“It’s all in the Data”: Telangana State Analysis

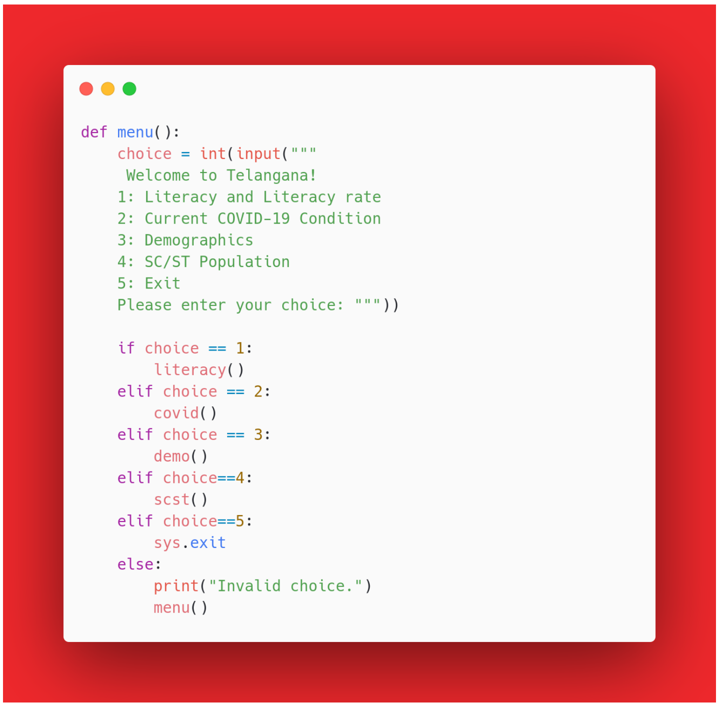
Version 1.0 (2020-2021)

Computer Science (083) Project

Developed By:

Natasha Arya (14637827)
Aayush Ranjan (14637807)
Class XII-E

Delhi Public School, R.K.Puram, New Delhi
www.dpsrkp.net



```
def menu():
    choice = int(input("""
     Welcome to Telangana!
    1: Literacy and Literacy rate
    2: Current COVID-19 Condition
    3: Demographics
    4: SC/ST Population
    5: Exit
    Please enter your choice: """))

    if choice == 1:
        literacy()
    elif choice == 2:
        covid()
    elif choice == 3:
        demo()
    elif choice==4:
        scst()
    elif choice==5:
        sys.exit
    else:
        print("Invalid choice.")
        menu()
```

Index

Certificate

This is to certify that “Telangana State: an Analysis” Computer Science project is developed by **Natasha Arya** and **Aayush Ranjan** of Class XII-E under my supervision in the session 2020-2021.

The work done by them is original.

____
Ms. Hema Jain
Computer Science Teacher

Date: ______________

Acknowledgement

I would like to express my sincere gratitude to my computer teacher Ms. Hema Jain for her invaluable support, teaching, and encouragement, without which this project would not have been possible. Her guidance enabled us to complete the project and her feedback, thoughts and made this project easy and accurate.

I wish to thank my parents for their undivided support and interest who inspired me and encouraged me to go my own way, without which I would be unable to complete my project.

References

1. Class Notes
2. https://stackoverflow.com
3. https://geeksforgeeks.com
4. https://api.covid19india.org/
5. https://data.telangana.gov.in/

Introduction

Telangana is the 29th state of India, and was formed in June 2019. The newly formed state comprises 112,077 square kilometres and constitutes of a population of 35,193,978 as per the latest census conducted in 2011. Telangana is famous for its rich culture and other tourist attractions.

Telangana is one of the pioneers for which data has been made publicly available, as per the Open Data Policy. Through this policy data collection can be catalyzed, thus making data-driven decision making possible and allowing development of important solutions for societal benefits.

Telangana is a new state, which is rapidly developing and culturally diverse; so, we chose to write a program which would help users know more about the state and create awareness. We have used our knowledge of Python, libraries, and CSV files, to create a simple menu-driven program.

We have divided the program into small functions to take care of smooth future upgrades and expansion. Every function (1-5) has two options: Overall and City-Specific. The following is the list of Functions along with their description:

Sno	Function Name	Description
1	**literacy()**	Literacy is critical to economic development as well as individual and community well-being. Through our program, users can acquire information related to literacy rate of a city, and literacy as a whole in Telangana.
2	**covid()**	In light of the current circumstances due to the COVID-19 pandemic, users can know the live status of the recent cases.
3	**covid1()**	This function is called by the covid() function. It displays the data of confirmed cases, deaths, and recoveries based on the user's choice.
4	**covid2()**	This function is also called by the covid() function.The users can check the efficiency based on the number of patients and hospital beds available.
5	**demo()**	Users can acquire information regarding the demographics of Telangana based on gender, as per 2011 census.
6	**scst()**	Information about the Scheduled Castes and Scheduled Tribes can be viewed, to ensure their due representation.
7	**menu()**	This is the main function which provides options to the user to navigate the program.
Other functions for data manipulation:		
8	**tg_daily_api_parser()**	This formats the data from JSON format from the COVID-19 API and selects the data of only Telangana, so that it can be accessed by our program.
9	**tg_current_api_parser()**	It pulls the data of Telangana from the API that provides states and COVID-case details as of the current day.
10	**bar_graph(height,title)**	The function plots a bar graph based on the height and title parameters given to it.

