a5,a6,a7,a8,
            a9,a10,a11,a12,
            a13,a14,a15]
        df.loc[-1]=l1
        print("DataFrame is updated")

```

```
o=input("Do you want to update in CSV? (Yes/No)")  
if o.upper()=="YES":  
    df.to_csv(r"C:\Users\Student\Desktop\IP PROJECT\  
PCA_AY_2011_Revised.csv"  
        ,index=False)  
elif o.upper()=="NO":  
    pass  
elif ch==2:  
    a=int(input("Enter index number of the record to be removed"))  
    df=df.drop([a],axis=0)  
    print("DataFrame is updated")  
    o=input("Do you want to update in CSV? (Yes/No)")  
    if o.upper()=="YES":  
        df.to_csv(r"C:\Users\Student\Desktop\IP PROJECT\  
PCA_AY_2011_Revised.csv"  
            ,index=False)  
    elif o.upper()=="NO":  
        pass  
  
main_menu()  
  
main_menu()
```

OUTPUTS

The screenshot shows a Jupyter Notebook interface with a code cell containing Python code and its output. The code cell displays a menu and several data frames.

```
Main Menu  
1) To write CSV to DataFrame  
2) To manipulate data  
3) To analyse Data  
4) To visualise Data  
5) Exit  
Enter choice 1:  
      State Code District Code          Area Name \\\n0            0           0             INDIA  
1            0           0             INDIA  
2            0           0             INDIA  
3            0           0             INDIA  
4            0           0             INDIA  
...  
12163       35          648  District - South Andaman (85)  
12164       35          649  District - South Andaman (85)  
12165       35          649  District - South Andaman (85)  
12166       35          649  District - South Andaman (85)  
12167       35          649  District - South Andaman (85)  
  
Total/ Rural/ Urban Adolescent and youth categories \\  
0   Total          All Ages    ...  
1   Total          10-14  
2   Total          15-19  
3   Total          20-24  
4   Total          Adolescent (10-19)  
...  
12163   Urban          10-14  
12164   Urban          15-19  
12165   Urban          20-24  
12166   Urban          Adolescent (10-19)  
12167   Urban          Youth (15-24)  
  
Total Population - Persons  Total Population - Males \\  
0   1210854977    623279258  
1   132709212     69418835  
2   120526449     63982396  
3   111424222     57584693  
4   253235681     133401231  
...  
12163   126000        6279  
12164   12415         6494  
12165   14272         7727  
12166   24464         12773  
12167   26688         14221  
  
Mode: Command 1 @ Python (Pydide) | Busy  Ln 51, Col 20 Untitled.ipynb 2 □
```

OUTPUT 1

The screenshot shows a Jupyter Notebook interface with a code cell containing Python code and its output. The code cell displays a menu and several data frames.

```
3          18679492    6467026  
4          25283316    11024342  
...  
12163       208         116  
12164       227         122  
12165       518         243  
12166       435         238  
12167       737         365  
  
Illiterates - Total - Females \\  
0          258709529  
1          6356613  
2          792321  
3          12212466  
4          14178974  
...  
12163       92  
12164       185  
12165       267  
12166       197  
12167       372  
  
Main Worker - Agricultural labourers - Persons  
0          8058700  
1          1108999  
2          6170641  
3          10521664  
4          7279639  
...  
12163       0  
12164       3  
12165       10  
12166       3  
12167       13  
  
[12168 rows x 15 columns]  
  
Main Menu  
1) To write CSV to DataFrame  
2) To manipulate data  
3) To analyse Data  
4) To visualise Data  
5) Exit  
Enter choice [ ]:  
Mode: Command 1 @ Python (Pydide) | Busy  Ln 51, Col 20 Untitled.ipynb 2 □
```

OUTPUT 2

```

Untitled.ipynb      X + Main Menu
+ X □ ▶ ■ C ▶▶ Code
Main Menu
1) To write CSV to DataFrame
2) To manipulate data
3) To analyse Data
4) To visualise Data
5) Exit
Enter choice 3

