**SHIV NADAR SCHOOL GURUGRAM**

**COMPUTER SCIENCE**

**PROJECT**

**ACCOUNT RECORDER**



**Submitted by:**

**Varun Mohanta**

**COMPUTER SCIENCE**

**PROJECT**

**ACCOUNT RECORDER**

***SUBMITTED BY:***

**Varun Mohanta**

***Under the guidance of***

**Mr. Ekta Kaper (PGT CS)**

**CONTENTS**

|  |  |  |  |
| --- | --- | --- | --- |
| S.No. | Topic | | Page No. |
| 1 | **Introduction** | | 5 |
|  | * + Introduction to Python | | 7 |
|  | * + Features of Python | | 8 |
|  | * + Introduction to MYSQL | | 9 |
|  | * + Features of MySQL | | 10 |
|  | * + Project Synopsis (Project Brief) | | 12 |
|  | * + Flowchat | | 15 |
| 2 | **Coding** | | 18 |
| 3 | | **Output** | 24 |
| 4 | | **Bibliography** | 30 |

**CERTIFICATE**

**This is to certify that Varun Mohanta of Class XI has prepared the report on Project entitled “Account Recorder”. The report is the result of their efforts and endeavours. The report is found worthy of acceptance as final project report for the subject Computer Science of Class XII as prescribed by CBSE for AISSCE 2021-2022 in Shiv Nadar School, Pahari Road, Sector 26, DLF City, Phase -1, Block -E, Gurugram, Haryana 122011.**

**Roll Number: Date: \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_**

**Signature of Signature of**

**Internal Examiner: External Examiner:**

***\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

**ACKNOWLEDGEMENT**

**I would like to express my special thanks of gratitude to my teacher Ms. Ekta Kaper, PGT(CS) as well as our Principal Ms. Monica Sagar who gave me the golden opportunity to do this wonderful project on the topic “Account Recorder”, which also helped me in doing a lot of research and I came to know about so many new things I am really thankful to them.**

**Finally, yet importantly, I would like to express my heartfelt thanks to my beloved parents for their blessings, my friends/classmates for their help and wishes for the successful completion of this project within the limited period.**

**Varun Mohanta**

**DECLARATION**

**I hereby declare that the project work entitled “Account Recorder” submitted to Shiv Nadar School, Gurugram for the subject Computer Science under the guidance of Ms. Ekta Kaper, PGT(CS) is a record of original work done by me.**

**I further declare that this project or any part of it has not been submitted elsewhere for any other class.**

**Class:**

**Place:**

**Date:**

**INTRODUCTION TO PYTHON**

Python is an interpreted, high-level, general-purpose programming language. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace. It provides constructs that enable clear programming on both small and large scales. Van Rossum led the language community until stepping down as leader in July 2018.

Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-riented, imperative, functional and procedural, and has a large and comprehensive standard library.

Python interpreters are available for many operating systems. CPython, the reference implementation of Python, is open source software and has a community-based development model, as do nearly all of Python's other implementations. Python and CPython are managed by the non-profit Python Software Foundation.

**FEATURES OF PYTHON**

There are many features in Python –

* **Easy-to-learn** − Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.
* **Easy-to-read** − Python code is more clearly defined and visible to the eyes.
* **Easy-to-maintain** − Python's source code is fairly easy-to-maintain.
* **A broad standard library** − Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.
* **Interactive Mode** − Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
* **Portable** − Python can run on a wide variety of hardware platforms and has the same interface on all platforms.
* **Extendable** − You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.
* **Databases** − Python provides interfaces to all major commercial databases.
* **GUI Programming** − Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.
* **Scalable** − Python provides a better structure and support for large programs than shell scripting.
* **Dynamically Typed Language** − Python is a dynamically-typed language. That means the type (for example- int, double, long, etc.) for a variable is decided at run time not in advance because of this feature we don’t need to specify the type of variable.

**INTRODUCTION TO 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](https://en.wikipedia.org/wiki/Proprietary_software) licenses. MySQL was owned and sponsored by the [Swedish](https://en.wikipedia.org/wiki/Sweden) company [MySQL AB](https://en.wikipedia.org/wiki/MySQL_AB), which was bought by [Sun Microsystems](https://en.wikipedia.org/wiki/Sun_Microsystems) (now Oracle Corporation). In 2010, when Oracle acquired Sun, Widenius [forked](https://en.wikipedia.org/wiki/Fork_(software_development)) the [open-source](https://en.wikipedia.org/wiki/Open-source) MySQL project to create [MariaDB](https://en.wikipedia.org/wiki/MariaDB).