CSV Files with structure used in the program

File Name: Hospitals.csv

S. No.	Column Name	Datatype	Description
1	Hospitals	String	Cities whose hospitals data is available
2	Beds in all Hospitals	Integer	Number of beds available in all hospitals in the city

File Name: Demographics.csv

S. No.	Field/Column Name	Datatype	Description
1	Districts	String	Districts whose data is available
2	Males	Integer	Number of male persons in the district
3	Females	Integer	Number of female persons in the district

File Name: ST_SC Population.csv

S. No.	Field/Column Name	Datatype	Description
1	Districts	String	Districts whose data is available
2	% SC Population	Float	Percentage of SC population in the district
3	% ST Population	Float	Percentage of ST population in the district

File Name: LiteracyRate.csv

S. No.	Field/Column Name	Datatype	Description
1	Districts	String	Districts whose data is available
2	Literacy Rate Males	Float	Literacy rate (%) of male persons in that district
3	Literacy Rate Females	Float	Literacy rate (%) of female persons in that district

External Packages/Modules used in the Program

- **PANDAS:** Pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series. To import this library: `import pandas as pd`

- **MATPLOTLIB:** Matplotlib is a plotting library for the Python programming language, which can create data visualisations. To import this library: `import matplotlib.pyplot as plt`

Salient Features of the Project

- Easy-to-use menu driven options for the user
- Exceptions used for incorrect menu option to avoid inconvenience to users
- Matplotlib library used to better understand data by visualising it through graphs
- Code organised into functions to avoid repetition of code
- Users can check data for particular cities/districts as well the state overall

