A Project Report

Of

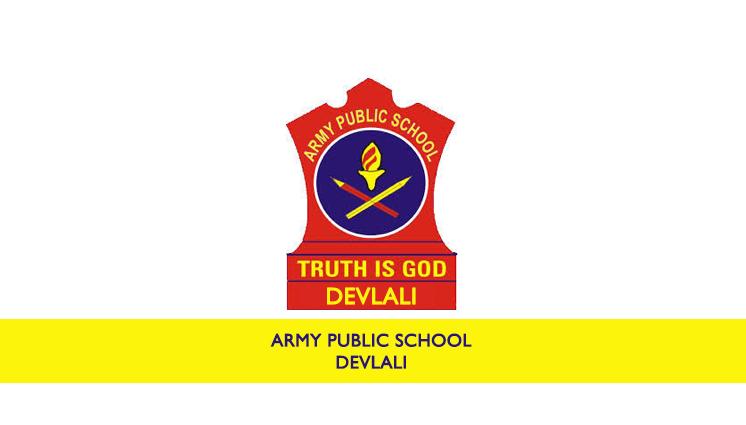
Informatics Practices

On

**“CD Library Management System”**

****

**Saiesh Agre**

****

**CERTIFICATE**

This is to certify that Master Saiesh Agre of class XII Science Of Army Public School, Devlali has successfully completed his Informatics Project on “CD Library Management System’ under the guidance of Mrs. Archana Patil for academic year 2020-21.

*Internal Examiner Principal External Examiner*

*Signature Signature Signature*

**ACKNOWLEDGEMENT**

I would like to give my sincere gratitude and appreciation to our principal **Mrs. Medha Gode** for giving me this opportunity.

A special thanks to our Informatics Practice teacher, **Mrs. Archana Patil**, who has given her efforts in guiding us and understanding our project as well as her encouragement to maintain our progress and also for helping, stimulating suggestions and encouraging, coordinating my project and successfully concluding it as this report.

**INDEX**

| **Sr. No.** | **Content** | **Page No.** |
| --- | --- | --- |
| **1** | **Description** | 5 |
| **2** | **Technical Specifications**:  **2.1** *-* Hardware Requirements  **2.2** - Software Requirements | 6 |
| **3** | **Screen Layouts and Coding**  **3.1 -**  Databases and tables used.  **3.2** - Modules imported and connectivity statements.  **3.3** - Python Coding | **7 – 68**  8-10  11  12-68 |
| **4** | **Bibliography**  **4.1** - Reference Books and Websites | 69 |

**Project Description**

1. **User Characteristics**

The users are categorized as:

* **Admin or Manager**

Authorized to update, alter, change or modify Staff and Customer tables, and responsible for System Maintenance.

* **Staff**

Authorized to update, alter, change, or modify Movies, Music, Games, and Software tables. Do not have access to Staff’s or Customer’s personal details.

* **Client**

Authorized to access CD’s collection, and Buy them.

1. **Software features**

* Easy to use, GUI interface.
* Anytime access to collections.
* Excellent data management, data reliability and accuracy.

**Technical Specifications**

* **Minimum Hardware Requirements**
* Intel Core i3 Processor
* CPU T6670 @ 2.20 GHz
* Color CRT/TFT 15.5” Monitor
* 64-bit Operating System
* 4.00 GB Installed Memory (RAM)
* 500 GB DDR3/DDR4 Hard Disk
* **Minimum Software Requirements**
* Python IDLE 3.5 or higher
* MySQL Server 5.0 or higher
* Windows 7 or higher or MacOS
* **Additional Modules Required**
* PyQt5 and Qt5designer
* Twilio

**Screen**

**Layouts**

**And**

**Coding**

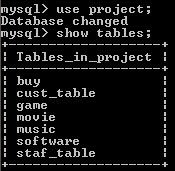
**Database**

**And**

**Tables**

**Used**

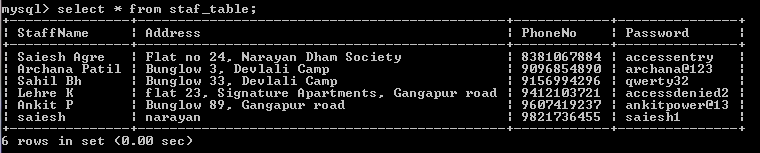
* **Database**

****

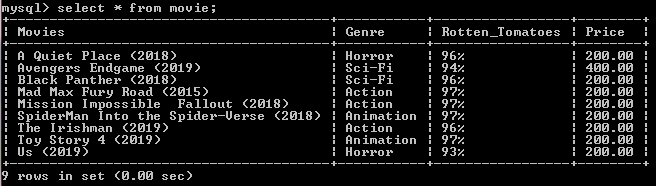
* **Customer table**



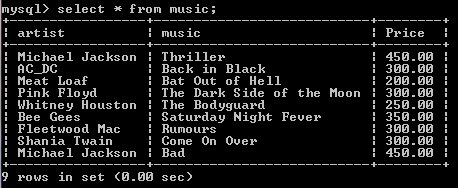
* **Staff table**

****

* **Movies table**

****

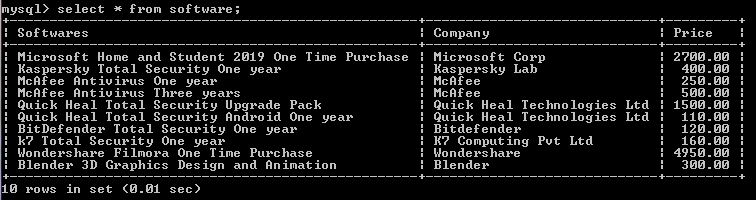
* **Music Records table**

****

* **Games table**

****

* **Software table**

****

**Modules imported**

**and**

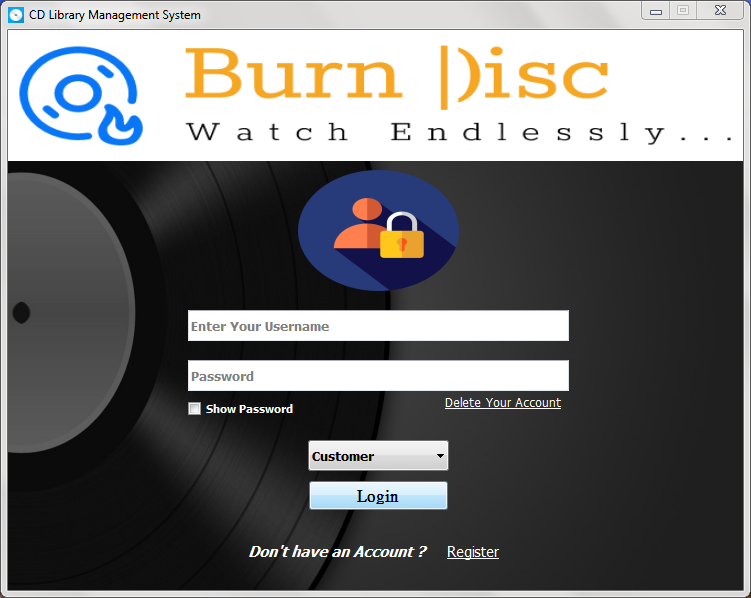
**Connectivity statements**

* **Connectivity Statements**
* import mysql.connector as sql
* **Modules imported**
* from PyQt5 import QtCore, QtGui, QtWidgets
* import subprocess
* import tkinter as Tk
* from tkinter import messagebox
* import random
* from twilio.rest import Client
* import sys

**Python**

**Coding**

**Main Login Window**

****

# Importing Modules

**from** PyQt5 **import** QtCore**,** QtGui**,** QtWidgets

**import** subprocess

**import** mysql**.**connector **as** sql

**import** tkinter **as** Tk

**from** tkinter **import** messagebox

#========================================#

root **=** Tk**.**Tk**()** # For hiding tkinter root window

root**.**withdraw**()**

# Connecting to MYSQL and Creating Required Tables

con**=**sql**.**connect**(**host**=**"localhost"**,** user**=**"root"**,** password**=**"saieshagre"**)**

cur**=**con**.**cursor**(**buffered**=True)**

cur**.**execute**(**" CREATE DATABASE IF NOT exists project;"**)**

cur**.**execute**(**"USE project;"**)**

# ---- Customer Table ----

cur**.**execute**(**'''CREATE table IF NOT exists cust\_table(Username varchar(20) NOT

NULL, Address varchar(80) NOT NULL,

PhoneNo BIGINT(10) NOT NULL PRIMARY KEY,

Password varchar(20) NOT NULL UNIQUE KEY);'''**)**

# ---- Staff Table ----

cur**.**execute**(**'''CREATE table IF NOT exists staf\_table(StaffName varchar(20) NOT

NULL, Address varchar(80) NOT NULL,

PhoneNo BIGINT(10) NOT NULL UNIQUE KEY,

Password varchar(30) NOT NULL UNIQUE KEY);'''**)**

# ---- Movies Table ----

cur**.**execute**(**'''CREATE table IF NOT exists Movie(Movies varchar(40) NOT NULL

UNIQUE, Genre varchar(10),

Rotten\_Tomatoes varchar(5),

Price float(5,2));'''**)**

# ---- Music Table ----

cur**.**execute**(**'''CREATE table IF NOT exists Music(Artist varchar(40),

Music varchar(30),

Price float(5,2));'''**)**

# ---- Game Table ----

cur**.**execute**(**'''CREATE table IF NOT exists Game(Games varchar(40),

Developer varchar(30),

Price float(7,2));'''**)**

# ---- Software Table ----

cur**.**execute**(**'''CREATE table IF NOT exists Software(Softwares varchar(50),

Company varchar(40), Price float(7,2));'''**)**

# ---- Temporary Buy List ----

cur**.**execute**(**'''CREATE table IF NOT exists Buy(Items varchar(50),

Price float(7,2));'''**)**

# ---- Deleting Buy List items from previous user ----

cur**.**execute**(**"Delete from Buy;"**)**

con**.**commit**()**

# Graphical User Interface

**class** **Ui\_LoginPg(object):**

# Opening Registration Window

**def** Register**(**self**):**

subprocess**.**call**(**'Windows\Register.py'**,** shell**=True)**

# Login Confirmation

**def** Login**(**self**):**

User **=** (self**.**username**.**text**()).**strip()

key **=** (self**.**password**.**text**()).**strip()