-----Analysis Data Menu -----
1) All Data
2) Top Records
3) Bottom Records
4) Total population, total male population and total female population
   aged 10 to 24 in India
5) Compare the sex ratio of urban areas and rural areas.
6) Which state is home to the maximum no. of ST in India? Which state
   has the minimum no. of ST in India?
7) Which State or Union Territory in India has the maximum number
   of illiterates in the youth ages?
8) Return to Main Menu

Enter choice 2
How many records from top do you want? 5
    State Code District Code Area Name Total/ Rural/ Urban \
0          0           0       INDIA      Total
1          0           0       INDIA      Total
2          0           0       INDIA      Total
3          0           0       INDIA      Total
4          0           0       INDIA      Total

Adolescent and youth categories   Total Population - Persons \
0           All Ages           1210854977
1           10-14                132709212
2           15-19                120526449
3           20-24                111424222
4           Adolescent (10-19)     253235661

Total Population - Males   Total Population - Females \
0           623270258           587584719
1           69418835            63290377
2           63982396            56544053
3           57584693            53839529
4           133401231           119834430

Scheduled Caste - Persons   Scheduled Tribe - Persons \
0           201378372            104545716

```

OUTPUT 3

```

Untitled.ipynb      Main Menu
+ X □ ▶ ■ C ▶▶ Code
Main Menu
1) To write CSV to DataFrame
2) To manipulate data
3) To analyse Data
4) To visualise Data
5) Exit
Enter choice 3

-----Analysis Data Menu -----
1) All Data
2) Top Records
3) Bottom Records
4) Total population, total male population and total female population
   aged 10 to 24 in India
5) Compare the sex ratio of urban areas and rural areas.
6) Which state is home to the maximum no. of ST in India? Which state
   has the minimum no. of ST in India?
7) Which State or Union Territory in India has the maximum number
   of illiterates in the youth ages?
8) Return to Main Menu

Enter choice 4
    Category  TotalPop  MalePop  FemalePop
1        10-14    132709212  69418835  63290377
2        15-19    120526449  63982396  56544053
3        20-24    111424222  57584693  53839529

Main Menu
1) To write CSV to DataFrame
2) To manipulate data
3) To analyse Data
4) To visualise Data
5) Exit
Enter choice 3

-----Analysis Data Menu -----
1) All Data
2) Top Records
3) Bottom Records
4) Total population, total male population and total female population
   aged 10 to 24 in India
5) Compare the sex ratio of urban areas and rural areas.
6) Which state is home to the maximum no. of ST in India? Which state
   has the minimum no. of ST in India?

```

OUTPUT 4

```

Untitled.ipynb + Code
2) To manipulate data
3) To analyse Data
4) To visualise Data
5) Exit
Enter choice 3

-----Analysis Data Menu -----
1) All Data
2) Top Records
3) Bottom Records
4) Total population, total male population and total female population
   aged 10 to 24 in India
5) Compare the sex ratio of urban areas and rural areas.
6) Which state is home to the maximum no. of ST in India? Which state
   has the minimum no. of ST in India?
7) Which State or Union Territory in India has the maximum number
   of illiterates in the youth ages?
8) Return to Main Menu

Enter choice 5
      Type  Sex Ratio
6   Rural  949.008345
12  Urban  929.038151

Main Menu
1) To write CSV to DataFrame
2) To manipulate data
3) To analyse Data
4) To visualise Data
5) Exit
Enter choice 4

-----Visualisation Data Menu-----
1) Line Chart(Agricultural Labourers age wise in a state)
2) Line Chart(ST Males age wise in a state)
3) Line Chart(Rural illiterates age wise in a state)
4) Bar Chart (Total population, total male population and total
   female population aged 10 to 24 in India)
5) Bar Chart (Sex ratio of urban areas and rural areas.)
6) Bar Chart (No. of SCs age wise in a state)
7) Bar Chart (No. of female illiterates age wise in a state)
8) Return to Main Menu