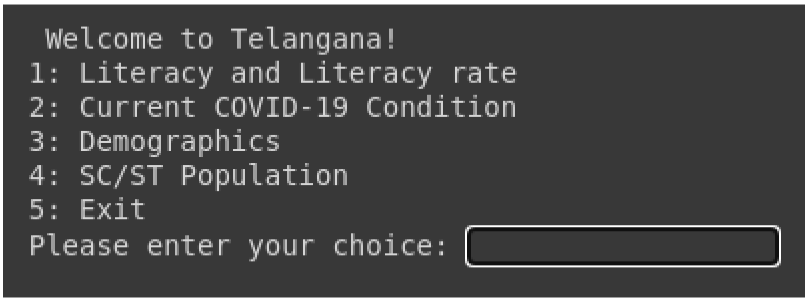
Source Code

```
# Project Title   : “It’s all in the Data” - Telangana State Analysis
# Version         : 1.0 2020-2021
# Developed By    : Natasha Arya & Aayush Ranjan
# Guide           : Ms. Hema Jain
# Last Updated On: 2021-03-20

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

COVID19ORGAPI = 'https://api.covid19india.org/'
COVID19STATESDATAAPI = 'https://api.covid19india.org/v4/data.json'
STATESDAILY = 'https://api.covid19india.org/states_daily.json'

hospitalsDF = pd.read_csv('Hospitals.csv')
demoDF = pd.read_csv('Demographics.csv')
scstDF = pd.read_csv('ST_SC Population.csv')
litdf = pd.read_csv('LiteracyRate.csv')

def tg_daily_api_parser(choice):
   r = requests.get(url = STATESDAILY)
   data = r.json()
   dailydata = data['states_daily']
   dailydata = dailydata[-30:]
   finaldata = {}
   if choice == 1:
       for i in dailydata:
           if i['status'] == 'Confirmed':
               finaldata[i['date']] = int(i['tg'])
       return finaldata
   elif choice == 2:
       for i in dailydata:
           if i['status'] == 'Recovered':
               finaldata[i['date']] = int(i['tg'])
       return finaldata
   elif choice == 3:
       for i in dailydata:
           if i['status'] == 'Deceased':
               finaldata[i['date']] = int(i['tg'])
       return finaldata
   else:
       return None

def tg_current_api_parser():

   r = requests.get(url = COVID19STATESDATAAPI)
   data = r.json()
   return data['TG']

def bar_graph(height, title):
   bars = ('A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J')
   y_pos = np.arange(len(bars))
   plt.bar(y_pos, height, color=(0.2, 0.4, 0.6, 0.6))
   print(height)
   plt.xlabel(title, fontweight='bold', color = 'orange', fontsize='17',
horizontalalignment='center')
   plt.show()

def submenu():
   global choice
   print('Specific city or Overall data?:')
   choice = input().lower()

def literacy():
   litdf.columns = litdf.columns.str.strip().str.lower().str.replace('
','_')
   district_list = litdf.districts.tolist()
   males_list = litdf.literacy_rate_males.tolist()
   females_list = litdf.literacy_rate_females.tolist()
   Gender = ['Male','Female']

   submenu()
   if choice == 'city':
       cityname = input("Enter City name: ").upper()
       citydf =litdf[litdf['districts'] == cityname]
       city_name = citydf.districts.tolist()
       city_males_list = citydf.literacy_rate_males.tolist()
       city_females_list = citydf.literacy_rate_females.tolist()
       city_gender_list = [0,0]
       city_gender_list[0] = city_males_list[0]
       city_gender_list[1] = city_females_list[0]
       fig1, ax1 = plt.subplots()
       ax1.pie(city_gender_list, labels=Gender, autopct='%1.1f%%',
shadow=True, startangle=90)
       ax1.pie(city_gender_list, labels=Gender, autopct='%1.1f%%',
shadow=True, startangle=90)
       ax1.axis('equal')
       plt.show()

   elif choice == 'overall':
       fig = plt.figure(figsize = (12,9))
       plt.bar(district_list,males_list, align='edge', width=0.6)
       plt.bar(district_list,females_list, align='edge', width=0.6)
       plt.xlabel("Districts")
       plt.xticks(rotation=90)
       plt.ylabel("Literacy Rate")
       plt.legend(Gender,loc=2)
       plt.show()

def covid():
   choice = int(input("""
                   View:
                   1. Chart showing recent daily cases
                   2. Efficiency rate of COVID-19
                   Please enter your choice: """))
   if choice == 1:
       covid1()
   elif choice == 2:
       covid2()
   else:
       print('Invalid choice.')
       covid()

def covid1():
   choice = int(input("""
               View:
               1. Confirmed Patients Graph
               2. Recovered Patients Graph
               3. Deceased Patients Graph
               Please enter your choice: """))
   title = ''
   if choice == 1:
       title = 'Confirmed Patients Graph'
   elif choice == 2:
       title = 'Recovered Patients Graph'
   elif choice == 3:
       title = 'Deceased Patients Graph'
   bar_graph(list(tg_daily_api_parser(choice).values()), title)


def covid2():
   # COVID-19
   covidData = tg_current_api_parser()
   allBeds = sum(hospitalsDF['Beds in all Hospitals'].to_list())
   totalConfirmedCases =
covidData['districts']['Unknown']['total']['confirmed']
   totalRecoveredCases =
covidData['districts']['Unknown']['total']['recovered']
   currentCases = totalConfirmedCases - totalRecoveredCases
   covidBeds = allBeds*0.7
   print('Considering that 30% of the beds will be occupied for other
patients. The beds available for COVID patients are: ' + str(covidBeds))
   print('Current COVID-19 Patients: ' + str(currentCases))
   print('Number of patients that can\'t be admitted to a hospital due to
lack of beds: ' + str(currentCases - covidBeds))
   print('Efficiency rate: ' + str((covidBeds/currentCases)*100))

def demo():
   # Demographics
   demoDF.columns = demoDF.columns.str.strip().str.lower().str.replace('
','_')
   district_list = demoDF.districts.tolist()
   males_list = demoDF.males.tolist()
   females_list = demoDF.females.tolist()
   Gender = ['Male','Female']
   urban_list = demoDF.urban.tolist()
   rural_list = demoDF.rural.tolist()

   submenu()
   if choice == 'city':
       cityname = input("Enter City name: ").upper()
       citydf =demoDF[demoDF['districts'] == cityname]
       city_name = citydf.districts.tolist()
       city_males_list = citydf.males.tolist()
       city_females_list = citydf.females.tolist()
       city_rural_list = citydf.rural.tolist()
       city_urban_list = citydf.urban.tolist()
       city_gender_list = [0,0]
       city_gender_list[0] = city_males_list[0]
       city_gender_list[1] = city_females_list[0]
       fig1, ax1 = plt.subplots()
       ax1.pie(city_gender_list, labels=Gender, autopct='%1.1f%%')
       ax1.axis('equal')
       plt.show()

   elif choice == 'overall':
       fig = plt.figure(figsize = (12,9))
       plt.bar(district_list, males_list, align='edge', width=0.3)
       plt.bar(district_list, females_list, align='edge', width=0.3)
       plt.xlabel("Districts")
       plt.xticks(rotation=90)
       plt.ylabel("Gender demography")
       plt.legend(Gender,loc=2)
       plt.show()

def scst():
   scstDF.columns = scstDF.columns.str.strip().str.lower().str.replace(' ','_').str.replace('%','perc')
   district_list = scstDF.districts.tolist()
   sc_ratio = scstDF.perc_sc_population.tolist()
   st_ratio = scstDF.perc_st_population.tolist()
   Legend = ['SC','ST']

   submenu()
   if choice == 'city':
       cityname = input("Enter City name: ").upper()
       citydf =scstDF[scstDF['districts'] == cityname]
       city_sc_ratio = citydf.perc_sc_population.tolist()
       city_st_ratio = citydf.perc_st_population.tolist()
       city_scst_list = [0,0]
       city_scst_list[0] = city_sc_ratio[0]
       city_scst_list[1] = city_st_ratio[0]
       fig1, ax1 = plt.subplots()
       ax1.pie(city_scst_list, labels=Legend, autopct='%1.1f%%')
       ax1.axis('equal')
       plt.show()

   elif choice == 'overall':
       fig = plt.figure(figsize = (12,9))
       plt.bar(district_list,sc_ratio, align='edge', width=0.3)
       plt.bar(district_list,st_ratio, align='edge', width=0.3,
color="pink")
       plt.xlabel("Districts")
       plt.xticks(rotation=90)
       plt.ylabel("% Population")
       plt.legend(Legend,loc=2)
       plt.show()

def menu():
   choice = int(input("""
    Welcome to Telangana!
   1: Literacy and Literacy rate
   2: Current COVID-19 Condition
   3: Demographics
   4: SC/ST Population
   5: Exit
   Please enter your choice: """))

   if choice == 1:
       literacy()
   elif choice == 2:
       covid()
   elif choice == 3:
       demo()
   elif choice==4:
       scst()
   elif choice==5:
       sys.exit
   else:
       print("Invalid choice.")
       menu()
```

```python
    print('Go back to menu? [y/n]')
    i = input().lower()
    if i == 'y':
        menu()
    else:
        pass

print(menu())
```

