A PROJECT REPORT

ON

## ***Analysis of Rainfall Data***

Submitted by:

**KRISHANGEE HANDIQUE**

Under the guidance of

**Mr. Prabhat Das**

(PGT-IP)

## 

**ARMY PUBLIC SCHOOL JORHAT**

**MES GATE ROWRIAH JORHAT**

**Assam-785004**

**CERTIFICATE BY PRINCIPAL**

## This is to certify that this project report entitled “Analysis of Rainfall Data” submitted by KRISHANGEE HANDIQUE to Army Public School Jorhat has been examined and evaluated.

The report has been prepared as per the regulations of CBSE and qualifies to be accepted.

Date:

Place:

Mrs. Firdausi Sultana Hazarika

(Principal)

Army Public School Jorhat

**CERTIFICATE BY EXAMINERS**

This is to certify that this project report entitled **“Analysis of Rainfall Data”** is the bonafide work of **KRISHANGEE HANDIQUE** who carried out the project work under my supervision and guidance.

To the best of my knowledge, the matter embodied in the report has not been submitted to any other institute for the award of any other degree.

Date:

Place:

Mr. Prabhat Das

(External Examiner) (InternalExaminer)

**ACKNOWLEDGEMENT**

I take this opportunity to extend my heart full gratitude to Army Public School Jorhat for providing me the opportunity.

I am highly grateful to my guide Mr. Prabhat Das, PGT-IP, Army Public School Jorhat for giving me the opportunity to work under him and providing me an ample guidance and support through the project.

Lastly, I would also like to thank the authors whose publications guided me regarding the project.

**DECLARATION**

I admit that this report is of my own work and all the sources of the information used in this report have fully acknowledged.

I hereby declare that the dissertation work entitled “**Analysis of Rainfall Data**” submitted to the Army Public School Jorhat, is prepared by me and was not submitted to any other institution for award of any other degree.

Date:

Place:

Signature

**Abstract**

Rainfall Prediction is the application of science and technology to predict the amount of rainfall over a region. It is important to exactly determine the rainfall for effective use of water resources, crop productivity and pre-planning of water structures. In this project, we’re going to introduce you to a data science project on rainfall data analysis. We can analyse a lot of data to implement neural networks to predict the average rainfall, maximum and minimum rainfall etc. The user here inputs their e-mail, which then receives an OTP through the program on the mail itself’ asking the user to authenticate the code. On the successful authentication if the code , the user is provided with the list of options, showing them a list of options from the database created in the program. The user is to select the option(s) they wish to see from the given list by feeding in the numbers allotted to them. On feeding in the value(s), they are displayed with the information stored in the database in the form of figures of graphs.

**CONTENTS**

No.                                                                               Page

1. Tools and libraries used                                              1
2. Introduction                                                                     4
3. Project Overview                                                              5

4. Source code                                                                    11

5. Commands used in MySQL                                             17

6. Conclusion and future work                                         18

7. References 19

**List of Figures**

* **Fig 1.**
* **Fig 2.**
* **Fig 3.**
* **Fig 4.**
* **Fig 5.**
* **Fig 6.**
* **Fig 7.**
* **Fig 8.**
* **Fig 9.**
* **Fig 10.**
* **Fig 11.**
* **Fig 12**

**Tools and Libraries Used**

**MySQL**

MySQL is an open-source relational database management system (RDBMS). Its name is a combination of "My", the name of co-founder Michael Widenius's daughter, and "SQL", the abbreviation for Structured Query Language. A relational database organizes data into one or more data tables in which data types may be related to each other; these relations help structure the data. SQL is a language, programmers use to create, modify and extract data from the relational database, as well as control user access to the database. In addition to relational databases and SQL, an RDBMS like MySQL works with an operating system to implement a relational database in a computer's storage system, manages users, allows for network access and facilitates testing database integrity and creation of backups.

MySQL is free and open-source software under the terms of the GNU General Public License, and is also available under a variety of proprietary licenses. MySQL was owned and sponsored by the Swedish company MySQL AB, which was bought by Sun Microsystems (now Oracle Corporation). In 2010, when Oracle acquired Sun, Widenius forked the open-source MySQL project to create MariaDB [1].

**PyCharm**

PyCharm is an [integrated development environment](https://en.wikipedia.org/wiki/Integrated_development_environment) (IDE) used in [computer programming](https://en.wikipedia.org/wiki/Computer_programming), specifically for the [Python](https://en.wikipedia.org/wiki/Python_(programming_language)) language. It is developed by the [Czech](https://en.wikipedia.org/wiki/Czech_Republic) company [JetBrains](https://en.wikipedia.org/wiki/JetBrains).[[5]](https://en.wikipedia.org/wiki/PyCharm#cite_note-5) It provides code analysis, a graphical debugger, an integrated unit tester, integration with [version control systems](https://en.wikipedia.org/wiki/Revision_control) (VCSes), and supports web development with [Django](https://en.wikipedia.org/wiki/Django_(web_framework)) as well as [data science](https://en.wikipedia.org/wiki/Data_science) with [Anaconda](https://en.wikipedia.org/wiki/Anaconda_(Python_distribution)).

PyCharm is [cross-platform](https://en.wikipedia.org/wiki/Cross-platform), with [Windows](https://en.wikipedia.org/wiki/Windows), [macOS](https://en.wikipedia.org/wiki/MacOS) and [Linux](https://en.wikipedia.org/wiki/Linux) versions. The Community Edition is released under the [Apache License](https://en.wikipedia.org/wiki/Apache_License), and there is also Professional Edition with extra features – released under a [proprietary license](https://en.wikipedia.org/wiki/Proprietary_software). [2]

**Pandas**

In [computer programming](https://en.wikipedia.org/wiki/Computer_programming), pandas is a [software library](https://en.wikipedia.org/wiki/Software_library) written for the [Python programming language](https://en.wikipedia.org/wiki/Python_(programming_language)) for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and [time series](https://en.wikipedia.org/wiki/Time_series). It is [free software](https://en.wikipedia.org/wiki/Free_software) released under the [three-clause BSD license](https://en.wikipedia.org/wiki/3-clause_BSD_license). The name is derived from the term "[panel data](https://en.wikipedia.org/wiki/Panel_data)", an [econometrics](https://en.wikipedia.org/wiki/Econometrics) term for data sets that include observations over multiple time periods for the same individuals. Its name is a play on the phrase "Python data analysis" itself. [Wes McKinney](https://en.wikipedia.org/wiki/Wes_McKinney) started building what would become pandas at [AQR Capital](https://en.wikipedia.org/wiki/AQR_Capital) while he was a researcher there from 2007 to 2010 [3].