MySQL has stand-alone clients that allow users to interact directly with a MySQL database using SQL, but more often, MySQL is used with other programs to implement applications that need relational database capability. MySQL is a component of the [LAMP](https://en.wikipedia.org/wiki/LAMP_(software_bundle)) [web application](https://en.wikipedia.org/wiki/Web_application) [software stack](https://en.wikipedia.org/wiki/Software_stack) (and [others](https://en.wikipedia.org/wiki/List_of_AMP_packages)), which is an acronym for [Linux](https://en.wikipedia.org/wiki/Linux), [Apache](https://en.wikipedia.org/wiki/Apache_HTTP_Server), MySQL, [Perl](https://en.wikipedia.org/wiki/Perl)/[PHP](https://en.wikipedia.org/wiki/PHP)/[Python](https://en.wikipedia.org/wiki/Python_(programming_language)).

**FEATURES OF MySQL**

* **Ease of Management –**The software very easily gets downloaded and also uses an event scheduler to schedule the tasks automatically.
* **Robust Transactional Support –**Holds the ACID (Atomicity, Consistency, Isolation, Durability) property, and also allows distributed multi-version support.
* **Comprehensive Application Development –**MySQL has plugin libraries to embed the database into any application. It also supports stored procedures, triggers, functions, views and many more for application development.
* **High Performance –**Provides fast load utilities with distinct memory caches and table index partitioning.
* **Low Total Cost Of Ownership –**This reduces licensing costs and hardware expenditures.
* **Open Source & 24 \* 7 Support –** This RDBMS can be used on any platform and offers 24\*7 support for open source and enterprise edition.
* **Secure Data Protection –**MySQL supports powerful mechanisms to ensure that only authorized users have access to the databases.
* **High Availability –**MySQL can run high-speed master/slave replication configurations and it offers cluster servers.
* **Scalability & Flexibility –**With MySQL you can run deeply embedded applications and create data warehouses holding a humongous amount of data.

**HARDWARE**

**AND**

**SOFTWARE USED**

**HARDWARE REQUIREMENTS:**

**Processer** Core i3 7th Gen

**RAM**  4GB

**Hard Disk Drive** 500GB

**Monitor** 18.5 LED monitor

**Mouse** Logitech

**Keyboard** 104 Keys Keyboard

**Port** USB Port

**SOFTWARE REQUIREMENTS:**

**Operating System** Windows 10

**Front End Tool** Python

**Back End Tool** MySQL and Binary file

**Report tool**  Data Report/Microsoft Word

**Project Aim and Objectives**

The project aims and objectives that will be achieved after   
completion of the system is as follows:

* To keep a list of accounts
* To show the password of any account
* To add or remove account
* To change password of the account
* To be user friendly
* The future scope is to categorise the accounts and increase security for those accounts which are related to banking & e-commerce.

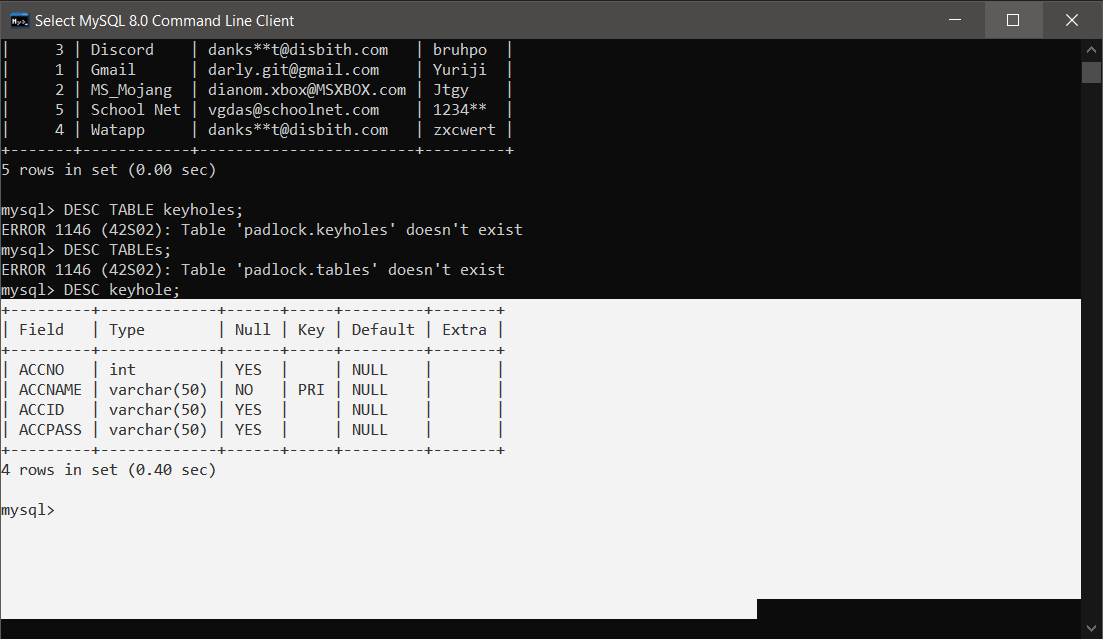
**Background of the Project**

People who have multiple accounts and can’t remember their password have a lot of issues signing in their accounts. Now a days, many softwares and apps need account to give services to the user. So, to tackle this problem I have developed a software known as ‘Account Recorder’ which stores all account details.