Enter choice [↑↓ for history. Search history with c-↑/c-↓]

```

OUTPUT 5

```

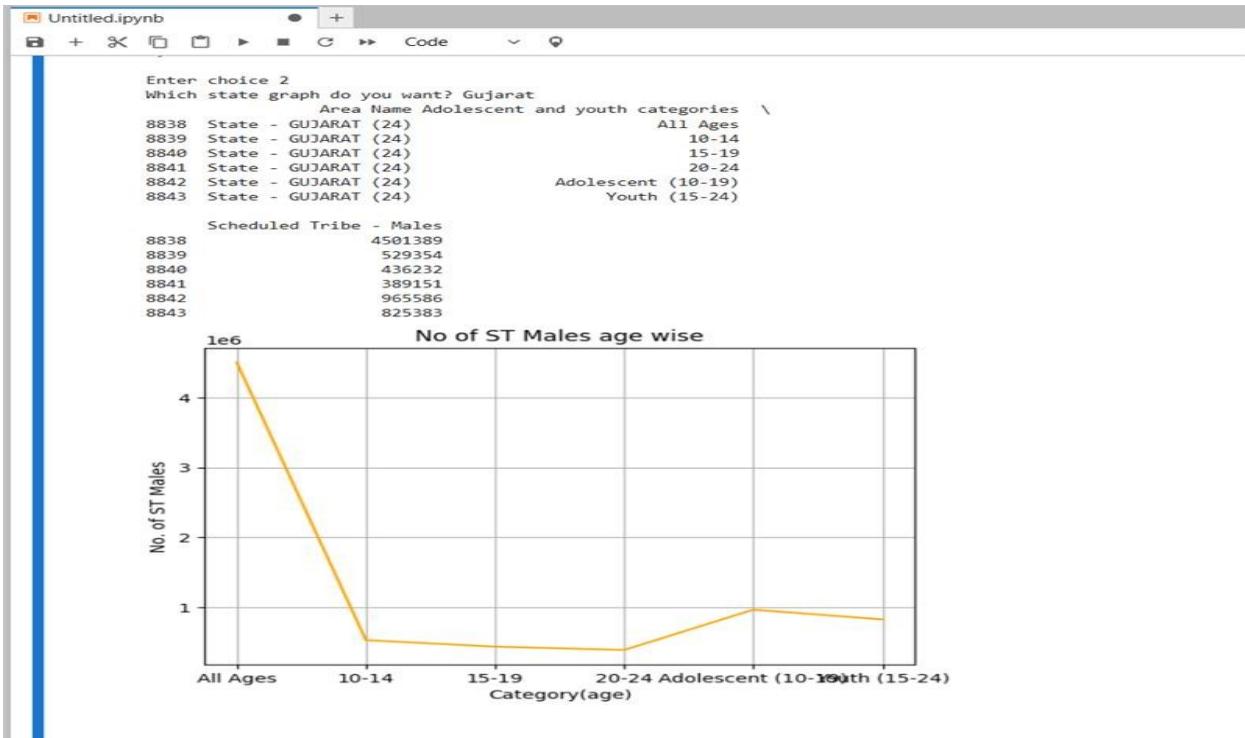
Untitled.ipynb + Code
Enter choice 1
Which state graph do you want? Rajasthan
      Area Name Adolescent and youth categories \
1908  State - RAJASTHAN (08)          All Ages
1909  State - RAJASTHAN (08)          10-14
1910  State - RAJASTHAN (08)          15-19
1911  State - RAJASTHAN (08)          20-24
1912  State - RAJASTHAN (08)          Adolescent (10-19)
1913  State - RAJASTHAN (08)          Youth (15-24)

Main Worker - Agricultural labourers - Persons
1908                2195304
1909                37491
1910                221863
1911                349758
1912                259354
1913                571621