Role **=** **str(**self**.**role**.**currentText**())**

**if** Role **==** "Customer" **:**

query**=**"select \* from cust\_table where Username=%s and Password=%s"

data**=**cur**.**execute**(**query**,(**User**,** key**))**

count**=**cur**.**rowcount

**if** count**==**1**:**

LoginPg**.**hide**()**

subprocess**.**call**(**'Windows\\User\_win.py'**,** shell**=True)**

LoginPg**.**close**()**

**else:**

messagebox**.**showinfo**(**"FAILED"**,** "Failed To Log In, Check Your

Credentials..."**)**

**elif** Role **==** "Staff"**:**

query**=**"select \* from staf\_table where StaffName=%s and Password=%s"

data**=**cur**.**execute**(**query**,(**User**,** key**))**

count**=**cur**.**rowcount

**if** count**==**1**:**

LoginPg**.**hide**()**

subprocess**.**call**(**'Windows\\Staff\_win.py'**,** shell**=True)**

LoginPg**.**close**()**

**else:**

messagebox**.**showinfo**(**"FAILED"**,** "Failed To Log In, Check Your

Credentials..."**)**

# Removing Users From Database

**def** del\_user**(**self**):**

name **=** self**.**username**.**text**()**

passw **=** self**.**password**.**text**()**

query1**=**"select \* from cust\_table where Username=%s and Password=%s"

data1**=**cur**.**execute**(**query1**,(**name**,** passw**))**

count1**=**cur**.**rowcount

**if** count1**==**0**:**

messagebox**.**showinfo**(**"Error 404"**,**"Your Account doesn't exists. \n

Check Your Credentials.."**)**

**else:**

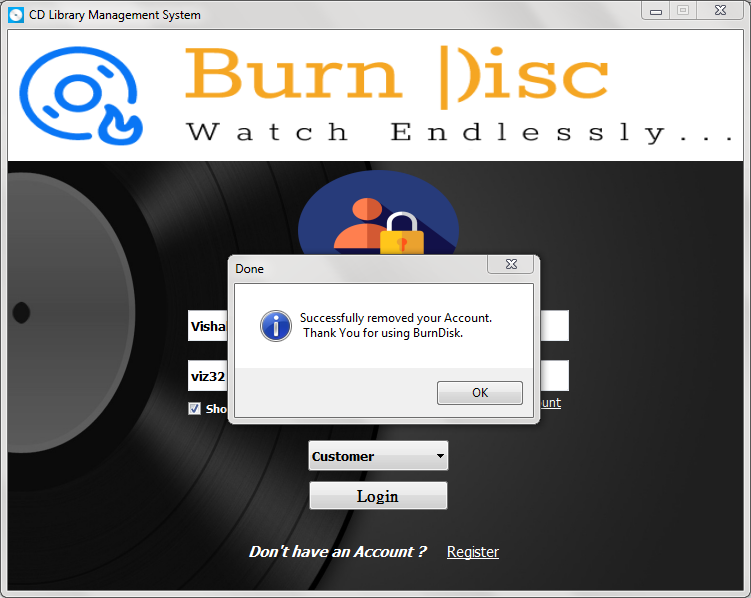
query2**=**"DELETE from cust\_table where Username=%s and Password=%s"

data2**=**cur**.**execute**(**query2**,(**name**,** passw**))**

con**.**commit**()**

messagebox**.**showinfo**(**"Done"**,**"Successfully removed your Account. \n

Thank You for using BurnDisk."**)**



# Hiding Delete Button if User is a Staff Member

**def** hide\_btns**(**self**):**

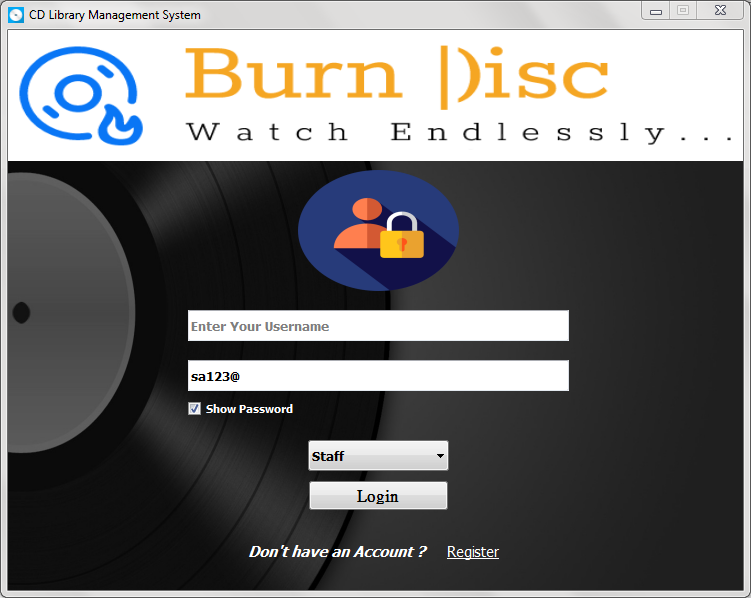
Role **=** **str(**self**.**role**.**currentText**())**

**if** Role **==** "Customer"**:**

self**.**delete\_1**.**show**()**

**else:**

self**.**delete\_1**.**hide**()**



# Showing / Hiding Password

**def** hide**(**self**):**

**if** self**.**checkBox**.**isChecked**():**

self**.**password**.**setEchoMode**(**QtWidgets**.**QLineEdit**.**Normal**)**

**else:**

self**.**password**.**setEchoMode**(**QtWidgets**.**QLineEdit**.**Password**)**

pass1.PNG

pass2.PNG

**def** setupUi**(**self**,** LoginPg**):**

LoginPg**.**setObjectName**(**"LoginPg"**)**

LoginPg**.**setFixedSize**(**735**,** 560**)**

sizePolicy **=** QtWidgets**.**QSizePolicy**(**QtWidgets**.**QSizePolicy**.**Fixed**,** \

QtWidgets**.**QSizePolicy**.**Fixed**)**

sizePolicy**.**setHorizontalStretch**(**0**)**

sizePolicy**.**setVerticalStretch**(**0**)**

sizePolicy**.**setHeightForWidth**(**LoginPg**.**sizePolicy**().**hasHeightForWidth**())**

LoginPg**.**setSizePolicy**(**sizePolicy**)**

icon **=** QtGui**.**QIcon**()**

icon**.**addPixmap**(**QtGui**.**QPixmap**(**"Icons/Mw.png"**),** QtGui**.**QIcon**.**Normal**,\**

QtGui**.**QIcon**.**Off**)**

LoginPg**.**setWindowIcon**(**icon**)**

self**.**centralwidget **=** QtWidgets**.**QWidget**(**LoginPg**)**

self**.**centralwidget**.**setObjectName**(**"centralwidget"**)**

self**.**Bg **=** QtWidgets**.**QLabel**(**self**.**centralwidget**)**

self**.**Bg**.**setGeometry**(**QtCore**.**QRect**(-**5**,** **-**1**,** 741**,** 571**))**

self**.**Bg**.**setAutoFillBackground**(False)**

self**.**Bg**.**setStyleSheet**(**"color: rgb(255, 255, 255);"**)**

self**.**Bg**.**setText**(**""**)**

self**.**Bg**.**setPixmap**(**QtGui**.**QPixmap**(**"Icons/Bg.png"**))**

self**.**Bg**.**setScaledContents**(True)**

self**.**Bg**.**setObjectName**(**"Bg"**)**

self**.**logo1 **=** QtWidgets**.**QLabel**(**self**.**centralwidget**)**

self**.**logo1**.**setGeometry**(**QtCore**.**QRect**(-**4**,** 0**,** 741**,** 131**))**

self**.**logo1**.**setText**(**""**)**

self**.**logo1**.**setPixmap**(**QtGui**.**QPixmap**(**"Icons/Main.png"**))**

self**.**logo1**.**setScaledContents**(True)**

self**.**logo1**.**setObjectName**(**"logo1"**)**

self**.**logo2 **=** QtWidgets**.**QLabel**(**self**.**centralwidget**)**

self**.**logo2**.**setGeometry**(**QtCore**.**QRect**(**290**,** 140**,** 161**,** 121**))**

self**.**logo2**.**setText**(**""**)**

self**.**logo2**.**setPixmap**(**QtGui**.**QPixmap**(**"Icons/Log.png"**))**

self**.**logo2**.**setScaledContents**(True)**

self**.**logo2**.**setObjectName**(**"logo2"**)**

self**.**username **=** QtWidgets**.**QLineEdit**(**self**.**centralwidget**)**

self**.**username**.**setGeometry**(**QtCore**.**QRect**(**180**,** 280**,** 381**,** 31**))**

font **=** QtGui**.**QFont**()**

font**.**setPointSize**(**10**)**

font**.**setBold**(True)**

font**.**setWeight**(**75**)**

self**.**username**.**setFont**(**font**)**

self**.**username**.**setFocusPolicy**(**QtCore**.**Qt**.**ClickFocus**)**

self**.**username**.**setAcceptDrops**(True)**

self**.**username**.**setToolTip**(**""**)**

self**.**username**.**setStatusTip**(**""**)**

self**.**username**.**setStyleSheet**(**""**)**

self**.**username**.**setObjectName**(**"username"**)**

self**.**password **=** QtWidgets**.**QLineEdit**(**self**.**centralwidget**)**

self**.**password**.**setGeometry**(**QtCore**.**QRect**(**180**,** 330**,** 381**,** 31**))**

font **=** QtGui**.**QFont**()**

font**.**setPointSize**(**10**)**

font**.**setBold**(True)**

font**.**setItalic**(False)**

font**.**setWeight**(**75**)**

self**.**password**.**setFont**(**font**)**

self**.**password**.**setFocusPolicy**(**QtCore**.**Qt**.**ClickFocus**)**

self**.**password**.**setEchoMode**(**QtWidgets**.**QLineEdit**.**Password**)**

self**.**password**.**setObjectName**(**"password"**)**

self**.**label **=** QtWidgets**.**QLabel**(**self**.**centralwidget**)**

self**.**label**.**setGeometry**(**QtCore**.**QRect**(**240**,** 510**,** 191**,** 21**))**

font **=** QtGui**.**QFont**()**

font**.**setPointSize**(**11**)**

font**.**setBold**(True)**

font**.**setItalic**(True)**

font**.**setWeight**(**75**)**

self**.**label**.**setFont**(**font**)**

self**.**label**.**setMouseTracking**(False)**

self**.**label**.**setStyleSheet**(**"color: rgb(255, 255, 255);"**)**

self**.**label**.**setScaledContents**(True)**

self**.**label**.**setObjectName**(**"label"**)**

self**.**register\_btn **=** QtWidgets**.**QPushButton**(**self**.**centralwidget**)**