**Numpy**

NumPy (pronounced /ˈnʌmpaɪ/ ([NUM-py](https://en.wikipedia.org/wiki/Help:Pronunciation_respelling_key)) or sometimes [/ˈnʌmpi/](https://en.wikipedia.org/wiki/Help:IPA/English) ([NUM-pee](https://en.wikipedia.org/wiki/Help:Pronunciation_respelling_key))) is a library for the Python programming language, adding support for large, multi-dimensional [arrays](https://en.wikipedia.org/wiki/Array_data_structure) and [matrices](https://en.wikipedia.org/wiki/Matrix_(math)), along with a large collection of [high-level](https://en.wikipedia.org/wiki/High-level_programming_language) [mathematical](https://en.wikipedia.org/wiki/Mathematics) [functions](https://en.wikipedia.org/wiki/Function_(mathematics)) to operate on these arrays. The ancestor of NumPy, Numeric, was originally created by [Jim Hugunin](https://en.wikipedia.org/wiki/Jim_Hugunin) with contributions from several other developers. In 2005, [Travis Oliphant](https://en.wikipedia.org/wiki/Travis_Oliphant) created NumPy by incorporating features of the competing Numarray into Numeric, with extensive modifications. NumPy is [open-source software](https://en.wikipedia.org/wiki/Open-source_software) and has many contributors. [[4]](https://en.wikipedia.org/wiki/NumPy#cite_note-Nature-5)

**Matplotlib**

Matplotlib is a [plotting](https://en.wikipedia.org/wiki/Plotter) [library](https://en.wikipedia.org/wiki/Library_(computer_science)) for the [Python](https://en.wikipedia.org/wiki/Python_(programming_language)) programming language and its numerical mathematics extension [NumPy](https://en.wikipedia.org/wiki/NumPy). It provides an [object-oriented](https://en.wikipedia.org/wiki/Object-oriented_programming) [API](https://en.wikipedia.org/wiki/API) for embedding plots into applications using general-purpose [GUI toolkits](https://en.wikipedia.org/wiki/GUI_toolkit) like [Tkinter](https://en.wikipedia.org/wiki/Tkinter" \o "Tkinter), [wxPython](https://en.wikipedia.org/wiki/WxPython" \o "WxPython), [Qt](https://en.wikipedia.org/wiki/Qt_(software)), or [GTK+](https://en.wikipedia.org/wiki/GTK%2B). There is also a [procedural](https://en.wikipedia.org/wiki/Procedural_programming) "pylab" interface based on a [state machine](https://en.wikipedia.org/wiki/State_machine) (like [OpenGL](https://en.wikipedia.org/wiki/OpenGL)), designed to closely resemble that of [MATLAB](https://en.wikipedia.org/wiki/MATLAB), though its use is discouraged. [SciPy](https://en.wikipedia.org/wiki/SciPy) makes use of Matplotlib.

Matplotlib was originally written by [John D. Hunter](https://en.wikipedia.org/wiki/John_D._Hunter). Since then it has an active development community and is distributed under a [BSD-style license](https://en.wikipedia.org/wiki/BSD_licenses). Michael Droettboom was nominated as matplotlib's lead developer shortly before John Hunter's death in August 2012 and was further joined by Thomas Caswell.[[5]](https://en.wikipedia.org/wiki/Matplotlib#cite_note-6)

**INTRODUCTION**

Indian Government has undertaken many research studies to analyze the impact of global warming and climate change on rainfall pattern in India. The analyses were made using observed rainfall data from more than 3000 rain-gauge stations spread over the country for 115 years (1901-2015). The major inferences from these studies based on the 115 years of rainfall data are as follows:

The analysis of 115 years of monsoon rainfall data suggests that there is no long term change or trend in the monsoon rainfall averaged over the country. Even though, there are no changes in the all-India rainfall, there are significant changes in annual rainfall in some meteorological sub-divisions. Rainfall over Kerala, East Madhya Pradesh, Jharkhand, Arunachal Pradesh and Nagaland, Manipur, Mizoram and Tripura (NMMT) show decreasing trends. However, rainfall over coastal Karnataka, Maharashtra and Jammu and Kashmir show an increasing trend. There is a general tendency of increasing frequency of extreme rainfall (heavy rainfall events) over India, especially over the central parts of India during the southwest (June- September) monsoon season. There is no evidence of global warming on the observed changes in annual or seasonal rainfall over India. However, there is growing evidence suggesting that increasing frequency of extreme rainfall is due to global warming. The climate change assessment made by the Intergovernmental Panel on Climate Change (IPCC) suggest that in future, frequency of extreme rainfall may increase over India due to increase in global warming. However, there are NO other long term changes/trends in rainfall over India which can be attributed to global warming. The Indian Monsoon is found to be a stable system. With this data with more variations of average rainfall, it is very difficult for a statistical model to predict the required data point. Here we implement neural networks to predict the average rainfall, the neural net is used to create multiple features that helps in predicting the data points with more seasonal variations. The following program is attempt to develop a program that can analyse the data of rainfall from a given dataset. It includes a simple registration system where the user can register him/her into the program with the help of a one-time password (OTP) sent to the email address used for the registration process. The program then gives us a number of options to get different information regarding rainfall present in the given dataset. A graph is also generated using the Matplotlib library that gives us a visual representation of the polarity of the given data.