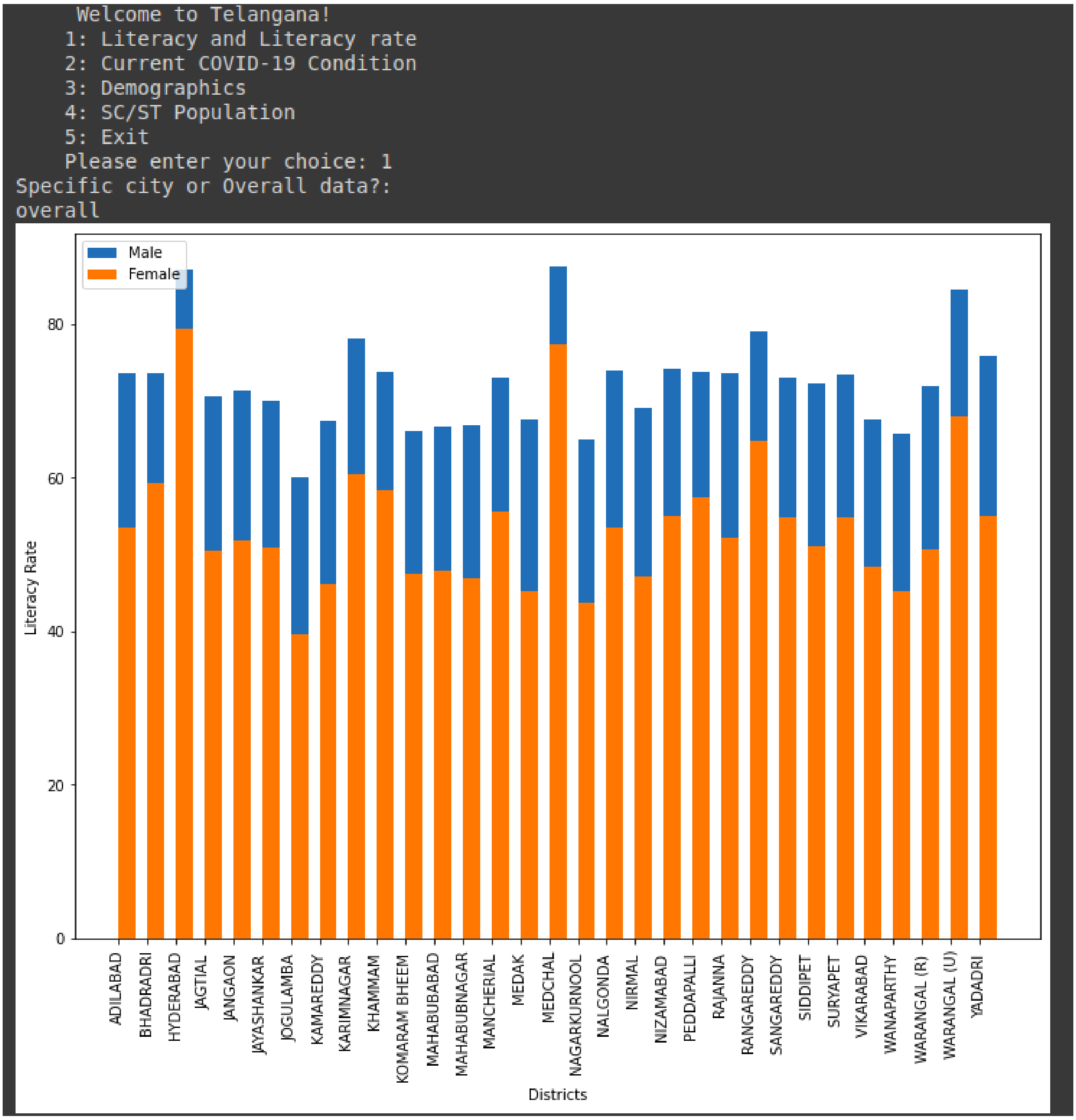
Output Screens

Main Menu:



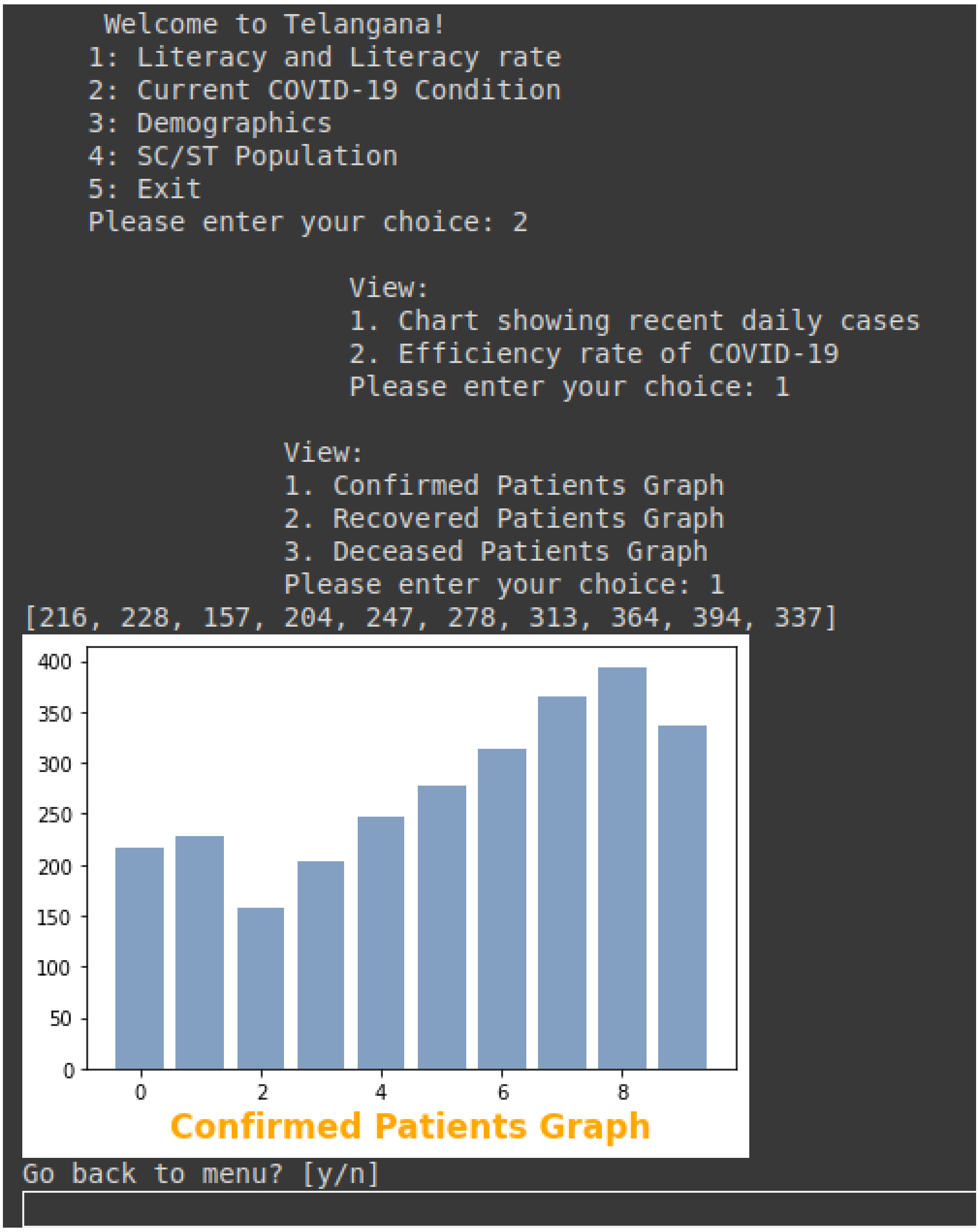
```
 Welcome to Telangana!
1: Literacy and Literacy rate
2: Current COVID-19 Condition
3: Demographics
4: SC/ST Population
5: Exit
Please enter your choice:
```

Operation 1: To check the literacy rate



```
     Welcome to Telangana!
    1: Literacy and Literacy rate
    2: Current COVID-19 Condition
    3: Demographics
    4: SC/ST Population
    5: Exit
    Please enter your choice: 1
Specific city or Overall data?:
overall
```