self**.**register\_btn**.**setGeometry**(**QtCore**.**QRect**(**420**,** 500**,** 91**,** 41**))**

self**.**register\_btn**.**clicked**.**connect**(**self**.**Register**)** # ---- Click Signal

font **=** QtGui**.**QFont**()**

font**.**setPointSize**(**11**)**

font**.**setBold**(False)**

font**.**setItalic**(False)**

font**.**setUnderline**(True)**

font**.**setWeight**(**50**)**

self**.**register\_btn**.**setFont**(**font**)**

self**.**register\_btn**.**setCursor**(**QtGui**.**QCursor**(**QtCore**.**Qt**.**PointingHandCursor**))**

self**.**register\_btn**.**setFocusPolicy**(**QtCore**.**Qt**.**ClickFocus**)**

self**.**register\_btn**.**setAutoFillBackground**(False)**

self**.**register\_btn**.**setStyleSheet**(**"color: rgb(255, 255, 255);"**)**

self**.**register\_btn**.**setCheckable**(False)**

self**.**register\_btn**.**setAutoDefault**(False)**

self**.**register\_btn**.**setDefault**(False)**

self**.**register\_btn**.**setFlat**(True)**

self**.**register\_btn**.**setObjectName**(**"register\_btn"**)**

self**.**login\_btn **=** QtWidgets**.**QPushButton**(**self**.**centralwidget**)**

self**.**login\_btn**.**setGeometry**(**QtCore**.**QRect**(**300**,** 450**,** 141**,** 31**))**

self**.**login\_btn**.**clicked**.**connect**(**self**.**Login**)** # ---- Click Signal

font **=** QtGui**.**QFont**()**

font**.**setFamily**(**"PMingLiU-ExtB"**)**

font**.**setPointSize**(**12**)**

font**.**setBold**(True)**

font**.**setItalic**(False)**

font**.**setWeight**(**75**)**

self**.**login\_btn**.**setFont**(**font**)**

self**.**login\_btn**.**setCursor**(**QtGui**.**QCursor**(**QtCore**.**Qt**.**PointingHandCursor**))**

self**.**login\_btn**.**setFocusPolicy**(**QtCore**.**Qt**.**ClickFocus**)**

self**.**login\_btn**.**setObjectName**(**"login\_btn"**)**

self**.**role **=** QtWidgets**.**QComboBox**(**self**.**centralwidget**)**

self**.**role**.**setGeometry**(**QtCore**.**QRect**(**300**,** 410**,** 141**,** 31**))**

self**.**role**.**activated**.**connect**(**self**.**hide\_btns**)**

font **=** QtGui**.**QFont**()**

font**.**setBold**(True)**

font**.**setItalic**(False)**

font**.**setUnderline**(False)**

font**.**setPointSize**(**10**)**

font**.**setWeight**(**75**)**

self**.**role**.**setFont**(**font**)**

self**.**role**.**setCursor**(**QtGui**.**QCursor**(**QtCore**.**Qt**.**PointingHandCursor**))**

self**.**role**.**setAutoFillBackground**(False)**

self**.**role**.**setFrame**(True)**

self**.**role**.**setObjectName**(**"role"**)**

self**.**role**.**addItem**(**""**)**

self**.**role**.**addItem**(**""**)**

self**.**checkBox **=** QtWidgets**.**QCheckBox**(**self**.**centralwidget**)**

self**.**checkBox**.**setGeometry**(**QtCore**.**QRect**(**180**,** 370**,** 111**,** 17**))**

font **=** QtGui**.**QFont**()**

font**.**setBold**(True)**

font**.**setWeight**(**75**)**

self**.**checkBox**.**setFont**(**font**)**

self**.**checkBox**.**setCursor**(**QtGui**.**QCursor**(**QtCore**.**Qt**.**PointingHandCursor**))**

self**.**checkBox**.**stateChanged**.**connect**(**self**.**hide**)**

self**.**checkBox**.**setFocusPolicy**(**QtCore**.**Qt**.**ClickFocus**)**

self**.**checkBox**.**setStyleSheet**(**"color: rgb(255, 255, 255);"**)**

self**.**checkBox**.**setObjectName**(**"checkBox"**)**

self**.**delete\_1 **=** QtWidgets**.**QPushButton**(**self**.**centralwidget**)**

self**.**delete\_1**.**setGeometry**(**QtCore**.**QRect**(**430**,** 360**,** 131**,** 23**))**

self**.**delete\_1**.**clicked**.**connect**(**self**.**del\_user**)** # ---- Click Signal

font **=** QtGui**.**QFont**()**

font**.**setPointSize**(**9**)**

font**.**setUnderline**(True)**

self**.**delete\_1**.**setFont**(**font**)**

self**.**delete\_1**.**setFocusPolicy**(**QtCore**.**Qt**.**ClickFocus**)**

self**.**delete\_1**.**setStyleSheet**(**"color: rgb(255, 255, 255);"**)**

self**.**delete\_1**.**setFlat**(True)**

self**.**delete\_1**.**setObjectName**(**"delete\_1"**)**

LoginPg**.**setCentralWidget**(**self**.**centralwidget**)**

self**.**retranslateUi**(**LoginPg**)**

QtCore**.**QMetaObject**.**connectSlotsByName**(**LoginPg**)**

**def** retranslateUi**(**self**,** LoginPg**):**

\_translate **=** QtCore**.**QCoreApplication**.**translate

LoginPg**.**setWindowTitle**(**\_translate**(**"LoginPg"**,** "CD Library Management

System"**))**

self**.**username**.**setPlaceholderText**(**\_translate**(**"LoginPg"**,** "Enter Your

Username"**))**

self**.**password**.**setPlaceholderText**(**\_translate**(**"LoginPg"**,** "Password"**))**

self**.**label**.**setText**(**\_translate**(**"LoginPg"**,** "Don\'t have an Account ?"**))**

self**.**register\_btn**.**setWhatsThis**(**\_translate**(**"LoginPg"**,** "Sign Up Now!!!"**))**

self**.**register\_btn**.**setText**(**\_translate**(**"LoginPg"**,** "Register"**))**

self**.**login\_btn**.**setText**(**\_translate**(**"LoginPg"**,** "Login"**))**

self**.**role**.**setItemText**(**0**,** \_translate**(**"LoginPg"**,** "Customer"**))**

self**.**role**.**setItemText**(**1**,** \_translate**(**"LoginPg"**,** "Staff"**))**

self**.**checkBox**.**setText**(**\_translate**(**"LoginPg"**,** "Show Password"**))**

self**.**delete\_1**.**setText**(**\_translate**(**"LoginPg"**,** "Delete Your Account"**))**

**if** \_\_name\_\_ **==** "\_\_main\_\_"**:**

**import** sys

app **=** QtWidgets**.**QApplication**(**sys**.**argv**)**

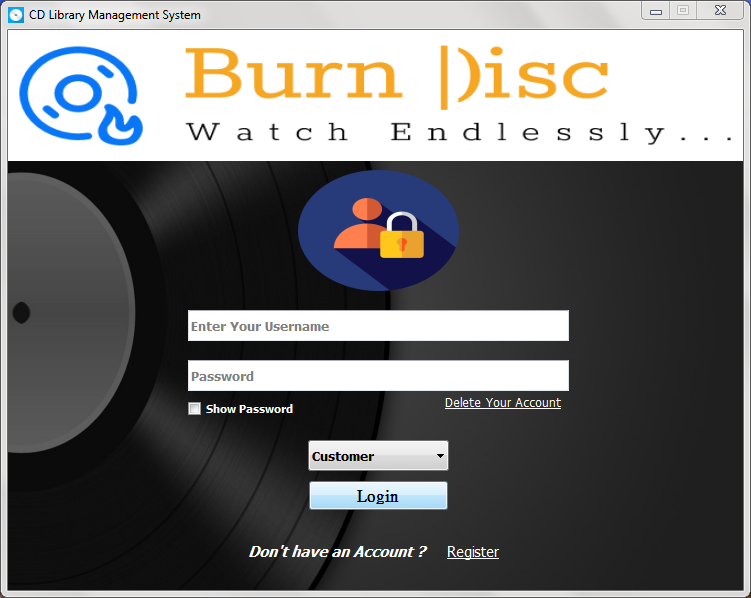
LoginPg **=** QtWidgets**.**QMainWindow**()**

ui **=** Ui\_LoginPg**()**

ui**.**setupUi**(**LoginPg**)**

LoginPg**.**show**()**

sys**.exit(**app**.**exec\_**())**

****

**Registration Window**

****

# Importing Modules and Connecting to MYSQL

**from** PyQt5 **import** QtCore**,** QtGui**,** QtWidgets

**import** random

**from** twilio**.**rest **import** Client

**from** tkinter **import** messagebox

**import** tkinter **as** Tk

**import** mysql**.**connector **as** sql

**import** sys

root **=** Tk**.**Tk**()** # For hiding tkinter root window

root**.**withdraw**()**

# Python Mysql Connection

con**=**sql**.**connect**(**host**=**"localhost"**,** user**=**"root"**,** password**=**"saieshagre"**)**

cur**=**con**.**cursor**(**buffered**=True)**

cur**.**execute**(**"USE project"**)**

# Graphical User Interface

**class** **Ui\_Register(object):**

# One Time Password

**def** send**(**self**):**

# Checking if any field are left empty

name **=** **str(**self**.**user**.**text**())**

key **=** **str(**self**.**password**.**text**())**

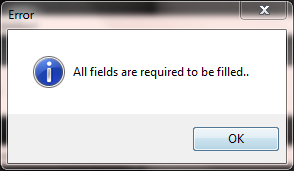
add **=** **str(**self**.**address**.**text**())**

phn **=** **str(**self**.**phno**.**text**())**

**if** **len(**name **or** key **or** add **or** phn**)==**0**:**

messagebox**.**showinfo**(**"Error"**,**"All fields are required to be filled.."**)**

**return**



**else:**

**pass**

# Send 6 digits OTP

digits**=**"abcdefghijklmnopqrstuvwxyz0123456789"

self**.**OTP**=**""

**for** i **in** **range(**6**):**

a**=**random**.**randint**(**0**,len(**digits**))**

self**.**OTP**+=**digits**[**a**]**

PHNO**=**"+91"