**PROJECT OVERVIEW**

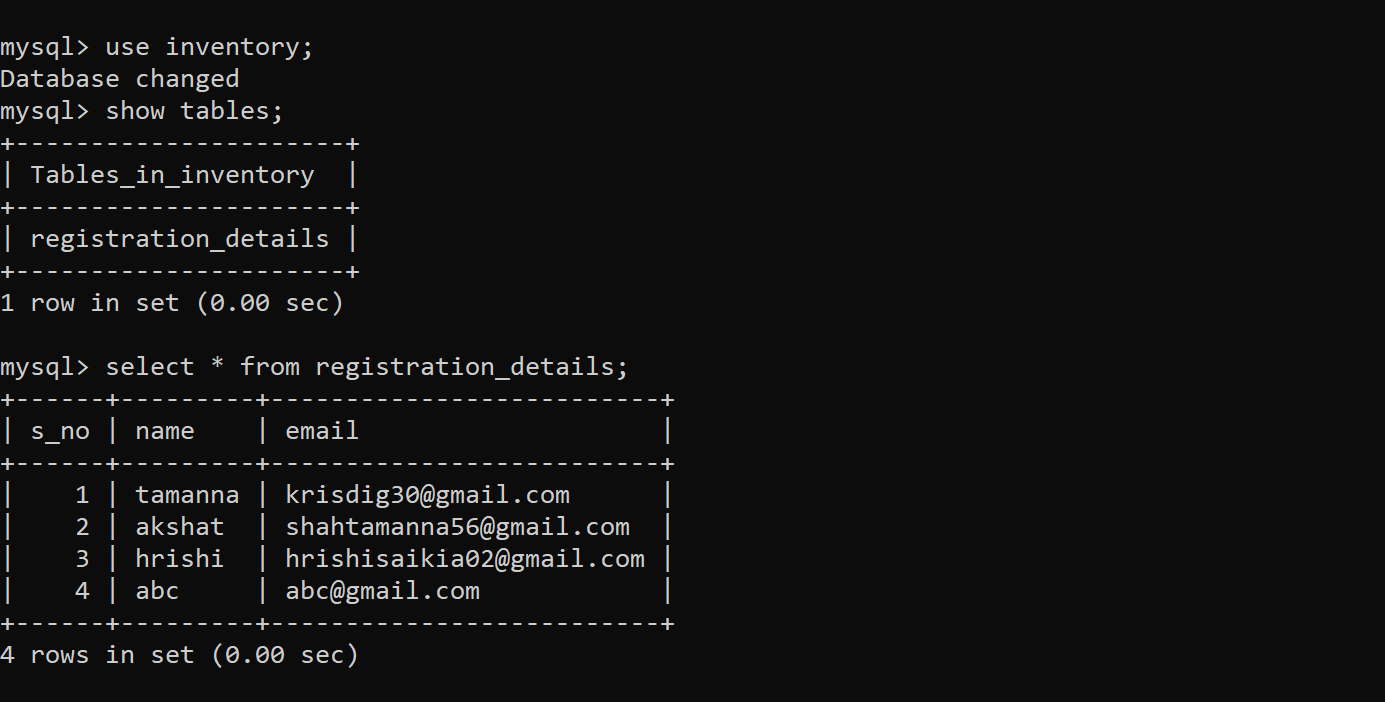
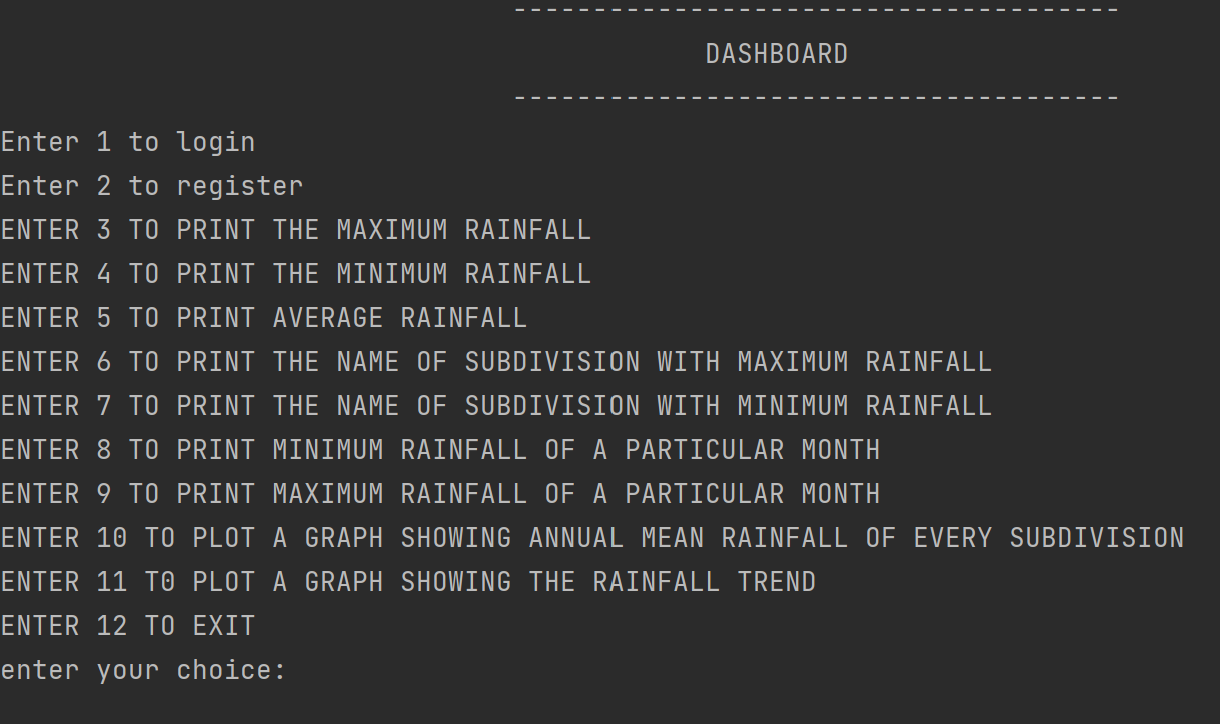
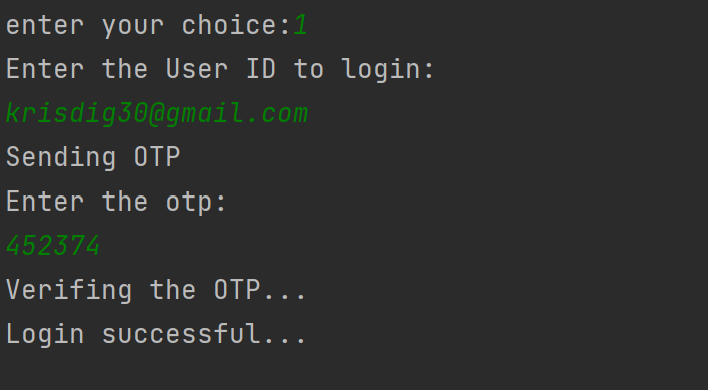
****

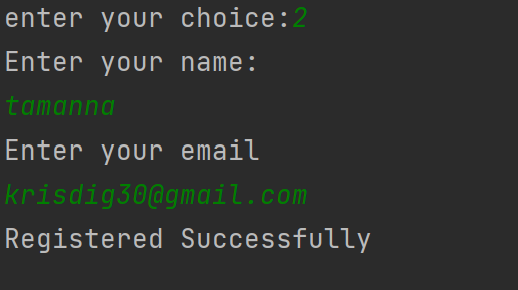
Figure 1: All the information regarding the users is stored in this table

****

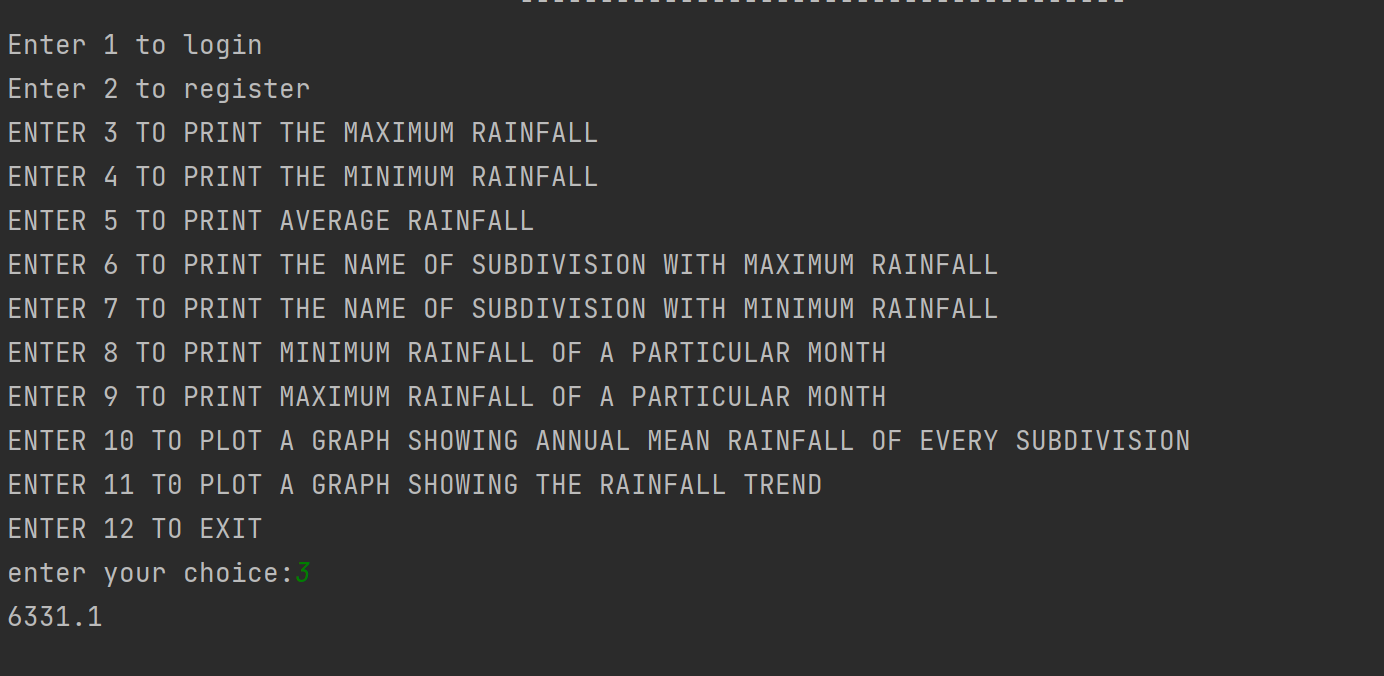
**Figure 2: the user is presented with the following list of options.**