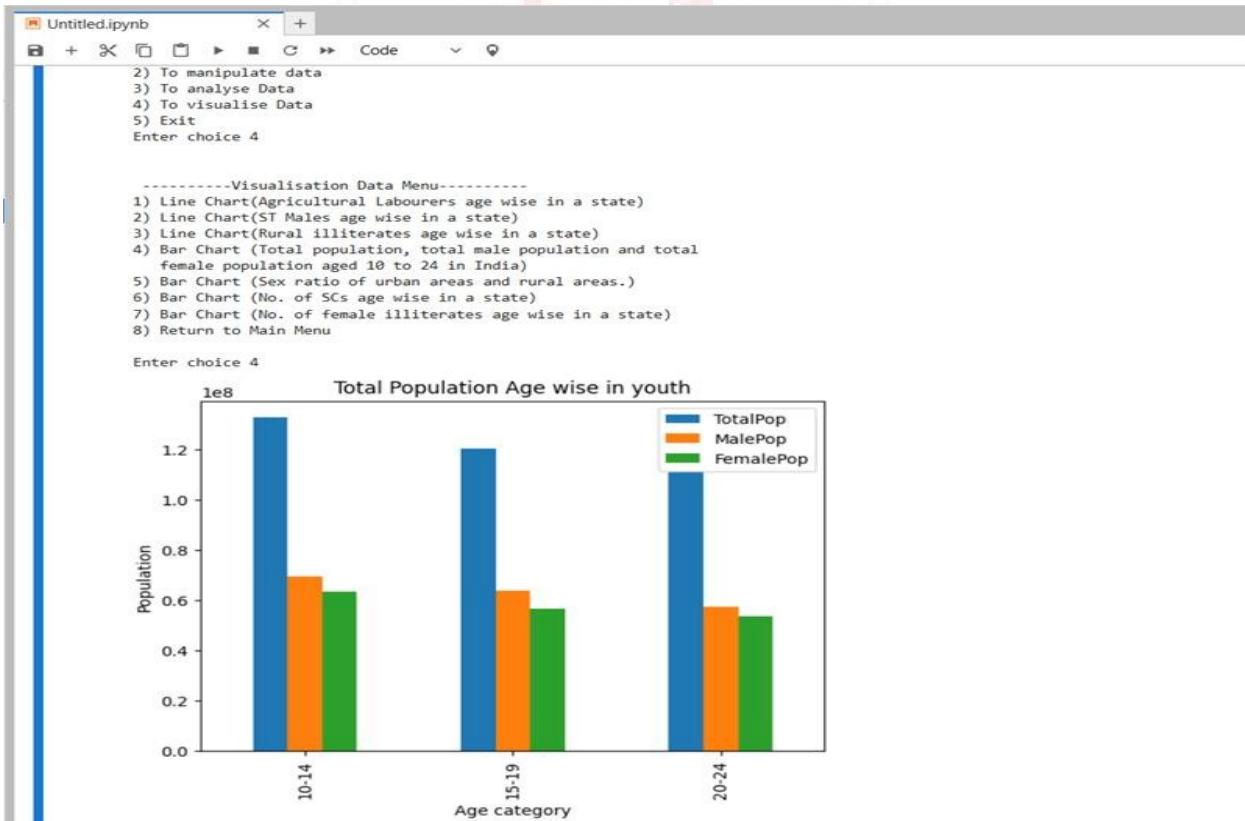
      No of agricultural workers age wise
      1e6
      2.0
      1.5
      1.0
      0.5
      0.0
      All Ages    10-14    15-19    20-24 Adolescent (10-19) Youth (15-24)
      Category(age)

```

OUTPUT 6



OUTPUT 7



OUTPUT 8

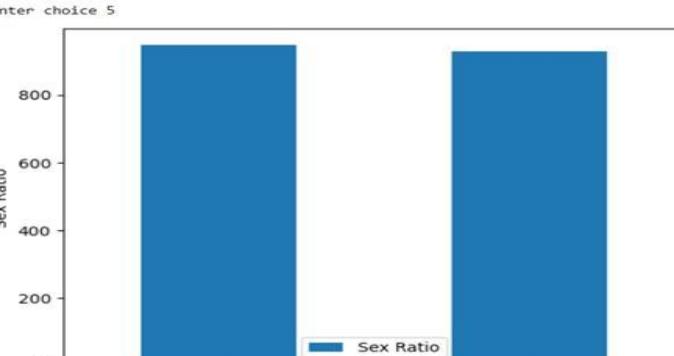
Untitled.ipynb

Code

2) To manipulate data
3) To analyse Data
4) To visualise Data
5) Exit
Enter choice 4

-----Visualisation Data Menu-----
1) Line Chart(Agricultural Labourers age wise in a state)
2) Line Chart(ST Males age wise in a state)
3) Line Chart(Rural illiterates age wise in a state)
4) Bar Chart (Total population, total male population and total female population aged 10 to 24 in India)
5) Bar Chart (Sex ratio of urban areas and rural areas.)
6) Bar Chart (No. of SCs age wise in a state)
7) Bar Chart (No. of female illiterates age wise in a state)
8) Return to Main Menu

Enter choice 5



A bar chart titled "Sex Ratio" comparing the sex ratio between Rural and Urban areas. The Y-axis represents the Sex Ratio, ranging from 0 to 800 with major ticks at 0, 200, 400, 600, and 800. The X-axis is labeled "Area type" with two categories: "Rural" and "Urban". Two blue bars are plotted: the "Rural" bar reaches approximately 880, and the "Urban" bar reaches approximately 850. A legend at the bottom left indicates that the blue color represents the "Sex Ratio".

| Area type | Sex Ratio |
|-----------|-----------|
| Rural | ~880 |
| Urban | ~850 |

OUTPUT 9