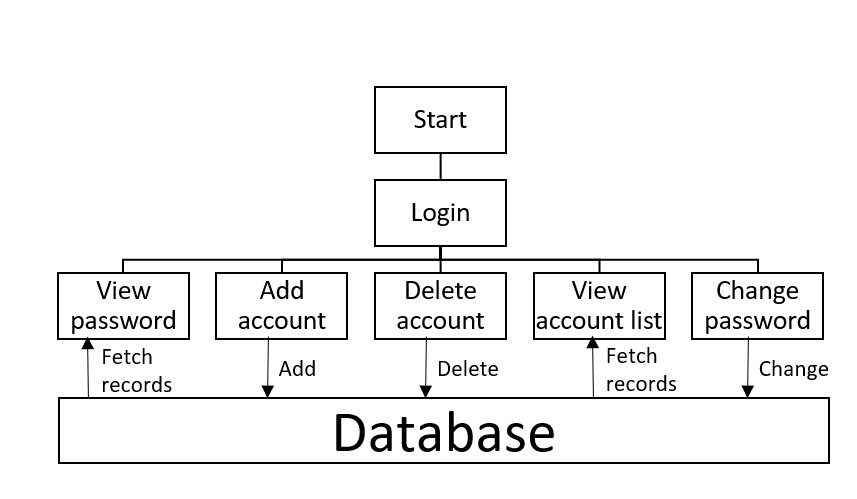
**Project Scope**

The aim of the project is that the software should store accounts from the user and show the password. Also it is user friendly.

To show the password only to the user, I used a special module called as Pygame which is out of my CBSE curriculum. It is only used for making 2D video games. So I used this module for its GUI features. I also used studio mask, a module in which whatever password is written over the executable python file is hidden in asterisk and can’t be copied. MySQL connector is used to connect the program with the database and allow the program to fetch or append data. Software is user friendly. There are detailed messages for each feature in the program. The programs are that the user can add an account, delete an account, view the password, change the password and view the account list from the database. If the user uses the app for the first time, then it ask the details of the user and stores it in the binary file. The binary file is a stamp on a device which tells the software that it has a database.

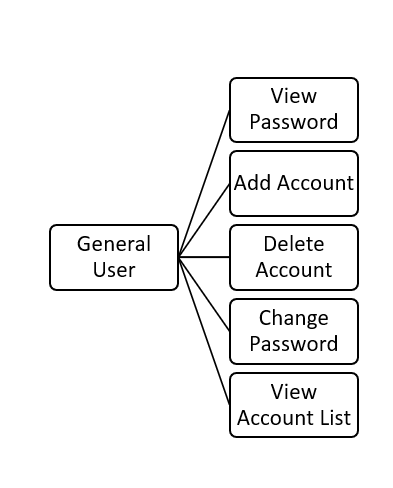
**Table Design**

**System Design**



End

**Use Case Diagram**

****

**CODING**

import mysql.connector as M

import stdiomask as P

import pygame as D

import pickle as Z

import time as T

def askpassword():

Tng=dict()

e=input('\nEnter your email id :- ')

ce=e.partition('@')

if ce[1]=='@' and ce[2][-4::1]=='.com':

print('Fetching account details...')

print('Press <spacebar> to show the password\nPress <Esc> or CLOSE button to close the window ')

print("Use the 'Left' or 'Right' arrow key to shift to other accounts")

urve=(e,)

t=("SELECT ACCNAME,ACCPASS FROM keyhole WHERE ACCID=%s")

kurs.execute(t,urve)

za=kurs.fetchall()

for i in range(1,len(za)+1):

Tng[i]=za[i-1]+('\*\*\*\*',)

if Tng=={}:

print("\nThe email id doesn't exist")

print("Press '~' to start the function again")

else:

D.init()

h=1

surface=D.display.set\_mode((468,150))

D.display.set\_caption('Password Viewer')

Font=D.font.Font('freesansbold.ttf',26)

Text\_Render=Font.render(Tng[h][0],True,(0,255,0))

Text\_rect=Text\_Render.get\_rect()

Text\_rect.center=(70,26)

Text\_Render\_1=Font.render(Tng[h][2],True,(0,255,0))

Text\_rect\_1=Text\_Render.get\_rect()

Text\_rect\_1.center=(285,31)

Text\_Render\_2=Font.render(str(h),True,(0,255,0))

Text\_rect\_2=Text\_Render.get\_rect()

Text\_rect\_2.center=(209,117)

Text\_Render\_3=Font.render('/'+str(len(Tng)),True,(0,255,0))

Text\_rect\_3=Text\_Render.get\_rect()

Text\_rect\_3.center=(226,118)

i=0

Loop=True

while Loop:

for event in D.event.get():

if event.type == D.QUIT:

Loop=False

keys=D.key.get\_pressed()

if keys[D.K\_ESCAPE]:

Loop=False

if keys[D.K\_SPACE]:

i=1

if not(keys[D.K\_SPACE]):

i=0

if len(Tng) != 1:

if keys[D.K\_LEFT]:

T.sleep(0.25)

h-=1

if h < 1:

h=len(Tng)

Text\_Render=Font.render(Tng[h][0],True,(0,255,0))

Text\_Render\_2=Font.render(str(h),True,(0,255,0))

if keys[D.K\_RIGHT]:

T.sleep(0.25)

h+=1

if h > len(Tng):

h=1

Text\_Render=Font.render(Tng[h][0],True,(0,255,0))

Text\_Render\_2=Font.render(str(h),True,(0,255,0))

if i:

Text\_Render\_1=Font.render(Tng[h][1],True,(0,255,0))

else:

Text\_Render\_1=Font.render(Tng[h][2],True,(0,255,0))

surface.fill((0,0,0))

surface.blit(Text\_Render,Text\_rect)

surface.blit(Text\_Render\_1,Text\_rect\_1)

surface.blit(Text\_Render\_2,Text\_rect\_2)

surface.blit(Text\_Render\_3,Text\_rect\_3)

D.display.update()

D.quit()

else:

print('Invalid email id')

print("Your email id should contain an '@' and '.com'")

print("Press '~' to start the function again")

def appendacc():

s=0

kurs.execute("SELECT MAX(ACCNO) FROM keyhole")

for i in kurs:

if i[0] == None:

s=1

else:

s=i[0]+1

kurs.execute("SELECT ACCNAME FROM keyhole")

jkl=kurs.fetchall()

while True:

a=input('\nEnter your account name :- ')

if a in jkl:

print('The Account Name','\''+a+'\'','already exists in the database')

print('Try using a different name')

else:

break

b=input('Enter your email id :- ')

ce=b.partition('@')

if ce[1]=='@' and ce[2][-4::1]=='.com':

d=P.getpass('Enter your password :- ')

e=P.getpass('Re-Enter your password :- ')

while d != e:

print('The password doesn\'t matches with the original password')

e=P.getpass('Re-Enter your password :- ')

t="INSERT INTO keyhole VALUES({},'{}','{}','{}')"

dat=(t.format(s,a,b,e))

kurs.execute(dat)

mydb.commit()

print('Account Details Inserted')

else:

print('Invalid email id')

print('Your email id',b,'is not legal')

print("Your email id should contain an '@' and '.com'")

print("Press '@' to start the function again")

def deleteacc():

a=input('\nEnter your account name :- ')

jck="SELECT \* FROM keyhole WHERE ACCNAME='{}'"

mat=(jck.format(a))

kurs.execute(mat)

ch=kurs.fetchall()

if len(ch) == 0:

print("Account name",a,"doesn't exists")

else:

while True:

b=input('R u sure?[y/n] :- ')

if b == 'y':

t="DELETE FROM keyhole WHERE ACCNAME='{}'"

dat=(t.format(a))

kurs.execute(dat)

mydb.commit()

print('Account is deleted')

break

elif b == 'n':

print('Account is not deleted')

break

else:

print("Invalid input, please enter 'y' or 'n'")

def changepassword():

a=input('\nEnter your account name :- ')

b=input('Enter your email id :- ')

ce=b.partition('@')

qw="SELECT \* FROM keyhole WHERE ACCNAME='{}' AND ACCID='{}'"

nj=qw.format(a,b)

kurs.execute(nj)

teg=kurs.fetchall()

if teg==[] and not(ce[1]=='@' and ce[2][-4::1]=='.com'):

print("\nThe Account name",a,"and the email id",b,"doesn't exists")

if not(ce[1]=='@' and ce[2][-4::1]=='.com'):

print('\nYour email id',b,'is not legal')

print("\nYour email id should contain an '@' and '.com'")

print("\nPress '%' to start the function again")

else:

d=P.getpass('Enter your new password :- ')

e=P.getpass('Re-Enter your new password :- ')

while d != e:

print('The password doesn\'t matches with the original password')

e=P.getpass('Re-Enter your password :- ')

t="UPDATE keyhole SET ACCPASS='{}' WHERE ACCNAME='{}' AND ACCID='{}'"

dat=(t.format(e,a,b))

kurs.execute(dat)

mydb.commit()

print('\nPassword Changed')

def listacc():

kurs.execute("SELECT ACCNO,ACCNAME FROM keyhole ORDER BY ACCNO")

for i in kurs:

print(i[0],i[1])

def firsttime():

try:

with open('profile\_id\_steup.dat','rb+') as f:

g=Z.load(f)

for i in g:

if g[i] == 'exwk':

return True

except FileNotFoundError:

c=dict()

print('Hello User,\nWelcome to Account recorder')

print('This app can manage all your internet acounts')

a=input('\nHave you installed the MySQL app? [y/n] :- ')

if a == 'y':

c['MySQL\_app']=a

c['Name\_first']=input('Your firstname? :- ')

c['Name\_last']=input('Your lastname? :- ')

print('\nCreating the database...')