****

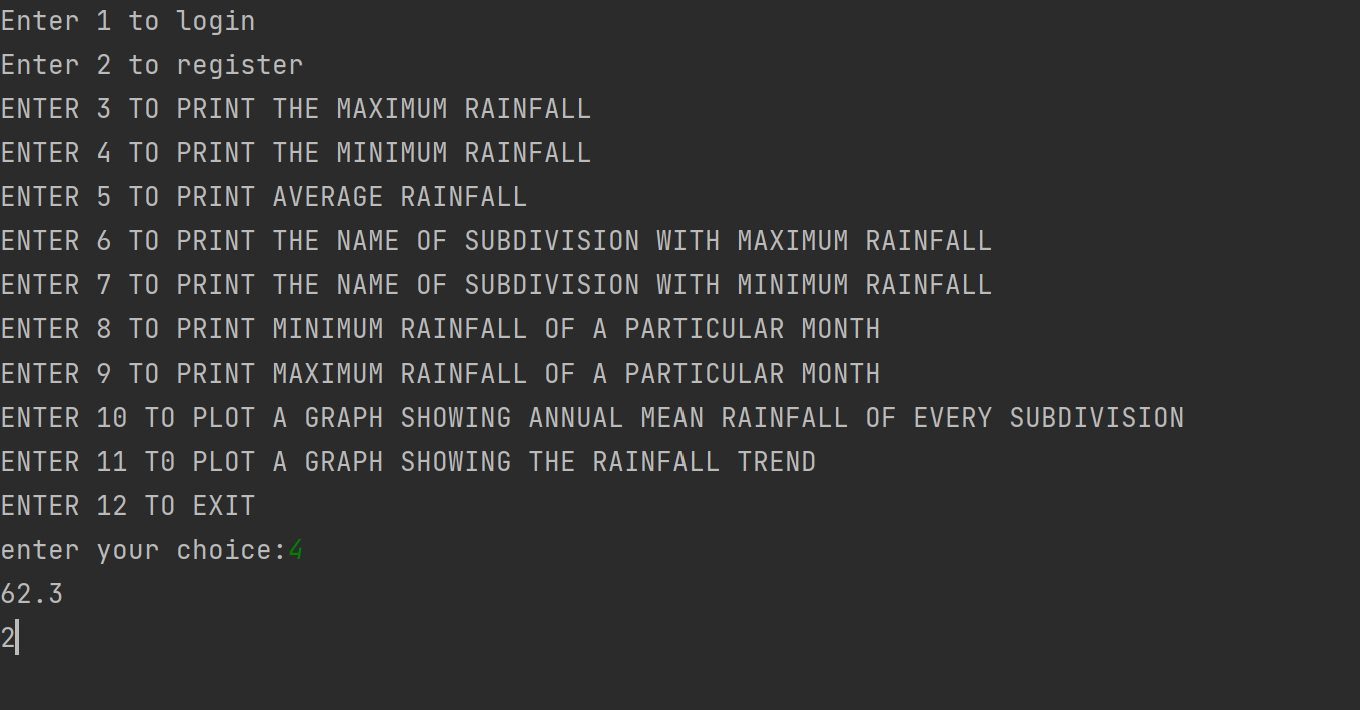
**Figure 2: When the home page is opened, program asks for user’s e-mail address, on entering an e-mail ID, program sends an OTP configuration to user’s e-mail**



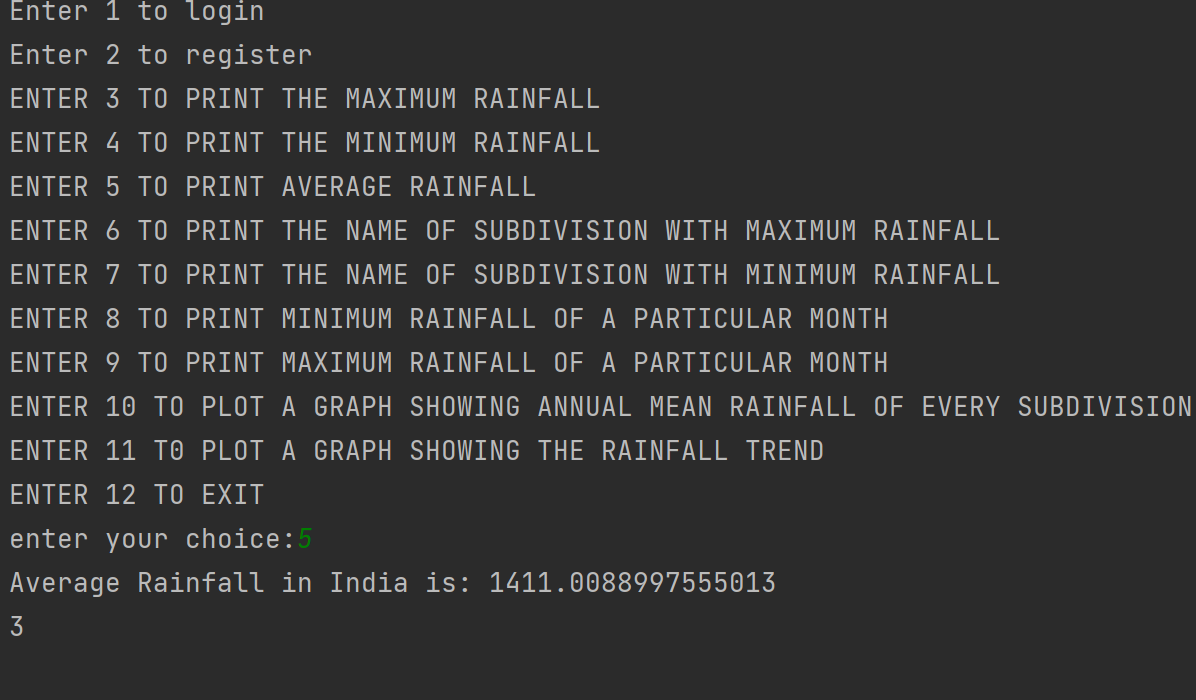
**Figure 3.2 Login procedure for new users**