PHNO**+=str(**self**.**phno**.**text**())**

ac\_sid**=**"ACe71f39874ed8990209c17a5c4795176a"

ath\_token**=**"4dd05774482007a30b98f2ca4112798a"

client **=** Client**(**ac\_sid**,** ath\_token**)**

**try:**

message **=** client**.**messages \

**.**create**(**

body**=**"Your One Time Password for BurnDisk Registration is \

"**+**self**.**OTP**,**

from\_**=**'+13123130823'**,**

to**=**PHNO

**)**

self**.**OK**.**setEnabled**(True)** # Enabling Button

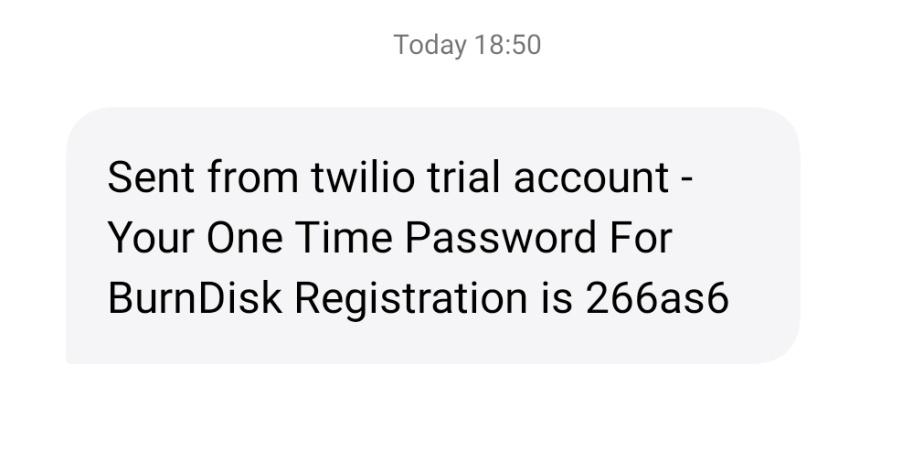
**except:**

messagebox**.**showinfo**(**"Error"**,**"Phone Number provided is not valid \n

Try again / Or Call Us : +91 9881527889"**)**

# Enabling Button

self**.**OK**.**setEnabled**(True)**



# Validating OTP

**def** check**(**self**):**

**if** **str(**self**.**otp\_fill**.**text**())==**self**.**OTP**:**

self**.**regis\_btn**.**setEnabled**(True)**

**else:**

messagebox**.**showinfo**(**"Oops !!"**,** "OTP is Wrong.. \n Try Again"**)**

# Adding New Customer in Database

**def** add\_data**(**self**):**

name **=** **str(**self**.**user**.**text**())**

key **=** **str(**self**.**password**.**text**())**

add **=** **str(**self**.**address**.**text**())**

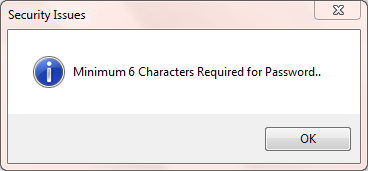
phn **=** **int(**self**.**phno**.**text**())**

# Checking the length of Password (Min 6 Characters)

**if** **len(**key**)<=**5**:**

messagebox**.**showinfo**(**"Security Issues"**,** "Minimum 6 Characters \

Required for Password.."**)**



**else:**

**pass**

# Checking if Password or Phone Number exists already

query **=** "select \* from cust\_table where PhoneNo=%s or Password=%s"

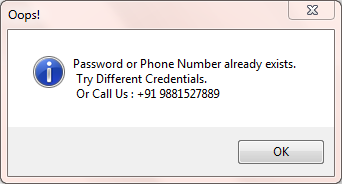
cur**.**execute**(**query**,(**name**,** key**))**

count **=** cur**.**rowcount**()**

**if** count **>=** 1**:**

messagebox**.**showinfo**(**"Oops!"**,**"Password or Phone Number already \

exists.\n Try Different Credentials. \n Or Call Us : +91 9881527889"**)**



**else:**

**pass**

# Adding Customer after confirmation of detail

query**=**"insert into cust\_table(Username, Password, Address, PhoneNo)

values(%s, %s, %s, %s)"

cur**.**execute**(**query**,(**name**,** key**,** add**,** phn**))**

con**.**commit**()**

query **=** "select \* from cust\_table where Username=%s and Password=%s"

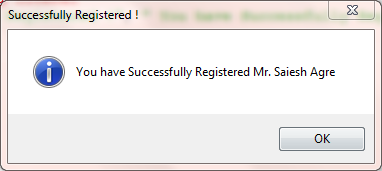
cur**.**execute**(**query**,(**name**,** key**))**

count **=** cur**.**rowcount**()**

**if** count**==**1**:**

messagebox**.**showinfo**(**"Successfully Registered !"**,** " You have \

Successfully Registered Mr. "**+**name**)**



**else:**

messagebox**.**showinfo**(**"Error"**,** "Looks like something's wrong try again

\n OR Call Us : +91 9881527889"**)**

# Showing / Hiding Password

**def** hide\_pass**(**self**):**

**if** self**.**checkBox**.**isChecked**():**

self**.**password**.**setEchoMode**(**QtWidgets**.**QLineEdit**.**Normal**)**

**else:**

self**.**password**.**setEchoMode**(**QtWidgets**.**QLineEdit**.**Password**)**

**pass1.PNG**

pass2.PNG

**def** setupUi**(**self**,** Register**):**

Register**.**setObjectName**(**"Register"**)**

Register**.**setWindowModality**(**QtCore**.**Qt**.**WindowModal**)**

Register**.**setFixedSize**(**735**,** 560**)**

icon **=** QtGui**.**QIcon**()**

icon**.**addPixmap**(**QtGui**.**QPixmap**(**"Icons/Mw.png"**),** QtGui**.**QIcon**.**Normal**,** \

QtGui**.**QIcon**.**Off**)**

Register**.**setWindowIcon**(**icon**)**

self**.**centralwidget **=** QtWidgets**.**QWidget**(**Register**)**

self**.**centralwidget**.**setObjectName**(**"centralwidget"**)**

self**.**label **=** QtWidgets**.**QLabel**(**self**.**centralwidget**)**

self**.**label**.**setGeometry**(**QtCore**.**QRect**(-**4**,** **-**8**,** 741**,** 571**))**

font **=** QtGui**.**QFont**()**

font**.**setPointSize**(**15**)**

self**.**label**.**setFont**(**font**)**

self**.**label**.**setFrameShape**(**QtWidgets**.**QFrame**.**Box**)**

self**.**label**.**setText**(**""**)**

self**.**label**.**setPixmap**(**QtGui**.**QPixmap**(**"Icons/Bg.png"**))**

self**.**label**.**setScaledContents**(True)**

self**.**label**.**setObjectName**(**"label"**)**

self**.**label\_2 **=** QtWidgets**.**QLabel**(**self**.**centralwidget**)**

self**.**label\_2**.**setGeometry**(**QtCore**.**QRect**(**251**,** 20**,** 91**,** 81**))**

self**.**label\_2**.**setFrameShape**(**QtWidgets**.**QFrame**.**NoFrame**)**

self**.**label\_2**.**setText**(**""**)**

self**.**label\_2**.**setPixmap**(**QtGui**.**QPixmap**(**"Icons/Regis.png"**))**

self**.**label\_2**.**setScaledContents**(True)**

self**.**label\_2**.**setObjectName**(**"label\_2"**)**

self**.**label\_3 **=** QtWidgets**.**QLabel**(**self**.**centralwidget**)**

self**.**label\_3**.**setGeometry**(**QtCore**.**QRect**(**341**,** 20**,** 141**,** 81**))**

font **=** QtGui**.**QFont**()**

font**.**setFamily**(**"Agency FB"**)**

font**.**setPointSize**(**23**)**

font**.**setBold**(True)**

font**.**setItalic**(False)**

font**.**setUnderline**(False)**

font**.**setWeight**(**75**)**

self**.**label\_3**.**setFont**(**font**)**

self**.**label\_3**.**setStyleSheet**(**"background-color: rgb(255, 255, 255);"**)**

self**.**label\_3**.**setFrameShape**(**QtWidgets**.**QFrame**.**NoFrame**)**

self**.**label\_3**.**setObjectName**(**"label\_3"**)**

self**.**frame **=** QtWidgets**.**QFrame**(**self**.**centralwidget**)**

self**.**frame**.**setGeometry**(**QtCore**.**QRect**(**49**,** 119**,** 641**,** 421**))**

self**.**frame**.**setFrameShape**(**QtWidgets**.**QFrame**.**Box**)**

self**.**frame**.**setFrameShadow**(**QtWidgets**.**QFrame**.**Raised**)**

self**.**frame**.**setObjectName**(**"frame"**)**

self**.**user **=** QtWidgets**.**QLineEdit**(**self**.**frame**)**

self**.**user**.**setGeometry**(**QtCore**.**QRect**(**122**,** 70**,** 421**,** 20**))**

font **=** QtGui**.**QFont**()**

font**.**setPointSize**(**10**)**

font**.**setBold**(True)**

font**.**setItalic**(False)**

font**.**setWeight**(**75**)**

self**.**user**.**setFont**(**font**)**

self**.**user**.**setClearButtonEnabled**(False)**

self**.**user**.**setObjectName**(**"user"**)**

self**.**phno **=** QtWidgets**.**QLineEdit**(**self**.**frame**)**

self**.**phno**.**setGeometry**(**QtCore**.**QRect**(**120**,** 120**,** 421**,** 20**))**

font **=** QtGui**.**QFont**()**

font**.**setPointSize**(**10**)**

font**.**setBold**(True)**

font**.**setItalic**(False)**

font**.**setWeight**(**75**)**

self**.**phno**.**setFont**(**font**)**

self**.**phno**.**setObjectName**(**"phno"**)**

self**.**password **=** QtWidgets**.**QLineEdit**(**self**.**frame**)**

self**.**password**.**setGeometry**(**QtCore**.**QRect**(**120**,** 250**,** 421**,** 20**))**

self**.**password**.**setEchoMode**(**QtWidgets**.**QLineEdit**.**Password**)**

font **=** QtGui**.**QFont**()**

font**.**setPointSize**(**10**)**

font**.**setBold**(True)**

font**.**setItalic**(False)**

font**.**setWeight**(**75**)**

self**.**password**.**setFont**(**font**)**

self**.**password**.**setObjectName**(**"password"**)**

self**.**checkBox **=** QtWidgets**.**QCheckBox**(**self**.**frame**)**

self**.**checkBox**.**setGeometry**(**QtCore**.**QRect**(**120**,** 280**,** 131**,** 17**))**

self**.**checkBox**.**stateChanged**.**connect**(**self**.**hide\_pass**)**

self**.**checkBox**.**setCursor**(**QtGui**.**QCursor**(**QtCore**.**Qt**.**PointingHandCursor**))**