a=P.getpass('Enter your Password to create your database: ')

mydb=M.connect(host = "localhost", user = "root",passwd=a)

kurs=mydb.cursor()

kurs.execute('CREATE DATABASE padlock')

kurs.execute('USE padlock')

kurs.execute('CREATE TABLE keyhole(ACCNO int(30),ACCNAME varchar(50) Primary key,ACCID varchar(50),ACCPASS varchar(50))')

kurs.close()

c['Database']='exwk'

with open('profile\_id\_steup.dat','wb+') as f:

d=Z.dump(c,f)

print('\nDatabase created\n')

return True

else:

print('\nPlease install MySQL app\n')

T.sleep(5)

print('Closing program...')

T.sleep(5)

return False

t=0

v=firsttime()

if v:

for i in range(3,-1,-1):

try:

a=P.getpass('Enter your Password to connect to your database: ')

mydb=M.connect(host = "localhost", user = "root",passwd=a,database='padlock')

t=1

except:

print('Access Denied')

print('You have',i,'chance to enter your password')

t=0

if t:

break

if not(t):

print('You have exhausted your chances')

T.sleep(5)

print('Restart your program','\nContact your administar')

T.sleep(5)

else:

kurs=mydb.cursor()

print('\nOn the left hand side = You have the functions in which you can operate the software')

print('On the right hand side = You have unique character "key" which can enable you to use the function')

print('You can enter the key in the "Enter command :- " to operate the software')

print('\nView password =','~')

print('View account list =','!')

print('Add account =','@')

print('Delete account =','#')

print('Change password =','%')

print('Exit program =','\*')

while True and t:

v=input('\nEnter command :- ')

if v == '~':

askpassword()

elif v == '!':

print('')

listacc()

elif v == '@':

appendacc()

elif v == '#':

deleteacc()

elif v == '%':

changepassword()

elif v == '\*':

kurs.close()

print('bye')

break

else:

print('\nInvalid Input')

print('Please enter these keywords below')

print('\nView password =','~')

print('View account list =','!')

print('Add account =','@')

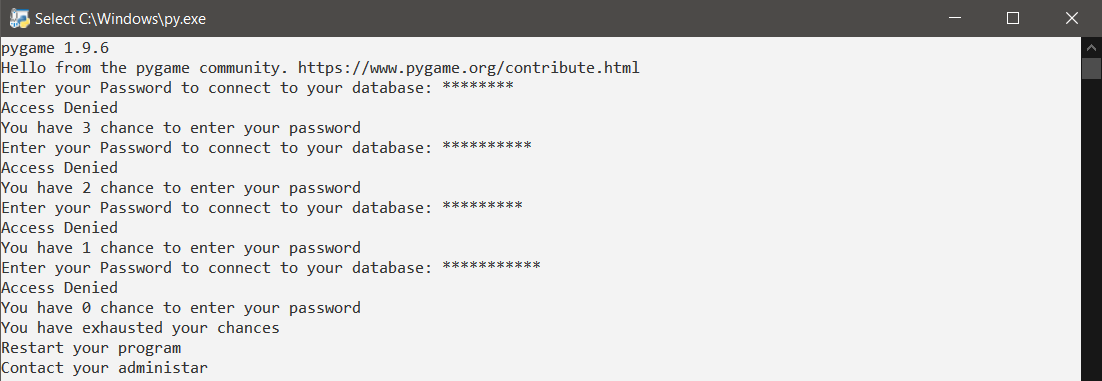
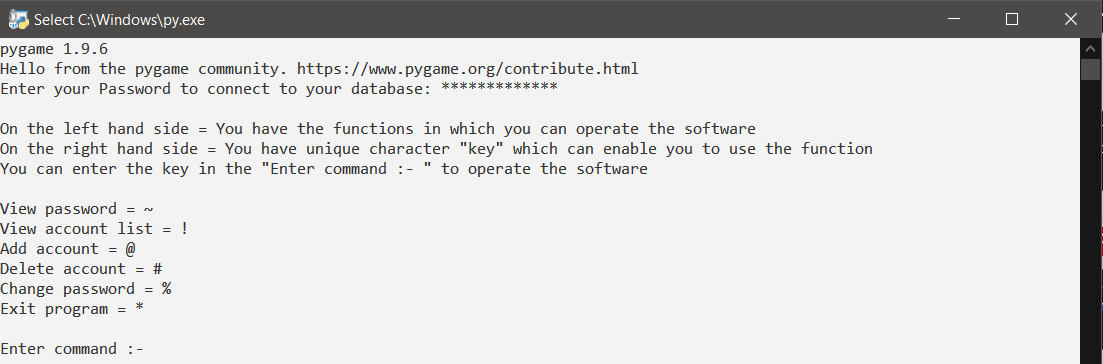
print('Delete account =','#')