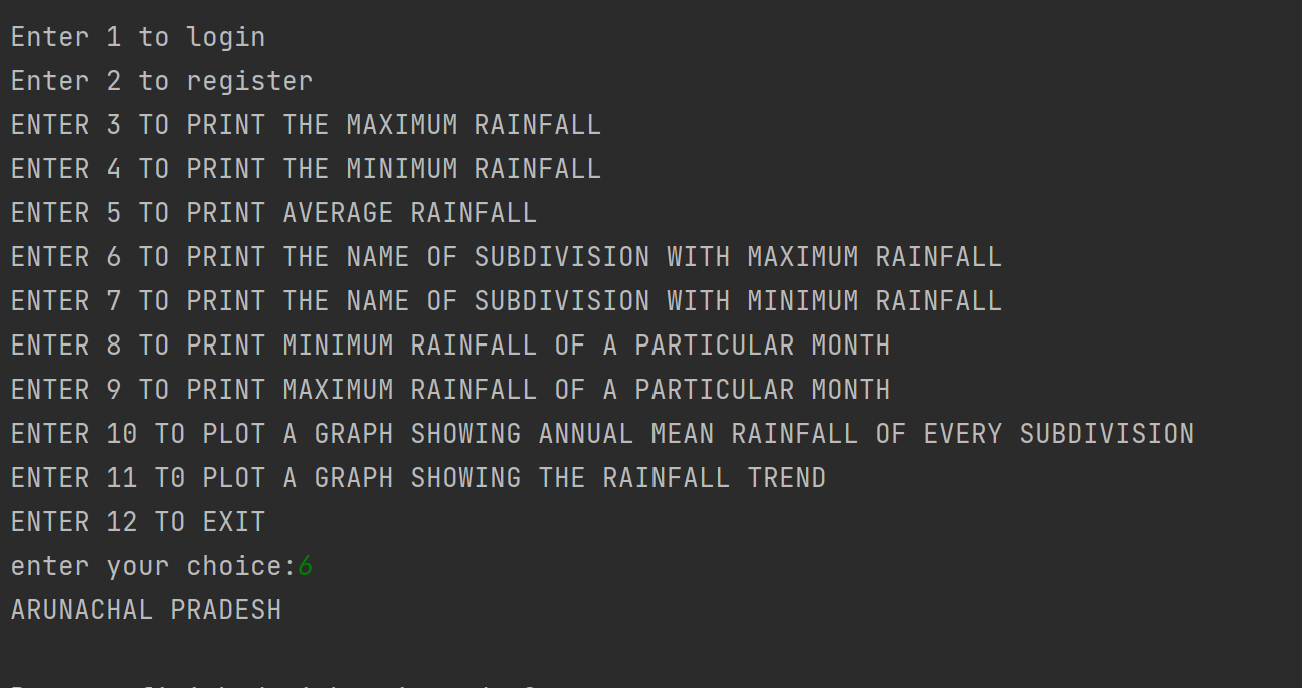
**Fig: Output for option 3 (maximum rainfall)**



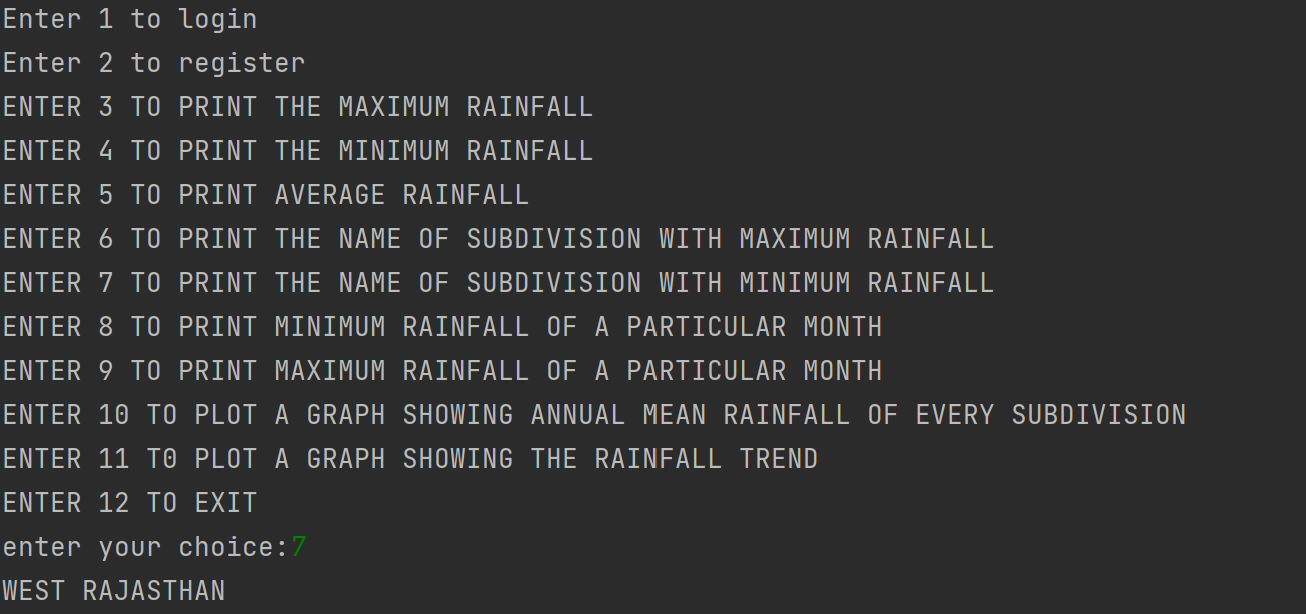
**Fig: Output for option 4(minimum rainfall)**