font **=** QtGui**.**QFont**()**

font**.**setPointSize**(**10**)**

font**.**setBold**(True)**

font**.**setWeight**(**75**)**

self**.**checkBox**.**setFont**(**font**)**

self**.**checkBox**.**setStyleSheet**(**"color: rgb(255, 255, 255);"**)**

self**.**checkBox**.**setObjectName**(**"checkBox"**)**

self**.**address **=** QtWidgets**.**QLineEdit**(**self**.**frame**)**

self**.**address**.**setGeometry**(**QtCore**.**QRect**(**120**,** 170**,** 421**,** 51**))**

font **=** QtGui**.**QFont**()**

font**.**setPointSize**(**10**)**

font**.**setBold**(True)**

font**.**setItalic**(False)**

font**.**setWeight**(**75**)**

self**.**address**.**setFont**(**font**)**

self**.**address**.**setClearButtonEnabled**(True)**

self**.**address**.**setObjectName**(**"address"**)**

self**.**otp\_btn **=** QtWidgets**.**QPushButton**(**self**.**frame**)**

self**.**otp\_btn**.**setGeometry**(**QtCore**.**QRect**(**130**,** 320**,** 181**,** 31**))**

self**.**otp\_btn**.**clicked**.**connect**(**self**.**send**)** # Click Signal

self**.**otp\_btn**.**setCursor**(**QtGui**.**QCursor**(**QtCore**.**Qt**.**PointingHandCursor**))**

self**.**otp\_btn**.**setObjectName**(**"otp\_btn"**)**

self**.**otp\_fill **=** QtWidgets**.**QLineEdit**(**self**.**frame**)**

self**.**otp\_fill**.**setGeometry**(**QtCore**.**QRect**(**320**,** 320**,** 151**,** 31**))**

font **=** QtGui**.**QFont**()**

font**.**setPointSize**(**9**)**

font**.**setBold**(True)**

font**.**setWeight**(**75**)**

self**.**otp\_fill**.**setFont**(**font**)**

self**.**otp\_fill**.**setObjectName**(**"otp\_fill"**)**

self**.**label\_4 **=** QtWidgets**.**QLabel**(**self**.**frame**)**

self**.**label\_4**.**setGeometry**(**QtCore**.**QRect**(**220**,** 20**,** 201**,** 20**))**

font **=** QtGui**.**QFont**()**

font**.**setPointSize**(**9**)**

font**.**setBold**(True)**

font**.**setItalic**(True)**

font**.**setWeight**(**75**)**

self**.**label\_4**.**setFont**(**font**)**

self**.**label\_4**.**setStyleSheet**(**"color: rgb(255, 0, 0);"**)**

self**.**label\_4**.**setObjectName**(**"label\_4"**)**

self**.**regis\_btn **=** QtWidgets**.**QPushButton**(**self**.**frame**)**

self**.**regis\_btn**.**setEnabled**(False)**

self**.**regis\_btn**.**setGeometry**(**QtCore**.**QRect**(**250**,** 370**,** 161**,** 31**))**

self**.**regis\_btn**.**clicked**.**connect**(**self**.**add\_data**)** # Click Signal

self**.**regis\_btn**.**setCursor**(**QtGui**.**QCursor**(**QtCore**.**Qt**.**PointingHandCursor**))**

font **=** QtGui**.**QFont**()**

font**.**setPointSize**(**10**)**

font**.**setBold**(True)**

font**.**setWeight**(**75**)**

self**.**regis\_btn**.**setFont**(**font**)**

self**.**regis\_btn**.**setObjectName**(**"regis\_btn"**)**

self**.**OK **=** QtWidgets**.**QPushButton**(**self**.**frame**)**

self**.**OK**.**setGeometry**(**QtCore**.**QRect**(**480**,** 320**,** 51**,** 31**))**

self**.**OK**.**setEnabled**(False)**

self**.**OK**.**clicked**.**connect**(**self**.**check**)** # Click Signal

self**.**OK**.**setCursor**(**QtGui**.**QCursor**(**QtCore**.**Qt**.**PointingHandCursor**))**

self**.**OK**.**setObjectName**(**"OK"**)**

self**.**label\_5 **=** QtWidgets**.**QLabel**(**self**.**centralwidget**)**

self**.**label\_5**.**setGeometry**(**QtCore**.**QRect**(**290**,** 540**,** 171**,** 16**))**

font **=** QtGui**.**QFont**()**

font**.**setPointSize**(**9**)**

self**.**label\_5**.**setFont**(**font**)**

self**.**label\_5**.**setStyleSheet**(**"color: rgb(255, 51, 15);"**)**

self**.**label\_5**.**setObjectName**(**"label\_5"**)**

Register**.**setCentralWidget**(**self**.**centralwidget**)**

self**.**retranslateUi**(**Register**)**

QtCore**.**QMetaObject**.**connectSlotsByName**(**Register**)**

**def** retranslateUi**(**self**,** Register**):**

\_translate **=** QtCore**.**QCoreApplication**.**translate

Register**.**setWindowTitle**(**\_translate**(**"Register"**,** "Regsitration"**))**

self**.**label\_3**.**setText**(**\_translate**(**"Register"**,** " Registration"**))**

self**.**user**.**setPlaceholderText**(**\_translate**(**"Register"**,** "Full Name"**))**

self**.**phno**.**setPlaceholderText**(**\_translate**(**"Register"**,** "Phone Number"**))**

self**.**password**.**setPlaceholderText**(**\_translate**(**"Register"**,** "Set Password"**))**

self**.**checkBox**.**setText**(**\_translate**(**"Register"**,** " Show Password"**))**

self**.**address**.**setPlaceholderText**(**\_translate**(**"Register"**,** "Address"**))**

self**.**otp\_btn**.**setText**(**\_translate**(**"Register"**,** "Get Your OTP.."**))**

self**.**otp\_fill**.**setPlaceholderText**(**\_translate**(**"Register"**,** "Enter Your OTP

Here !"**))**

self**.**label\_4**.**setText**(**\_translate**(**"Register"**,** "All mentioned fields to be

filled"**))**

self**.**regis\_btn**.**setText**(**\_translate**(**"Register"**,** "Register"**))**

self**.**OK**.**setText**(**\_translate**(**"Register"**,** "OK"**))**

self**.**label\_5**.**setText**(**\_translate**(**"Register"**,** "For Customer Registration

Only"**))**

**if** \_\_name\_\_ **==** "\_\_main\_\_"**:**

**import** sys

app **=** QtWidgets**.**QApplication**(**sys**.**argv**)**

Register **=** QtWidgets**.**QMainWindow**()**

ui **=** Ui\_Register**()**

ui**.**setupUi**(**Register**)**

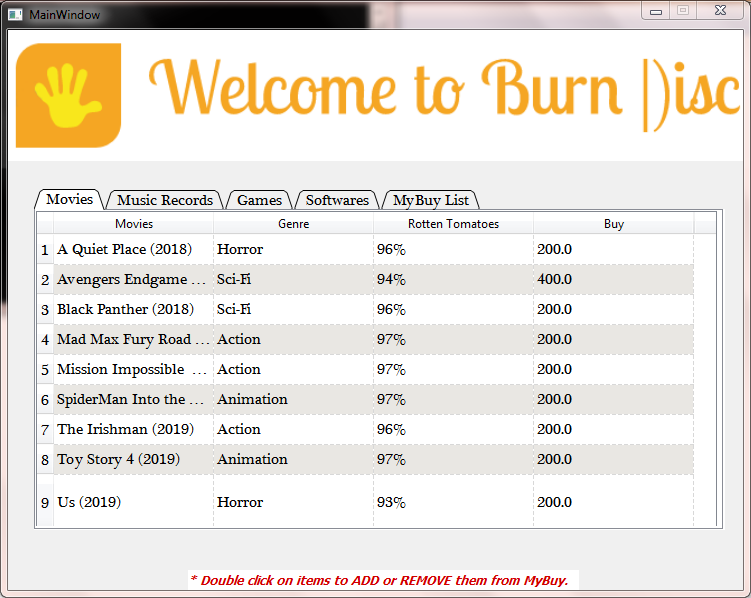
Register**.**show**()**

sys**.exit(**app**.**exec\_**()**

****

**Collections Window**

**(Shopping Window)**

****

# Importing Modules and Connecting to MYSQL

**from** PyQt5 **import** QtCore**,** QtGui**,** QtWidgets

**import** mysql**.**connector **as** sql

**from** tkinter **import** messagebox

**import** tkinter **as** Tk

**import** subprocess

root **=** Tk**.**Tk**()** # For hiding tkinter root window

root**.**withdraw**()**

# Python Mysql Connection

con**=**sql**.**connect**(**host**=**"localhost"**,** user**=**"root"**,** password**=**"saieshagre"**)**

cur**=**con**.**cursor**(**buffered**=True)**

cur**.**execute**(**"USE project;"**)**

# Graphical User Interface

**class** **Ui\_MainWindow(object):**

# Calculating Total Amount Payable and Redirecting to Payment Gateway

**def** Buy**(**self**):**

cur**.**execute**(**"select SUM(Price) from Buy;"**)**

a **=** cur**.**fetchone**()**

Total **=** a**[**0**]**

**if** Total **==** **None:**

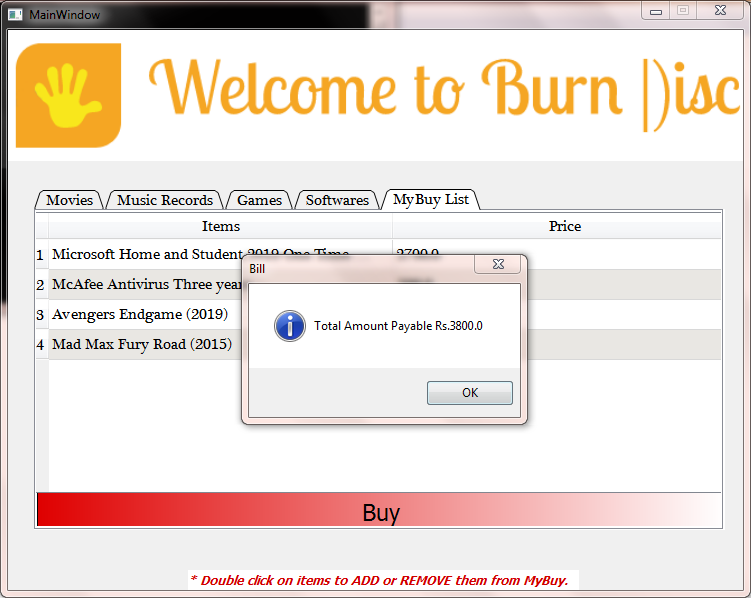
messagebox**.**showinfo**(**"Invalid Purchase "**,** "You have not purchased anything.."**)**