```
Untitled.ipynb + Code
2) To manipulate data
3) To analyse Data
4) To visualise Data
5) Exit
Enter choice 4

-----Visualisation Data Menu-----
1) Line Chart(Agricultural Labourers age wise in a state)
2) Line Chart(ST Males age wise in a state)
3) Line Chart(Rural illiterates age wise in a state)
4) Bar Chart (Total population, total male population and total
   female population aged 10 to 24 in India)
5) Bar Chart (Sex ratio of urban areas and rural areas.)
6) Bar Chart (No. of SCs age wise in a state)
7) Bar Chart (No. of female illiterates age wise in a state)
8) Return to Main Menu

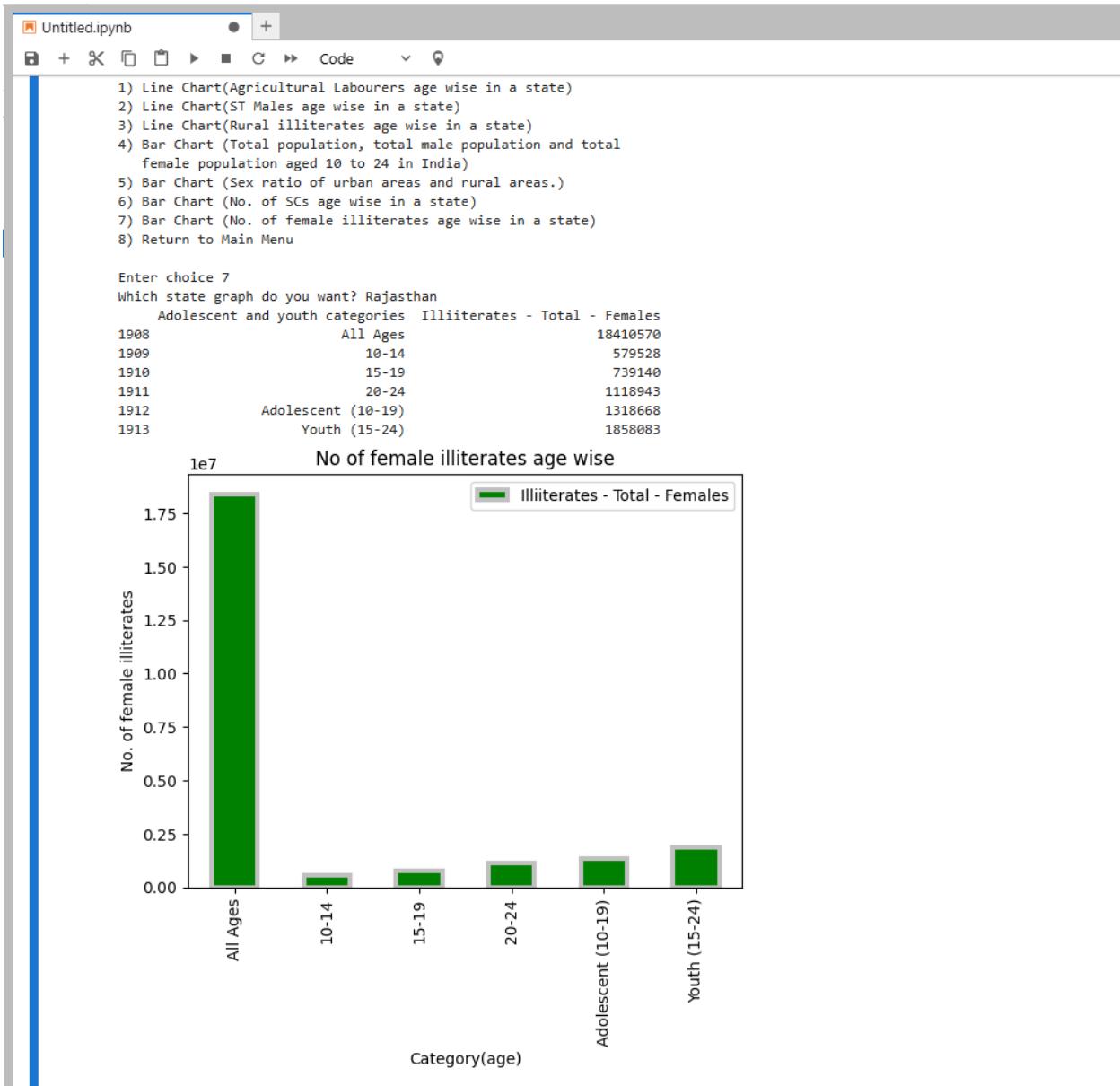
Enter choice 6
Which state graph do you want? Rajasthan
No of SCs age wise



| Category(age)      | No. of SCs |
|--------------------|------------|
| All Ages           | ~1.2e7     |
| 10-14              | ~0.15e7    |
| 15-19              | ~0.15e7    |
| 20-24              | ~0.12e7    |
| Adolescent (10-19) | ~0.3e7     |
| Youth (15-24)      | ~0.25e7    |


```

OUTPUT 10



OUTPUT 11

BIBLIOGRAPHY

The following sources, tools, and references were used during the creation and development of the project “Analysis of State-Wise Adolescent and Youth Population of India”:

Dataset Source: <https://data.gov.in>

(Used for obtaining the CSV dataset “PCA_AY_2011_Revised.csv” containing adolescent and youth population data of India.)

Python Official Documentation

<https://docs.python.org/3/> (Reference for Python syntax, functions, and built-in modules.)

Pandas Library Documentation

<https://pandas.pydata.org/> (Used for understanding CSV reading, DataFrame manipulation, and data analysis techniques.)

Matplotlib Library Documentation

<https://matplotlib.org/stable/contents.html> (Referred for creating line and bar charts to visualize data trends.)

NumPy Library Documentation

<https://numpy.org/doc/> (Used for understanding numerical operations and data processing in Python.)

Jupyter Notebook – Project Environment

<https://jupyter.org/> (Used as the development and execution environment for the Python program.)

NCERT Informatics Practices Textbook – Class XII

(For conceptual understanding of data handling, CSV operations, and Python libraries.)