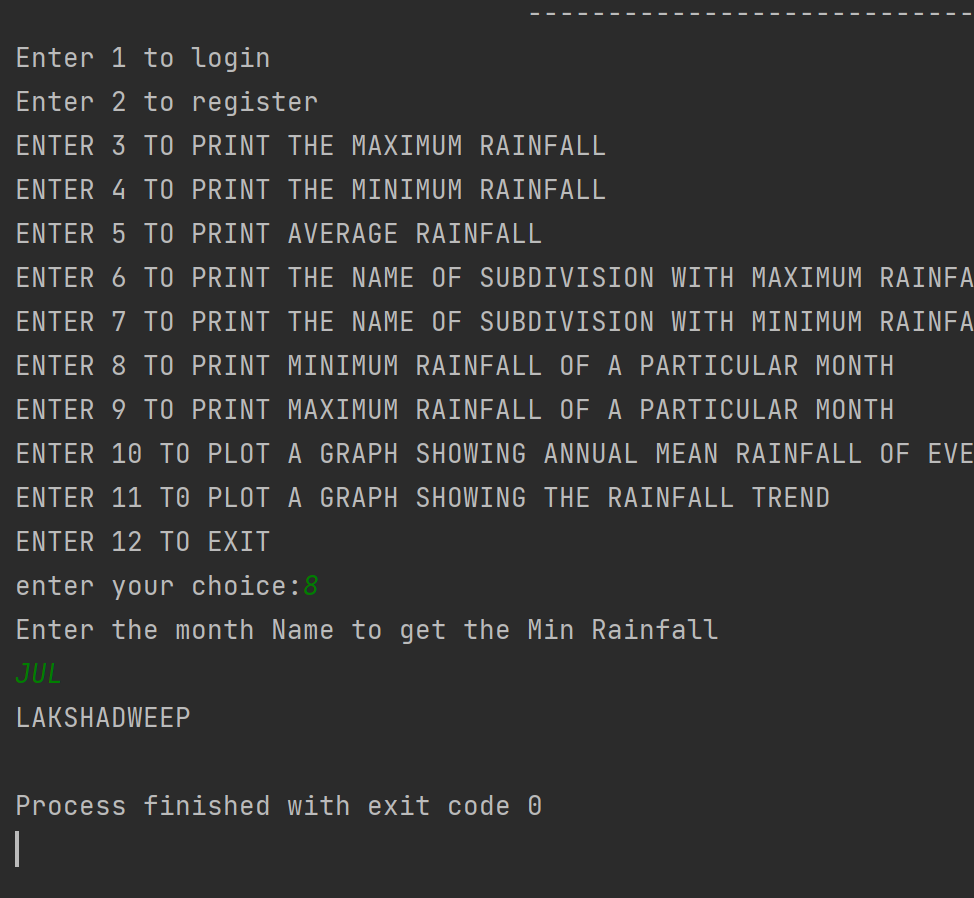
**Fig: output for option 5 (average rainfall)**



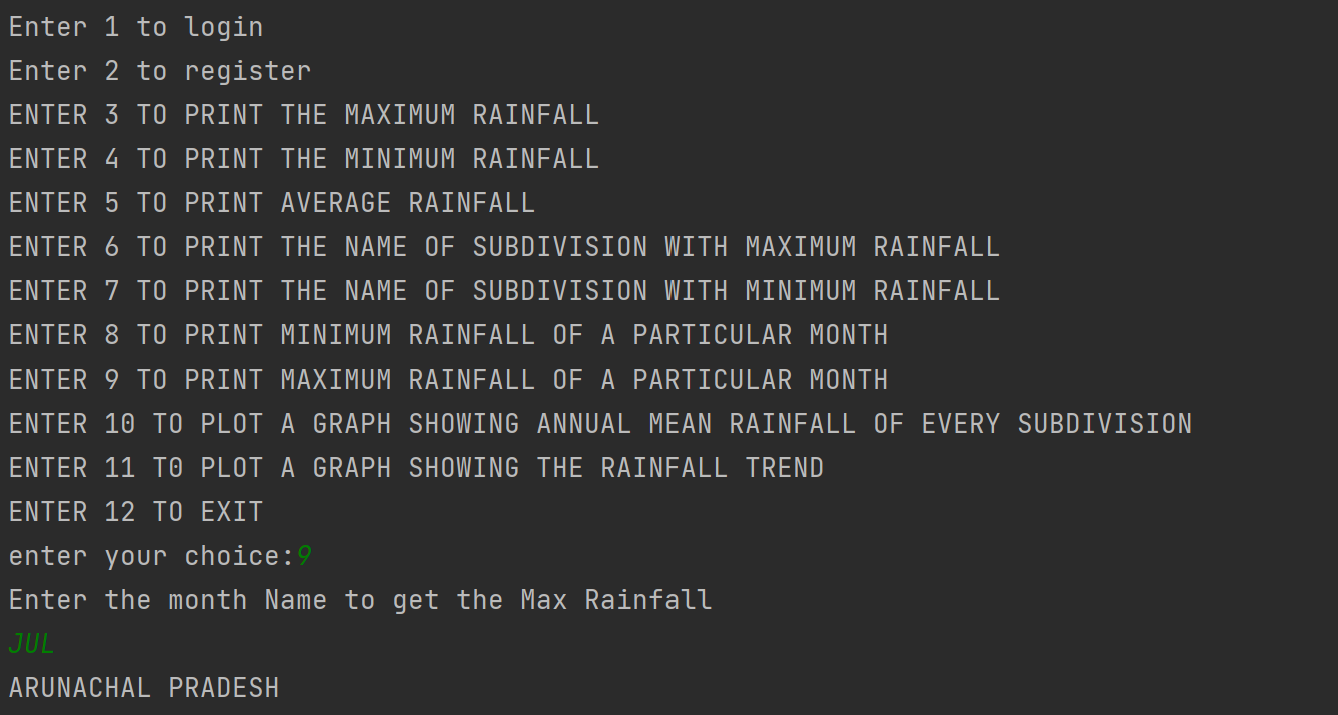
**Fig: output for option 6(name of subdivision with max rainfall)**



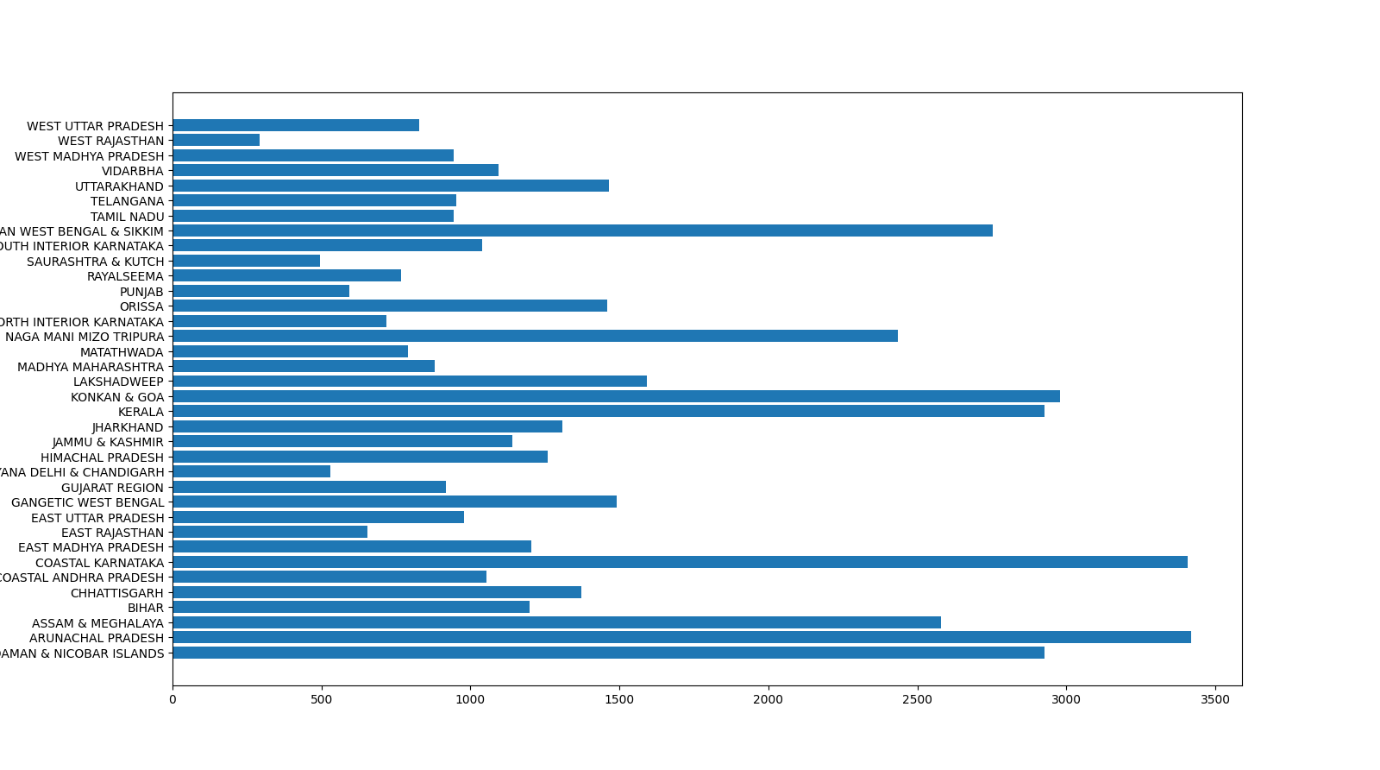
**Fig:output for option 7(name of subdivision with minimun rainfall)**