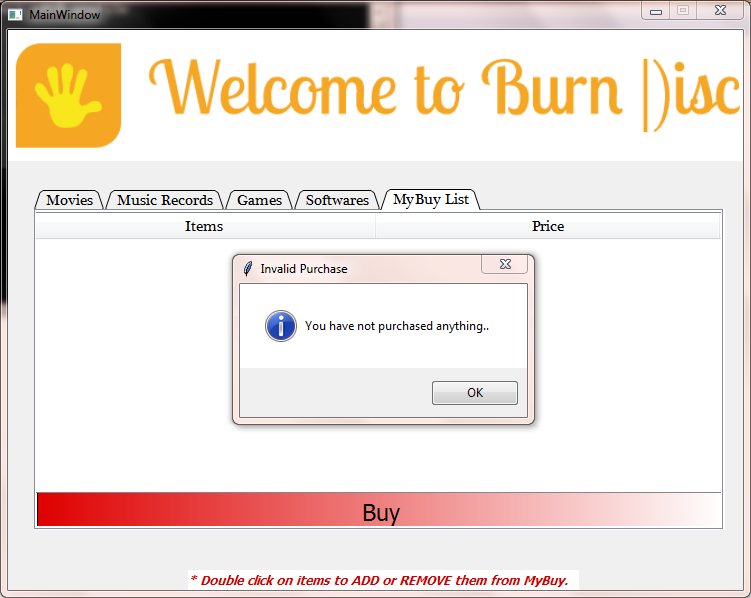
**return**

**else:**

messagebox**.**showinfo**(**"Bill"**,**"Total Amount Payable Rs."**+str(**Total**))**

subprocess**.**call**(**"Windows\\Pay\_way.py"**,** shell**=True)**





# Adding Items To MyBuy List

# Adding from Movies Table

**def** add\_1**(**self**):**

row **=** self**.**movies**.**currentRow**()**

item1 **=** self**.**movies**.**item**(**row**,** 0**).**text**()**

item2 **=** self**.**movies**.**item**(**row**,** 3**).**text**()**

query**=**"INSERT into Buy values(%s, %s)"

data**=**cur**.**execute**(**query**,(**item1**,** item2**))**

con**.**commit**()**

cur**.**execute**(**"select \* from Buy"**)**

result **=** cur**.**fetchall**()**

count **=** cur**.**rowcount

self**.**buy\_list**.**setRowCount**(**count**)**

**for** row\_number**,** row\_data **in** **enumerate(**result**):**

**for** column\_number**,** data **in** **enumerate(**row\_data**):**

self**.**buy\_list**.**setItem**(**row\_number**,** column\_number**,** QtWidgets**.**QTableWidgetItem**(str(**data**)))**

# Adding from Music Table

**def** add\_2**(**self**):**

row **=** self**.**music**.**currentRow**()**

item1 **=** self**.**music**.**item**(**row**,** 0**).**text**()**

item2 **=** self**.**music**.**item**(**row**,** 2**).**text**()**

query**=**"INSERT into Buy values(%s, %s)"

data**=**cur**.**execute**(**query**,(**item1**,** item2**))**

con**.**commit**()**

cur**.**execute**(**"select \* from Buy"**)**

result **=** cur**.**fetchall**()**

count **=** cur**.**rowcount

self**.**buy\_list**.**setRowCount**(**count**)**

**for** row\_number**,** row\_data **in** **enumerate(**result**):**

**for** column\_number**,** data **in** **enumerate(**row\_data**):**

self**.**buy\_list**.**setItem**(**row\_number**,** column\_number**,** QtWidgets**.**QTableWidgetItem**(str(**data**)))**

# Adding from Games Table

**def** add\_3**(**self**):**

row **=** self**.**games**.**currentRow**()**

item1 **=** self**.**games**.**item**(**row**,** 0**).**text**()**

item2 **=** self**.**games**.**item**(**row**,** 2**).**text**()**

query**=**"INSERT into Buy values(%s, %s)"

data**=**cur**.**execute**(**query**,(**item1**,** item2**))**

con**.**commit**()**

cur**.**execute**(**"select \* from Buy"**)**

result **=** cur**.**fetchall**()**

count **=** cur**.**rowcount

self**.**buy\_list**.**setRowCount**(**count**)**

**for** row\_number**,** row\_data **in** **enumerate(**result**):**

**for** column\_number**,** data **in** **enumerate(**row\_data**):**

self**.**buy\_list**.**setItem**(**row\_number**,** column\_number**,** QtWidgets**.**QTableWidgetItem**(str(**data**)))**

# Adding from Software Table

**def** add\_4**(**self**):**

row **=** self**.**software**.**currentRow**()**

item1 **=** self**.**software**.**item**(**row**,** 0**).**text**()**

item2 **=** self**.**software**.**item**(**row**,** 2**).**text**()**

query **=** "INSERT into Buy values(%s, %s)"

data **=** cur**.**execute**(**query**,(**item1**,** item2**))**

con**.**commit**()**

cur**.**execute**(**"select \* from Buy"**)**

result **=** cur**.**fetchall**()**

count **=** cur**.**rowcount

self**.**buy\_list**.**setRowCount**(**count**)**

**for** row\_number**,** row\_data **in** **enumerate(**result**):**

**for** column\_number**,** data **in** **enumerate(**row\_data**):**

self**.**buy\_list**.**setItem**(**row\_number**,** column\_number**,** QtWidgets**.**QTableWidgetItem**(str(**data**)))**

# Removing Items from MYBuy List

**def** remove**(**self**):**

row **=** self**.**buy\_list**.**currentRow**()**

item1 **=** self**.**buy\_list**.**item**(**row**,** 0**).**text**()**

item2 **=** **float(**self**.**buy\_list**.**item**(**row**,** 1**).**text**())**

query **=** "Delete from Buy where Items=%s and Price=%s "

data **=** cur**.**execute**(**query**,(**item1**,** item2**))**

con**.**commit**()**

cur**.**execute**(**"select \* from Buy"**)**

result **=** cur**.**fetchall**()**

count **=** cur**.**rowcount

self**.**buy\_list**.**setRowCount**(**count**)**

**for** row\_number**,** row\_data **in** **enumerate(**result**):**

**for** column\_number**,** data **in** **enumerate(**row\_data**):**

self**.**buy\_list**.**setItem**(**row\_number**,** column\_number**,** QtWidgets**.**QTableWidgetItem**(str(**data**)))**

# Showing Data from MYSQL Database to Python Table

**def** show\_data**(**self**):**

# Movies Table

con**.**commit**()**

self**.**show\_btn**.**hide**()**

cur**.**execute**(**"select \* from movie"**)**

result **=** cur**.**fetchall**()**

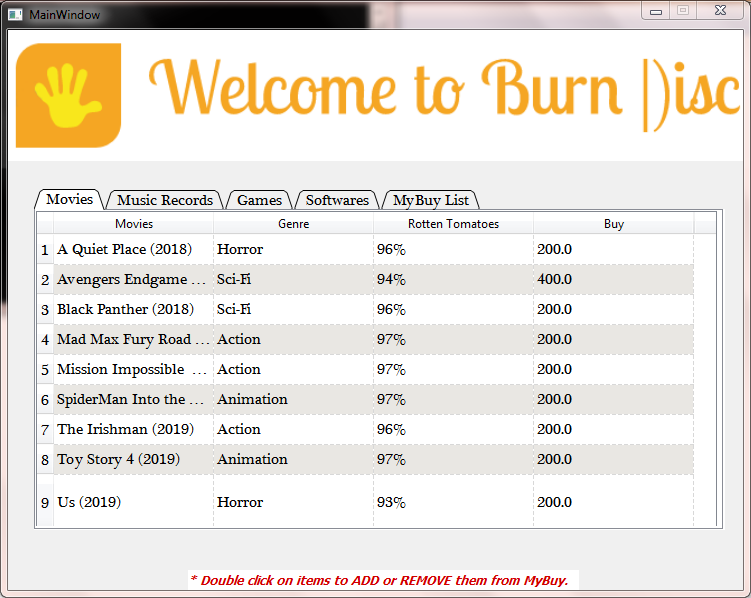
count **=** cur**.**rowcount

self**.**movies**.**setRowCount**(**count**)**

**for** row\_number**,** row\_data **in** **enumerate(**result**):**

**for** column\_number**,** data **in** **enumerate(**row\_data**):**

self**.**movies**.**setItem**(**row\_number**,** column\_number**,** QtWidgets**.**QTableWidgetItem**(str(**data**)))**



# Music Table

cur**.**execute**(**"select \* from music"**)**

result **=** cur**.**fetchall**()**

count **=** cur**.**rowcount

self**.**music**.**setRowCount**(**count**)**

**for** row\_number**,** row\_data **in** **enumerate(**result**):**

**for** column\_number**,** data **in** **enumerate(**row\_data**):**

self**.**music**.**setItem**(**row\_number**,** column\_number**,** QtWidgets**.**QTableWidgetItem**(str(**data**)))**



# Game Table

cur**.**execute**(**"select \* from game"**)**

result **=** cur**.**fetchall**()**

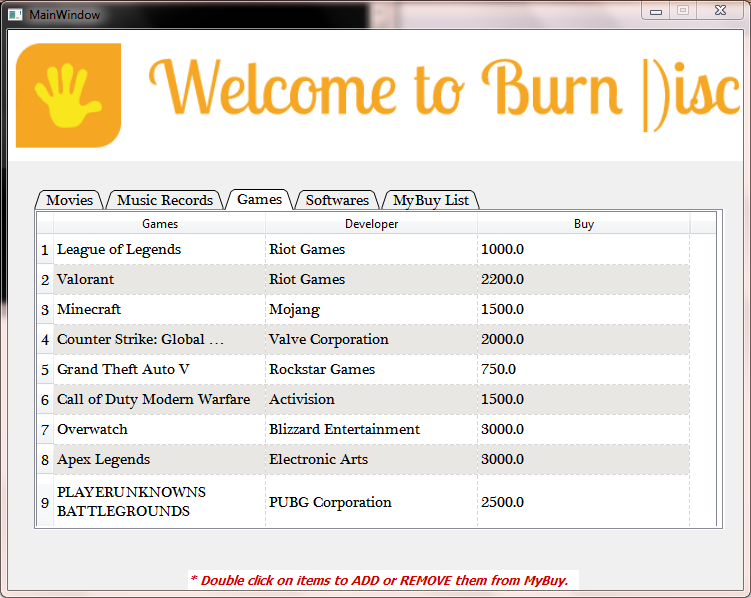
count **=** cur**.**rowcount

self**.**games**.**setRowCount**(**count**)**

**for** row\_number**,** row\_data **in** **enumerate(**result**):**

**for** column\_number**,** data **in** **enumerate(**row\_data**):**

self**.**games**.**setItem**(**row\_number**,** column\_number**,** QtWidgets**.**QTableWidgetItem**(str(**data**)))**