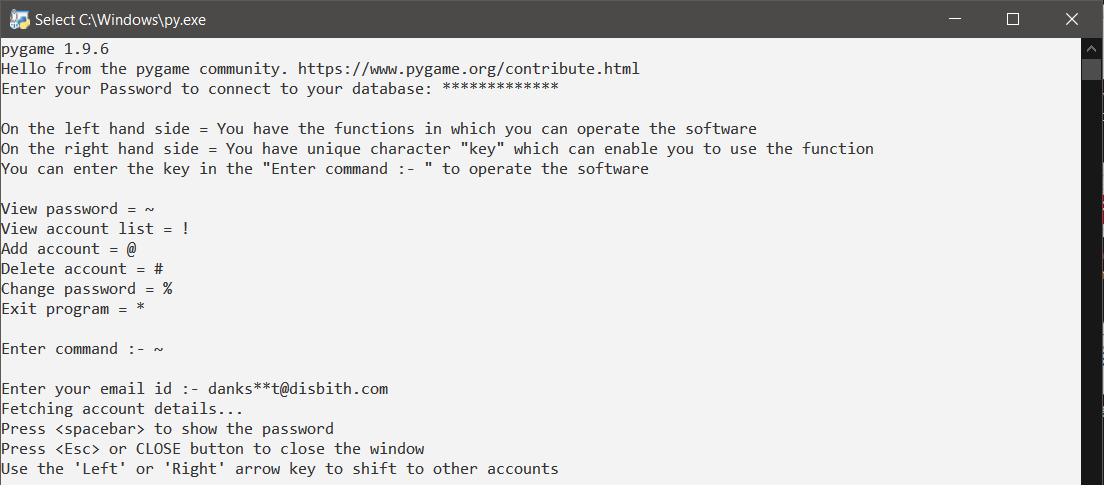
print('Change password =','%')

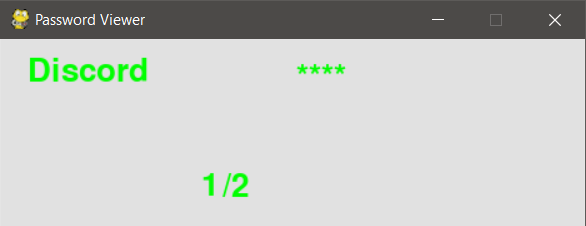
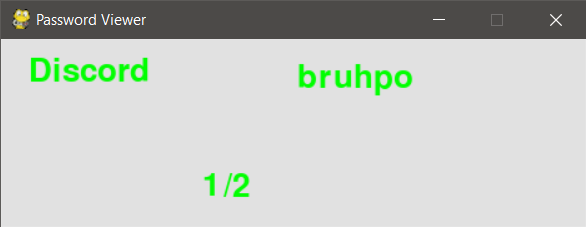
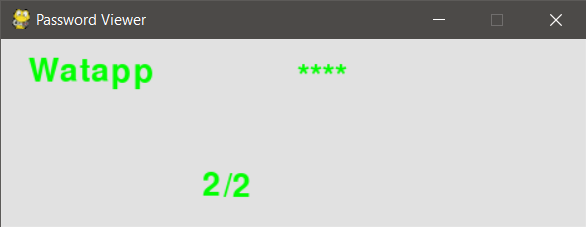
print('Exit program =','\*')