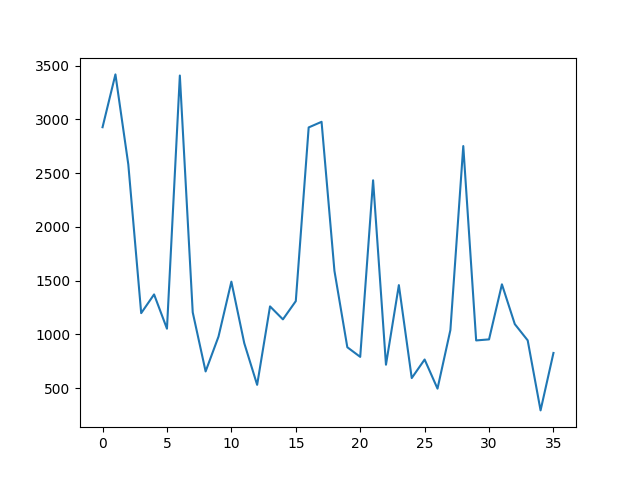
**Fig: output for option 8(minimum rainfall of a particular month)**



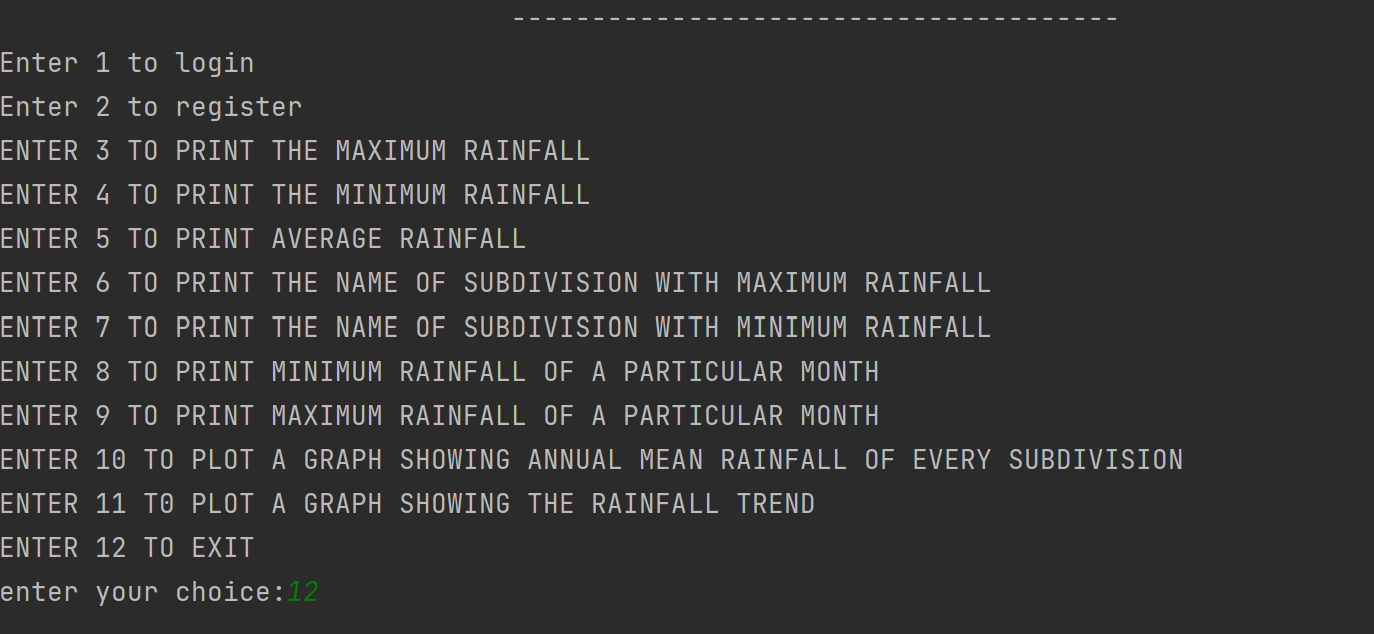
**Fig: output for option 9(maximum rainfall of a particular month)**



**Fig: output for option 10(annual mean rainfall)**



**Fig: output for option 11(rainfall trend)**



**Figure: Output for option 12(Exit)**

**Source Code**

**Main**

import auth  
import otp\_sender  
import dashboard as dash  
print("ENTER MAIL ID")  
email=str(str.lower((input())))  
if auth.auth\_user(email)==1:  
 rcv\_otp=otp\_sender.otp\_sender(email)  
 print("AN OTP HAS BEEN SENT TO THE REG. MAIL ID. PLEASE ENTER THE OTP TO LOGIN !")  
 inp\_otp=input()  
 if rcv\_otp == inp\_otp:  
 print("VALIDATION SUCCESSFUL")  
 dash.dashboard()  
 else:  
 print("INVALID OTP")  
 exit()  
else:  
 print("USER NOT REGISTERED")  
 print("PLEASE SEND AN EMAIL TO group4@apsjorhat.org TO REGISTER")  
 exit()

**Authentication**

import connector as con  
def auth\_user(email):  
 count=0  
 query = "select \* from reg\_users;"  
 con.cursor.execute(query)  
 for i in con.cursor:  
 if i[0]==email:  
 count=1  
 return count

**Connector**

import mysql.connector as mc  
try:  
 dbc = mc.connect(host="localhost",user="root",passwd="root",database="inventory")  
 cursor=dbc.cursor()  
except Exception as e:  
 print(e)

**OTP Sender**

import smtplib  
import random  
def otp\_sender(user\_id):  
 s= smtplib.SMTP('smtp.gmail.com', 587)  
 s.starttls()  
 s.login('group4@apsjorhat.org', 'apsj#12345678')  
 otp= random.randint(111111,999999)  
 message= str(otp)  
 s.sendmail("group4@apsjorhat.org",user\_id,message)  
 s.quit()  
 print("Enter the otp: ")  
 val= int(input())  
 if val==otp:  
 print("Verifing the OTP...")  
 print("Login successful...")  
  
 else:  
 print("Incorrect OTP...")