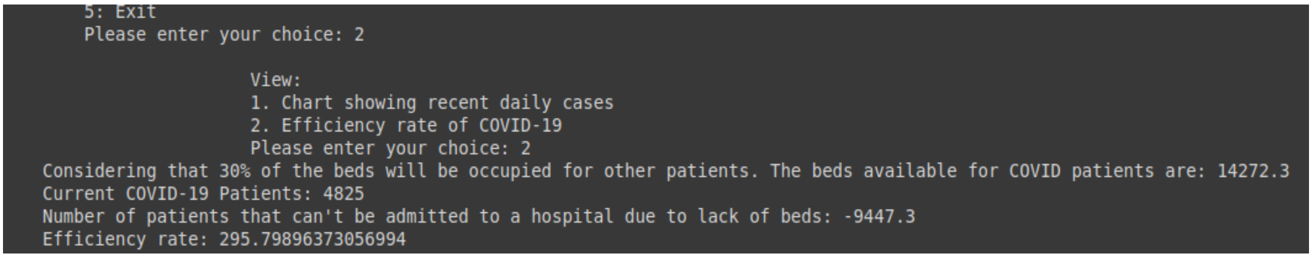
Operation 2: To check the current COVID-19 condition



```
      Welcome to Telangana!
     1: Literacy and Literacy rate
     2: Current COVID-19 Condition
     3: Demographics
     4: SC/ST Population
     5: Exit
     Please enter your choice: 2

                     View:
                     1. Chart showing recent daily cases
                     2. Efficiency rate of COVID-19
                     Please enter your choice: 1

                 View:
                 1. Confirmed Patients Graph
                 2. Recovered Patients Graph
                 3. Deceased Patients Graph
                 Please enter your choice: 1
[216, 228, 157, 204, 247, 278, 313, 364, 394, 337]
```

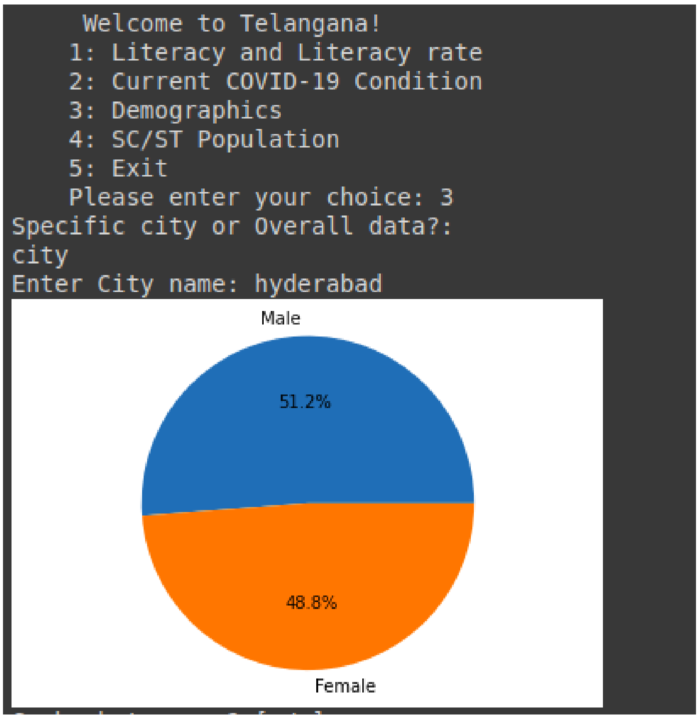
```
Go back to menu? [y/n]
```

(graph of past 10 days as of 23 March 2021)