**OUTPUT**

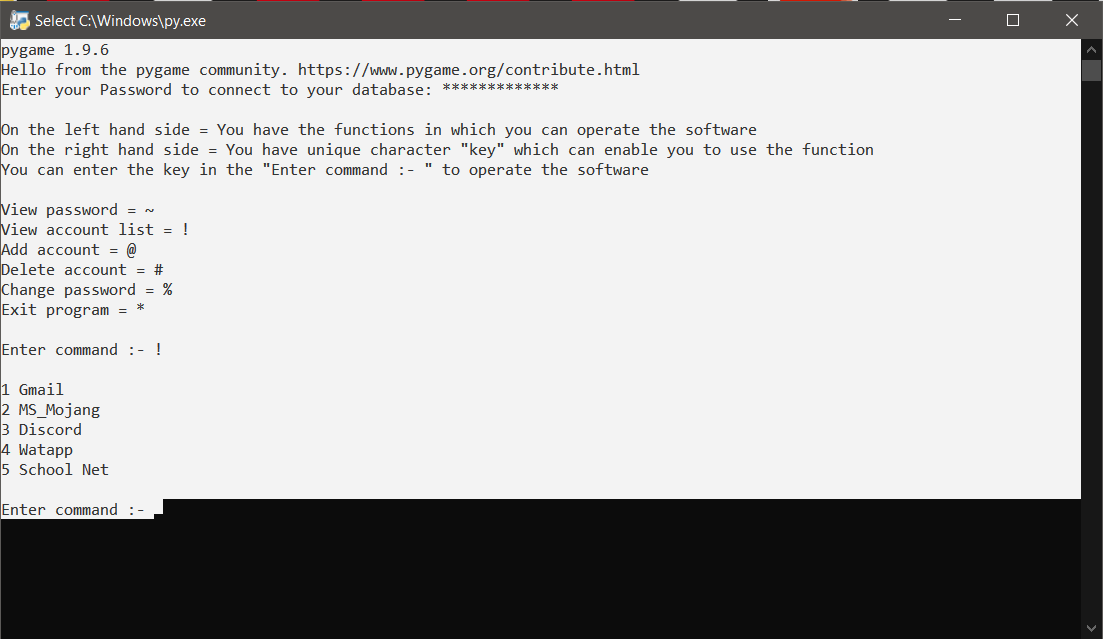
**Login**

If the MySQL password is valid it will give successfully login to the account otherwise the user will receive 3 chances to give the valid password. If the chances are exhausted, then it will automatically close the program and the user needs to open the app again to login.

**View Password**

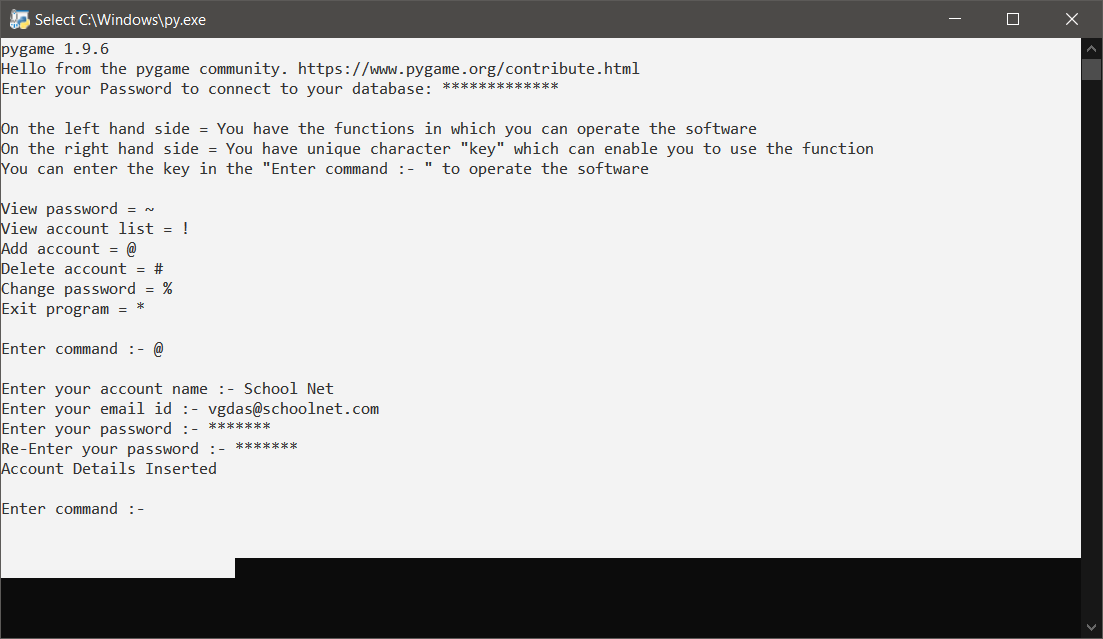


When the user wants to see the password of his respective account, he enters a key ‘~’ . The user needs to enter the respective account id to see the password. Another window will pop up. Pressing spacebar will show the password. If any account uses a common id, then the user can move to its desired account using the left and right arrow keys. You can close the window by pressing esc button or close button.