**Dashboard**

import functions  
import connector as con  
import otp\_sender  
  
def registration():  
 print("Enter your name: ")  
 name=input()  
 print("Enter your email")  
 email=input()  
 query="insert into registration\_details (name,email) values" +"('"+name+"','"+email+"');"  
 print("Registered Successfully")  
 con.cursor.execute(query)  
 con.dbc.commit()  
  
def login():  
 print("Enter the User ID to login:")  
 user\_id = input()  
 fetch\_query="select \* from registration\_details;"  
 con.cursor.execute(fetch\_query)  
 count=0  
 for i in con.cursor:  
 if user\_id ==i[2]:  
 count = count+1  
 print("Sending OTP")  
 otp\_sender.otp\_sender(user\_id)  
 if count==0:  
 print("User not registered")  
  
  
inp=0  
print(" --------------------------------------")  
print(" DASHBOARD")  
print(" --------------------------------------")  
print("Enter 1 to login ")  
print("Enter 2 to register")  
print("ENTER 3 TO PRINT THE MAXIMUM RAINFALL")  
print("ENTER 4 TO PRINT THE MINIMUM RAINFALL")  
print("ENTER 5 TO PRINT AVERAGE RAINFALL")  
print("ENTER 6 TO PRINT THE NAME OF SUBDIVISION WITH MAXIMUM RAINFALL")  
print("ENTER 7 TO PRINT THE NAME OF SUBDIVISION WITH MINIMUM RAINFALL")  
print("ENTER 8 TO PRINT MINIMUM RAINFALL OF A PARTICULAR MONTH")  
print("ENTER 9 TO PRINT MAXIMUM RAINFALL OF A PARTICULAR MONTH")  
print("ENTER 10 TO PLOT A GRAPH SHOWING ANNUAL MEAN RAINFALL OF EVERY SUBDIVISION")  
print("ENTER 11 T0 PLOT A GRAPH SHOWING THE RAINFALL TREND")  
print("ENTER 12 TO EXIT")  
  
inp=int(input("enter your choice:"))  
if inp==1:  
 login()  
elif inp==2:  
 registration()  
elif inp == 3:  
 functions.max\_rainfall()  
elif inp==4:  
 functions.min\_rainfall()  
elif inp==5:  
 functions.avg\_rainfall()  
elif inp==6:  
 functions.name\_max()  
elif inp==7:  
 functions.name\_min()  
elif inp==8:  
 functions.month\_min()  
elif inp==9:  
 functions.month\_max()  
elif inp==10:  
 functions.graph\_1()  
elif inp==11:  
 functions.graph\_2()

**Functions**

import pandas as pd  
import matplotlib.pyplot as plt  
df=pd.read\_csv(r"C:\Users\digan\OneDrive\Desktop\rainfall\_dataset.csv")  
  
  
def max\_rainfall():  
 print(df['ANNUAL'].max())  
  
  
def min\_rainfall():  
 print(df['ANNUAL'].min())  
 print(2)  
  
  
def avg\_rainfall():  
 no\_of\_sub\_div = df['ANNUAL'].count()  
 total\_rainfall = df['ANNUAL'].sum()  
 print("Average Rainfall in India is:", total\_rainfall / no\_of\_sub\_div)  
 print(3)  
  
  
def name\_max():  
 max\_rainfall = df['ANNUAL'].max()  
 val = df[df['ANNUAL'] == max\_rainfall]  
 for i in val['SUBDIVISION']:  
 print(i)  
  
  
def name\_min():  
 min\_rainfall = df['ANNUAL'].min()  
 val = df[df['ANNUAL'] == min\_rainfall]  
 for i in val['SUBDIVISION']:  
 print(i)  
  
  
def month\_min():  
 print("Enter the month Name to get the Min Rainfall")  
 month = input()  
 min\_month = df[month].min()  
 val = df[df[month] == min\_month]  
 for i in val['SUBDIVISION']:  
 print(i)  
  
  
def month\_max():  
 print("Enter the month Name to get the Max Rainfall")  
 month = input()  
 min\_month = df[month].max()  
 val = df[df[month] == min\_month]  
 for i in val['SUBDIVISION']:  
 print(i)  
  
  
def graph\_1():  
 groups = df.groupby(by='SUBDIVISION')  
 list\_of\_sub\_div = list(groups.groups.keys())  
 list\_of\_annual\_mean\_rainfall = []  
 for i in groups.groups.keys():  
 data = groups.get\_group(i)  
 value = data['ANNUAL'].mean()  
 list\_of\_annual\_mean\_rainfall.append(value)  
 plt.barh(list\_of\_sub\_div, list\_of\_annual\_mean\_rainfall)  
 plt.show()  
  
  
def graph\_2():  
 groups = df.groupby(by='SUBDIVISION')  
 list\_of\_sub\_div = list(groups.groups.keys())  
 list\_of\_annual\_mean\_rainfall = []  
 for i in groups.groups.keys():  
 data = groups.get\_group(i)  
 value = data['ANNUAL'].mean()  
 list\_of\_annual\_mean\_rainfall.append(value)  
 plt.plot(list\_of\_annual\_mean\_rainfall)  
 plt.show()

**Commands Used in MySQL**

**Creating Database**

Create database analysis\_rainfall\_data;

**Using Database**

Use analysis\_rainfall\_data;

**Creating Table and Inserting Values**

Create table registration\_details(id int(10) primary key auto\_increment ,name varchar(100),email varchar(100) unique);

**Desc table**

Desc registration\_details**;**

**Inserting values**

insert into reg\_users values((1,"tamanna ", "krisdig30@gmail.com"),(2,"akshat","shahtamanna56@gmail.com"),(3,”hrishi","hrishisaikia02@gmail.com"),(4, “abc” “abc@gmail.com”);

**To Fetch All Values**

Select \* from registration\_details;

**Conclusion and Future Work**

By visualising data in form of bar and line graphs we are able to easily analyse the trend in the monsoon rainfall averaged over the country.

Most of the world’s data is unstructured and unorganised and a program such as this can help sort the data in an efficient manner and generate useful information regarding the rainfall pattern in India.

The functions performed by this project program are in accordance to my assumptions for further upgradation, we can also add features such as a Graphical User Interface to this project, which can be developed in the form of a web, desktop or mobile application. This project can also be scaled to analyse global data in real time.

**References**

[1] <https://en.wikipedia.org/wiki/MySQL>

[2] <https://en.wikipedia.org/wiki/PyCharm>

[3] <https://en.wikipedia.org/wiki/Pandas_(software)>

[4 ]<https://textblob.readthedocs.io/en/dev/>

[5] <https://en.wikipedia.org/wiki/Matplotlib>