# Software Table

cur**.**execute**(**"select \* from software"**)**

result **=** cur**.**fetchall**()**

count **=** cur**.**rowcount

self**.**software**.**setRowCount**(**count**)**

**for** row\_number**,** row\_data **in** **enumerate(**result**):**

**for** column\_number**,** data **in** **enumerate(**row\_data**):**

self**.**software**.**setItem**(**row\_number**,** column\_number**,** QtWidgets**.**QTableWidgetItem**(str(**data**)))**



**def** setupUi**(**self**,** MainWindow**):**

MainWindow**.**setObjectName**(**"MainWindow"**)**

MainWindow**.**setFixedSize**(**735**,** 560**)**

self**.**centralwidget **=** QtWidgets**.**QWidget**(**MainWindow**)**

self**.**centralwidget**.**setObjectName**(**"centralwidget"**)**

self**.**label **=** QtWidgets**.**QLabel**(**self**.**centralwidget**)**

self**.**label**.**setGeometry**(**QtCore**.**QRect**(-**4**,** **-**8**,** 741**,** 571**))**

font **=** QtGui**.**QFont**()**

font**.**setBold**(False)**

font**.**setWeight**(**50**)**

self**.**label**.**setFont**(**font**)**

self**.**label**.**setFrameShape**(**QtWidgets**.**QFrame**.**NoFrame**)**

self**.**label**.**setText**(**""**)**

self**.**label**.**setPixmap**(**QtGui**.**QPixmap**(**"Windows\\Icons\\Bg2.png"**))**

self**.**label**.**setScaledContents**(True)**

self**.**label**.**setWordWrap**(True)**

self**.**label**.**setObjectName**(**"label"**)**

self**.**label\_2 **=** QtWidgets**.**QLabel**(**self**.**centralwidget**)**

self**.**label\_2**.**setGeometry**(**QtCore**.**QRect**(**0**,** 0**,** 741**,** 131**))**

self**.**label\_2**.**setText**(**""**)**

self**.**label\_2**.**setPixmap**(**QtGui**.**QPixmap**(**"Windows\\Icons\\Welcome.png"**))**

self**.**label\_2**.**setScaledContents**(True)**

self**.**label\_2**.**setObjectName**(**"label\_2"**)**

self**.**tabWidget **=** QtWidgets**.**QTabWidget**(**self**.**centralwidget**)**

self**.**tabWidget**.**setGeometry**(**QtCore**.**QRect**(**26**,** 159**,** 691**,** 341**))**

font **=** QtGui**.**QFont**()**

font**.**setFamily**(**"Plantagenet Cherokee"**)**

font**.**setPointSize**(**11**)**

font**.**setBold**(False)**

font**.**setItalic**(False)**

font**.**setUnderline**(False)**

font**.**setWeight**(**50**)**

self**.**tabWidget**.**setFont**(**font**)**

self**.**tabWidget**.**setCursor**(**QtGui**.**QCursor**(**QtCore**.**Qt**.**PointingHandCursor**))**

self**.**tabWidget**.**setTabShape**(**QtWidgets**.**QTabWidget**.**Triangular**)**

self**.**tabWidget**.**setObjectName**(**"tabWidget"**)**

self**.**mov **=** QtWidgets**.**QWidget**()**

self**.**mov**.**setObjectName**(**"mov"**)**

self**.**movies **=** QtWidgets**.**QTableWidget**(**self**.**mov**)**

self**.**movies**.**setGeometry**(**QtCore**.**QRect**(**0**,** 0**,** 681**,** 321**))**

self**.**movies**.**cellDoubleClicked**.**connect**(**self**.**add\_1**)** # Double Click Signal

self**.**movies**.**viewport**().**setProperty**(**"cursor"**,** QtGui**.**QCursor**(**QtCore**.**Qt**.**ArrowCursor**))**

self**.**movies**.**setFocusPolicy**(**QtCore**.**Qt**.**ClickFocus**)**

self**.**movies**.**setAutoFillBackground**(False)**

self**.**movies**.**setStyleSheet**(**""**)**

self**.**movies**.**setFrameShape**(**QtWidgets**.**QFrame**.**StyledPanel**)**

self**.**movies**.**setLineWidth**(**1**)**

self**.**movies**.**setHorizontalScrollBarPolicy**(**QtCore**.**Qt**.**ScrollBarAlwaysOff**)**

self**.**movies**.**setSizeAdjustPolicy**(**QtWidgets**.**QAbstractScrollArea**.**AdjustToContentsOnFirstShow**)**

self**.**movies**.**setEditTriggers**(**QtWidgets**.**QAbstractItemView**.**NoEditTriggers**)**

self**.**movies**.**setAlternatingRowColors**(True)**

self**.**movies**.**setSelectionMode**(**QtWidgets**.**QAbstractItemView**.**SingleSelection**)**

self**.**movies**.**setSelectionBehavior**(**QtWidgets**.**QAbstractItemView**.**SelectRows**)**

self**.**movies**.**setShowGrid**(True)**

self**.**movies**.**setGridStyle**(**QtCore**.**Qt**.**DashLine**)**

self**.**movies**.**setWordWrap**(True)**

self**.**movies**.**setCornerButtonEnabled**(False)**

self**.**movies**.**setRowCount**(**50**)**

self**.**movies**.**setColumnCount**(**4**)**

self**.**movies**.**setObjectName**(**"movies"**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**movies**.**setHorizontalHeaderItem**(**0**,** item**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**movies**.**setHorizontalHeaderItem**(**1**,** item**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**movies**.**setHorizontalHeaderItem**(**2**,** item**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**movies**.**setHorizontalHeaderItem**(**3**,** item**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

item**.**setFlags**(**QtCore**.**Qt**.**ItemIsSelectable**|**QtCore**.**Qt**.**ItemIsEnabled**)**

self**.**movies**.**setItem**(**0**,** 0**,** item**)**

self**.**movies**.**horizontalHeader**().**setCascadingSectionResizes**(False)**

self**.**movies**.**horizontalHeader**().**setDefaultSectionSize**(**160**)**

self**.**movies**.**horizontalHeader**().**setMinimumSectionSize**(**30**)**

self**.**movies**.**verticalHeader**().**setDefaultSectionSize**(**30**)**

self**.**movies**.**verticalHeader**().**setMinimumSectionSize**(**23**)**

self**.**movies**.**verticalHeader**().**setStretchLastSection**(True)**

self**.**tabWidget**.**addTab**(**self**.**mov**,** ""**)**

self**.**mus **=** QtWidgets**.**QWidget**()**

self**.**mus**.**setObjectName**(**"mus"**)**

self**.**music **=** QtWidgets**.**QTableWidget**(**self**.**mus**)**

self**.**music**.**setGeometry**(**QtCore**.**QRect**(**0**,** 0**,** 681**,** 321**))**

self**.**music**.**cellDoubleClicked**.**connect**(**self**.**add\_2**)** # Double Click Signal

self**.**music**.**viewport**().**setProperty**(**"cursor"**,** QtGui**.**QCursor**(**QtCore**.**Qt**.**ArrowCursor**))**

self**.**music**.**setStyleSheet**(**""**)**

self**.**music**.**setFrameShape**(**QtWidgets**.**QFrame**.**StyledPanel**)**

self**.**music**.**setLineWidth**(**1**)**

self**.**music**.**setHorizontalScrollBarPolicy**(**QtCore**.**Qt**.**ScrollBarAlwaysOff**)**

self**.**music**.**setEditTriggers**(**QtWidgets**.**QAbstractItemView**.**NoEditTriggers**)**

self**.**music**.**setAlternatingRowColors**(True)**

self**.**music**.**setSelectionMode**(**QtWidgets**.**QAbstractItemView**.**SingleSelection**)**

self**.**music**.**setSelectionBehavior**(**QtWidgets**.**QAbstractItemView**.**SelectRows**)**

self**.**music**.**setGridStyle**(**QtCore**.**Qt**.**DashLine**)**

self**.**music**.**setWordWrap**(True)**

self**.**music**.**setCornerButtonEnabled**(True)**

self**.**music**.**setRowCount**(**25**)**

self**.**music**.**setColumnCount**(**3**)**

self**.**music**.**setObjectName**(**"music"**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**music**.**setHorizontalHeaderItem**(**0**,** item**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**music**.**setHorizontalHeaderItem**(**1**,** item**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**music**.**setHorizontalHeaderItem**(**2**,** item**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

item**.**setFlags**(**QtCore**.**Qt**.**ItemIsSelectable**|**QtCore**.**Qt**.**ItemIsEnabled**)**

self**.**music**.**setItem**(**0**,** 0**,** item**)**

self**.**music**.**horizontalHeader**().**setCascadingSectionResizes**(False)**

self**.**music**.**horizontalHeader**().**setDefaultSectionSize**(**212**)**

self**.**music**.**horizontalHeader**().**setMinimumSectionSize**(**30**)**

self**.**music**.**verticalHeader**().**setDefaultSectionSize**(**30**)**

self**.**music**.**verticalHeader**().**setMinimumSectionSize**(**23**)**

self**.**music**.**verticalHeader**().**setStretchLastSection**(True)**

self**.**tabWidget**.**addTab**(**self**.**mus**,** ""**)**

self**.**gam **=** QtWidgets**.**QWidget**()**

self**.**gam**.**setObjectName**(**"gam"**)**

self**.**games **=** QtWidgets**.**QTableWidget**(**self**.**gam**)**

self**.**games**.**setGeometry**(**QtCore**.**QRect**(**0**,** 0**,** 681**,** 321**))**

self**.**games**.**cellDoubleClicked**.**connect**(**self**.**add\_3**)** # Double Click Signal

self**.**games**.**viewport**().**setProperty**(**"cursor"**,** QtGui**.**QCursor**(**QtCore**.**Qt**.**ArrowCursor**))**