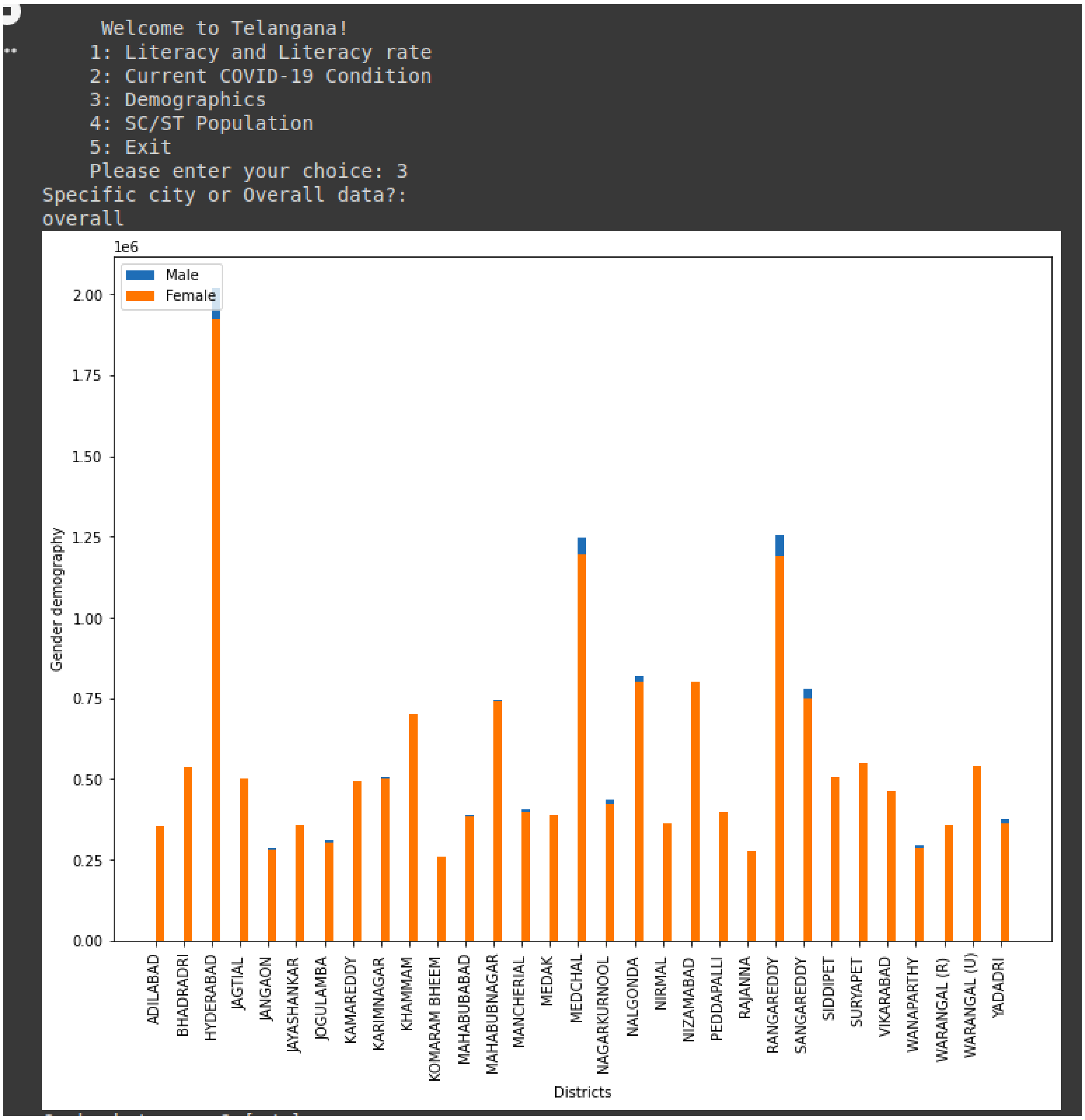


```
     5: Exit
     Please enter your choice: 2

                View:
                1. Chart showing recent daily cases
                2. Efficiency rate of COVID-19
                Please enter your choice: 2
  Considering that 30% of the beds will be occupied for other patients. The beds available for COVID patients are: 14272.3
  Current COVID-19 Patients: 4825
  Number of patients that can't be admitted to a hospital due to lack of beds: -9447.3
  Efficiency rate: 295.79896373056994
```

Operation 3: To check the gender demographics

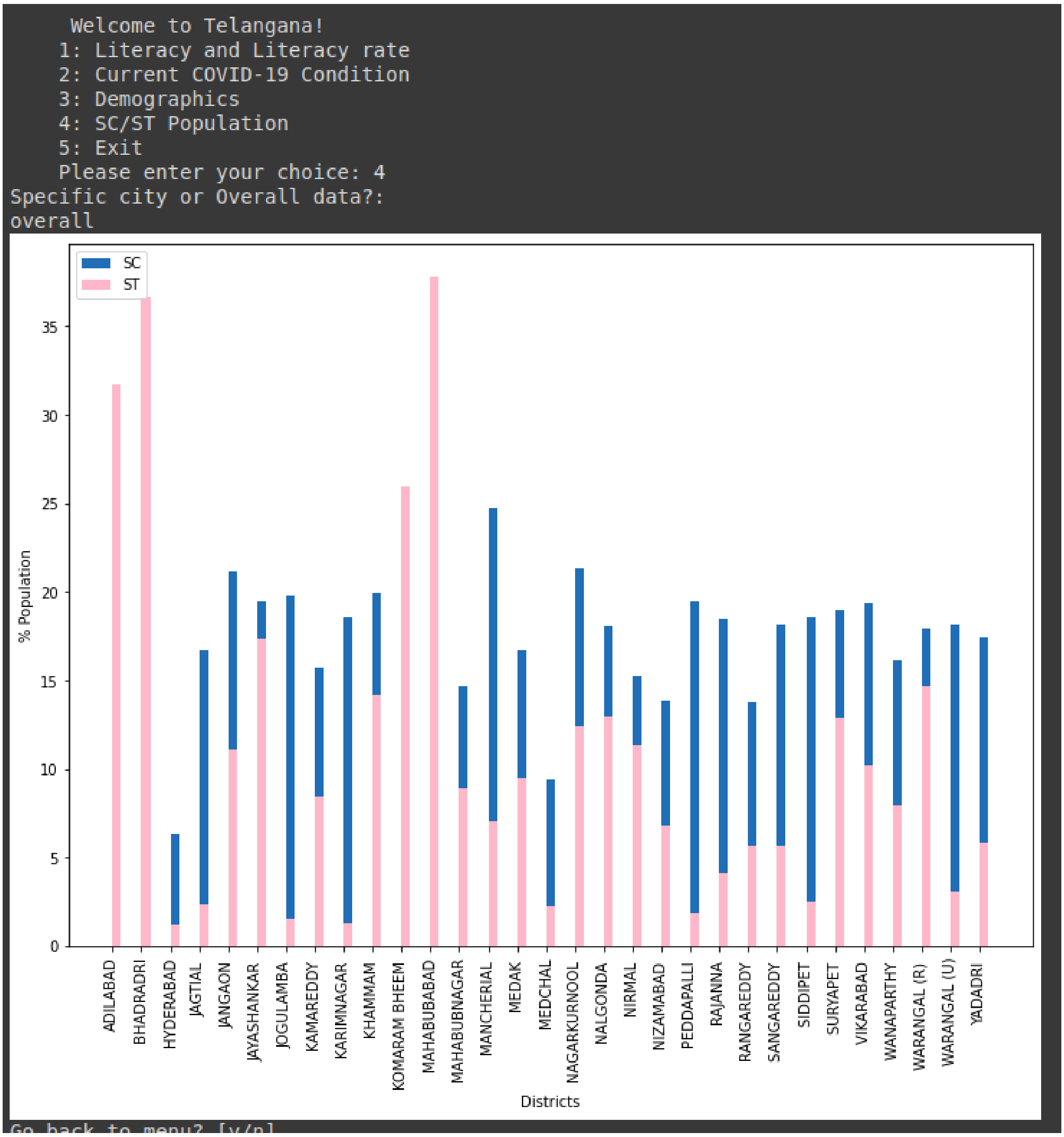


```
     Welcome to Telangana!
    1: Literacy and Literacy rate
    2: Current COVID-19 Condition
    3: Demographics
    4: SC/ST Population
    5: Exit
    Please enter your choice: 3
Specific city or Overall data?:
city
Enter City name: hyderabad
```