**View Account List**

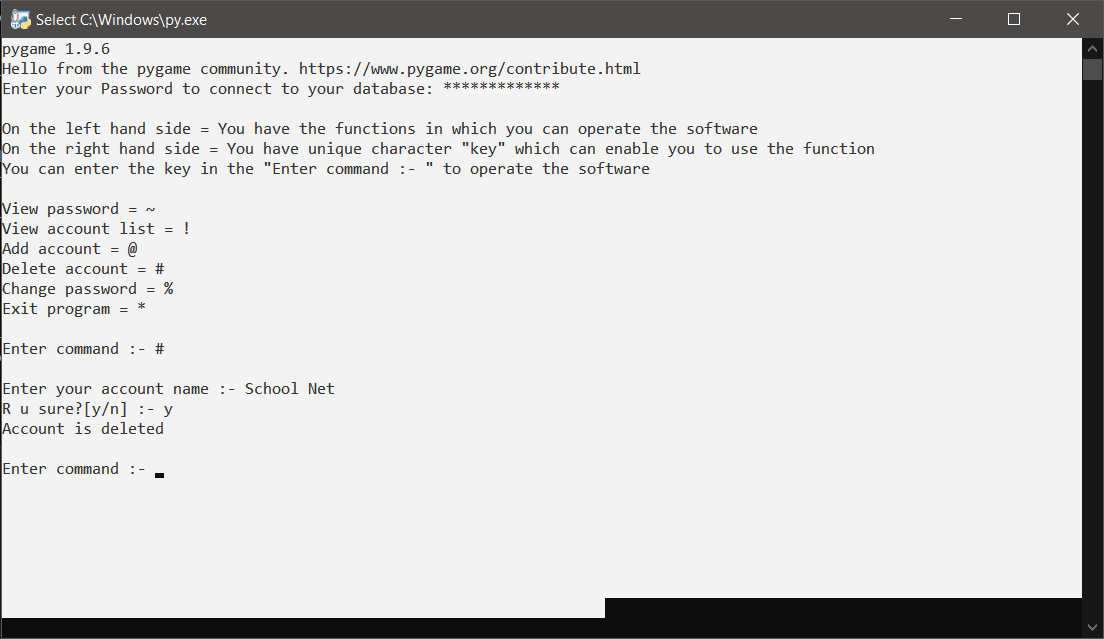
The user can see all the accounts stored in the database using this function by entering ‘!’.

**Add Account**

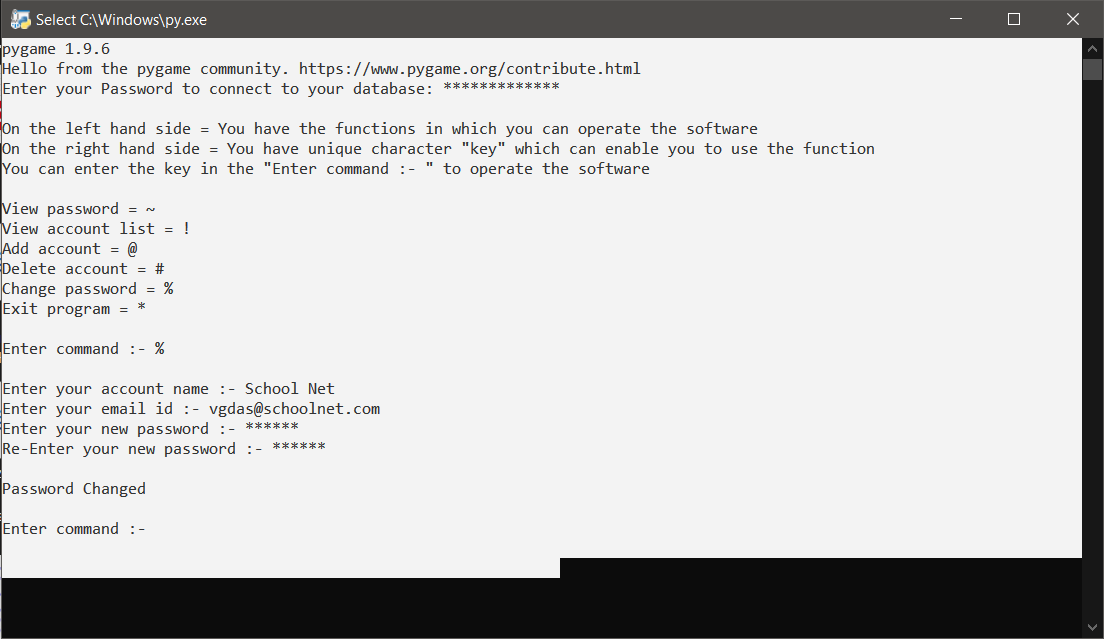
The user can add the account by entering the name of the account, account id and password.

**Delete Account**

The user can delete any account by entering only the account name and confirming the deletion of the account.



**Change Password**

The user can change the password of the account stored in the database by mentioning the account in the software.

BIBLIOGRAPHY

**Book Source**

Computer Science With Python Class 11 -Preeti Arora

Computer Science With Python Class 12 -Preeti Arora

**Internet source**

<https://stackoverflow.com/questions/35805078/how-do-i-convert-a-password-into-asterisks-while-it-is-being-entered>

<https://www.sololearn.com/Discuss/2085381/how-can-i-make-user-input-password-hidden-with-asterisk-in-python>

<https://www.youtube.com/watch?v=opn6_z27HJs>

<https://www.youtube.com/results?search_query=hide+password+in+dots+in+python>

<https://www.youtube.com/watch?v=b4FEI9iqf70>

<https://www.youtube.com/watch?v=s4Iw_n0jdBc>

<https://www.youtube.com/results?search_query=enter+password+in+python>

<https://www.geeksforgeeks.org/getpass-and-getuser-in-python-password-without-echo/>

<https://stackoverflow.com/questions/157938/hiding-a-password-in-a-python-script-insecure-obfuscation-only>

<https://www.geeksforgeeks.org/python-display-text-to-pygame-window/>

<https://www.pygame.org/docs/ref/font.html>