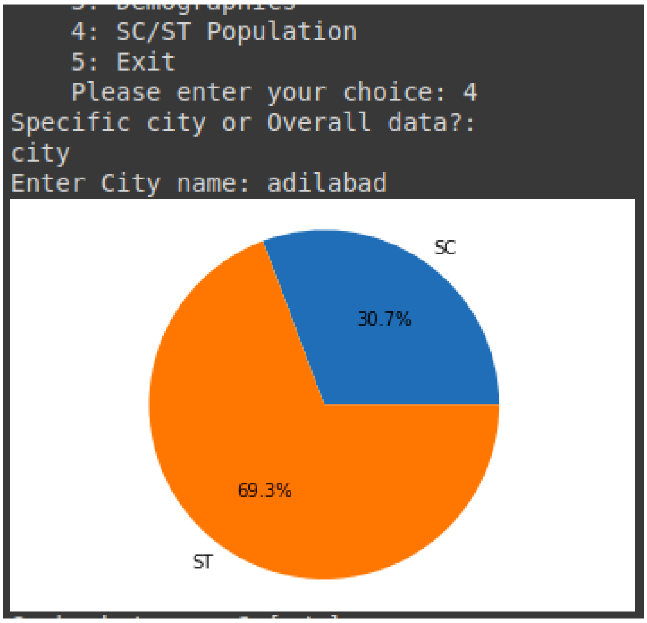
```
     Welcome to Telangana!
    1: Literacy and Literacy rate
    2: Current COVID-19 Condition
    3: Demographics
    4: SC/ST Population
    5: Exit
    Please enter your choice: 3
Specific city or Overall data?:
overall
```

Operation 4: To check the SC/ST population

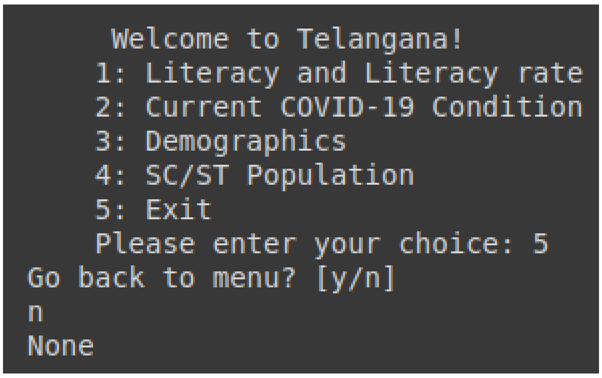


```
     Welcome to Telangana!
    1: Literacy and Literacy rate
    2: Current COVID-19 Condition
    3: Demographics
    4: SC/ST Population
    5: Exit
    Please enter your choice: 4
Specific city or Overall data?:
overall
```

```
Go back to menu? [y/n]
```



```
    3: Demographics
    4: SC/ST Population
    5: Exit
    Please enter your choice: 4
Specific city or Overall data?:
city
Enter City name: adilabad
```



```
     Welcome to Telangana!
    1: Literacy and Literacy rate
    2: Current COVID-19 Condition
    3: Demographics
    4: SC/ST Population
    5: Exit
    Please enter your choice: 5
Go back to menu? [y/n]
n
None
```

Hardware & Software Requirement

Hardware Requirement

PC/Laptop/MacBook with
Intel core/i3/i5/i7 or any equivalent
With at least 2 GB RAM
10 MB free space on Hard Disk
LCD/LED Monitor

Operating System & Compiler

MS Windows/Ubuntu/MacOS

Python IDE with related libraries used for data analysis

- Pandas
- Matplotlib

OR
Colab Online IDE - colab.research.google.com