self**.**games**.**setStyleSheet**(**""**)**

self**.**games**.**setFrameShape**(**QtWidgets**.**QFrame**.**StyledPanel**)**

self**.**games**.**setLineWidth**(**1**)**

self**.**games**.**setHorizontalScrollBarPolicy**(**QtCore**.**Qt**.**ScrollBarAlwaysOff**)**

self**.**games**.**setEditTriggers**(**QtWidgets**.**QAbstractItemView**.**NoEditTriggers**)**

self**.**games**.**setAlternatingRowColors**(True)**

self**.**games**.**setSelectionMode**(**QtWidgets**.**QAbstractItemView**.**SingleSelection**)**

self**.**games**.**setSelectionBehavior**(**QtWidgets**.**QAbstractItemView**.**SelectRows**)**

self**.**games**.**setGridStyle**(**QtCore**.**Qt**.**DashLine**)**

self**.**games**.**setWordWrap**(True)**

self**.**games**.**setCornerButtonEnabled**(True)**

self**.**games**.**setRowCount**(**25**)**

self**.**games**.**setColumnCount**(**3**)**

self**.**games**.**setObjectName**(**"games"**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**games**.**setHorizontalHeaderItem**(**0**,** item**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**games**.**setHorizontalHeaderItem**(**1**,** item**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**games**.**setHorizontalHeaderItem**(**2**,** item**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

item**.**setFlags**(**QtCore**.**Qt**.**ItemIsSelectable**|**QtCore**.**Qt**.**ItemIsEnabled**)**

self**.**games**.**setItem**(**0**,** 0**,** item**)**

self**.**games**.**horizontalHeader**().**setCascadingSectionResizes**(False)**

self**.**games**.**horizontalHeader**().**setDefaultSectionSize**(**212**)**

self**.**games**.**horizontalHeader**().**setMinimumSectionSize**(**30**)**

self**.**games**.**verticalHeader**().**setDefaultSectionSize**(**30**)**

self**.**games**.**verticalHeader**().**setMinimumSectionSize**(**23**)**

self**.**games**.**verticalHeader**().**setStretchLastSection**(True)**

self**.**tabWidget**.**addTab**(**self**.**gam**,** ""**)**

self**.**soft **=** QtWidgets**.**QWidget**()**

self**.**soft**.**setObjectName**(**"soft"**)**

self**.**software **=** QtWidgets**.**QTableWidget**(**self**.**soft**)**

self**.**software**.**setGeometry**(**QtCore**.**QRect**(**0**,** 0**,** 681**,** 321**))**

self**.**software**.**cellDoubleClicked**.**connect**(**self**.**add\_4**)** # Double Click Signal

self**.**software**.**viewport**().**setProperty**(**"cursor"**,** QtGui**.**QCursor**(**QtCore**.**Qt**.**ArrowCursor**))**

self**.**software**.**setStyleSheet**(**""**)**

self**.**software**.**setFrameShape**(**QtWidgets**.**QFrame**.**StyledPanel**)**

self**.**software**.**setLineWidth**(**1**)**

self**.**software**.**setHorizontalScrollBarPolicy**(**QtCore**.**Qt**.**ScrollBarAlwaysOff**)**

self**.**software**.**setEditTriggers**(**QtWidgets**.**QAbstractItemView**.**NoEditTriggers**)**

self**.**software**.**setAlternatingRowColors**(True)**

self**.**software**.**setSelectionMode**(**QtWidgets**.**QAbstractItemView**.**SingleSelection**)**

self**.**software**.**setSelectionBehavior**(**QtWidgets**.**QAbstractItemView**.**SelectRows**)**

self**.**software**.**setGridStyle**(**QtCore**.**Qt**.**DashLine**)**

self**.**software**.**setWordWrap**(True)**

self**.**software**.**setCornerButtonEnabled**(True)**

self**.**software**.**setRowCount**(**25**)**

self**.**software**.**setColumnCount**(**3**)**

self**.**software**.**setObjectName**(**"software"**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**software**.**setHorizontalHeaderItem**(**0**,** item**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**software**.**setHorizontalHeaderItem**(**1**,** item**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**software**.**setHorizontalHeaderItem**(**2**,** item**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

item**.**setFlags**(**QtCore**.**Qt**.**ItemIsSelectable**|**QtCore**.**Qt**.**ItemIsEnabled**)**

self**.**software**.**setItem**(**0**,** 0**,** item**)**

self**.**software**.**horizontalHeader**().**setCascadingSectionResizes**(False)**

self**.**software**.**horizontalHeader**().**setDefaultSectionSize**(**212**)**

self**.**software**.**horizontalHeader**().**setMinimumSectionSize**(**30**)**

self**.**software**.**verticalHeader**().**setDefaultSectionSize**(**30**)**

self**.**software**.**verticalHeader**().**setMinimumSectionSize**(**23**)**

self**.**software**.**verticalHeader**().**setStretchLastSection**(True)**

self**.**tabWidget**.**addTab**(**self**.**soft**,** ""**)**

self**.**kart **=** QtWidgets**.**QWidget**()**

self**.**kart**.**setObjectName**(**"kart"**)**

self**.**buy\_list **=** QtWidgets**.**QTableWidget**(**self**.**kart**)**

self**.**buy\_list**.**setGeometry**(**QtCore**.**QRect**(-**5**,** 1**,** 691**,** 281**))**

self**.**buy\_list**.**cellDoubleClicked**.**connect**(**self**.**remove**)** # Double Click Signal

self**.**buy\_list**.**setHorizontalScrollBarPolicy**(**QtCore**.**Qt**.**ScrollBarAlwaysOff**)**

self**.**buy\_list**.**setEditTriggers**(**QtWidgets**.**QAbstractItemView**.**NoEditTriggers**)**

self**.**buy\_list**.**setAlternatingRowColors**(True)**

self**.**buy\_list**.**setSelectionMode**(**QtWidgets**.**QAbstractItemView**.**SingleSelection**)**

self**.**buy\_list**.**setSelectionBehavior**(**QtWidgets**.**QAbstractItemView**.**SelectRows**)**

self**.**buy\_list**.**setShowGrid**(True)**

self**.**buy\_list**.**setGridStyle**(**QtCore**.**Qt**.**SolidLine**)**

self**.**buy\_list**.**setRowCount**(**0**)**

self**.**buy\_list**.**setColumnCount**(**2**)**

self**.**buy\_list**.**setObjectName**(**"buy\_list"**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**buy\_list**.**setHorizontalHeaderItem**(**0**,** item**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**buy\_list**.**setHorizontalHeaderItem**(**1**,** item**)**

self**.**buy\_list**.**horizontalHeader**().**setDefaultSectionSize**(**344**)**

self**.**buy **=** QtWidgets**.**QPushButton**(**self**.**kart**)**

self**.**buy**.**setGeometry**(**QtCore**.**QRect**(**0**,** 280**,** 691**,** 41**))**

self**.**buy**.**clicked**.**connect**(**self**.**Buy**)** # Click Signal

font **=** QtGui**.**QFont**()**

font**.**setPointSize**(**17**)**

self**.**buy**.**setFont**(**font**)**

self**.**buy**.**setCursor**(**QtGui**.**QCursor**(**QtCore**.**Qt**.**PointingHandCursor**))**

self**.**buy**.**setFocusPolicy**(**QtCore**.**Qt**.**ClickFocus**)**

self**.**buy**.**setStyleSheet**(**"background-color: qlineargradient(spread:pad, x1:0, y1:0, x2:1, y2:0,\

stop:0 rgba(223, 0, 0, 255), stop:1 rgba(255, 255, 255, 255));"**)**

self**.**buy**.**setFlat**(False)**

self**.**buy**.**setObjectName**(**"buy"**)**

self**.**buy**.**raise\_**()**

self**.**buy\_list**.**raise\_**()**

self**.**tabWidget**.**addTab**(**self**.**kart**,** ""**)**

self**.**label\_3 **=** QtWidgets**.**QLabel**(**self**.**centralwidget**)**

self**.**label\_3**.**setGeometry**(**QtCore**.**QRect**(**180**,** 540**,** 391**,** 20**))**

font **=** QtGui**.**QFont**()**

font**.**setPointSize**(**9**)**

font**.**setBold**(True)**

font**.**setItalic**(True)**

font**.**setWeight**(**75**)**

self**.**label\_3**.**setFont**(**font**)**

self**.**label\_3**.**setStyleSheet**(**"color: rgb(209, 0, 0);\n"

"background-color: rgb(255, 255, 255);"**)**

self**.**label\_3**.**setObjectName**(**"label\_3"**)**

self**.**show\_btn **=** QtWidgets**.**QPushButton**(**self**.**centralwidget**)**

self**.**show\_btn**.**setGeometry**(**QtCore**.**QRect**(**270**,** 500**,** 201**,** 31**))**

self**.**show\_btn**.**clicked**.**connect**(**self**.**show\_data**)** # Click Signal

font **=** QtGui**.**QFont**()**

font**.**setFamily**(**"Segoe UI"**)**

font**.**setPointSize**(**10**)**

font**.**setBold**(True)**

font**.**setItalic**(False)**

font**.**setWeight**(**75**)**

self**.**show\_btn**.**setFont**(**font**)**

self**.**show\_btn**.**setCursor**(**QtGui**.**QCursor**(**QtCore**.**Qt**.**PointingHandCursor**))**

self**.**show\_btn**.**setFocusPolicy**(**QtCore**.**Qt**.**ClickFocus**)**

self**.**show\_btn**.**setStyleSheet**(**"background-color: rgb(255, 170, 0);\n"

"color: rgb(255, 255, 255);"**)**

self**.**show\_btn**.**setDefault**(False)**

self**.**show\_btn**.**setFlat**(False)**

self**.**show\_btn**.**setObjectName**(**"show\_btn"**)**

MainWindow**.**setCentralWidget**(**self**.**centralwidget**)**

self**.**retranslateUi**(**MainWindow**)**

self**.**tabWidget**.**setCurrentIndex**(**0**)**

QtCore**.**QMetaObject**.**connectSlotsByName**(**MainWindow**)**

**def** retranslateUi**(**self**,** MainWindow**):**

\_translate **=** QtCore**.**QCoreApplication**.**translate

MainWindow**.**setWindowTitle**(**\_translate**(**"MainWindow"**,** "MainWindow"**))**

self**.**movies**.**setSortingEnabled**(True)**

item **=** self**.**movies**.**horizontalHeaderItem**(**0**)**

item**.**setText**(**\_translate**(**"MainWindow"**,** "Movies"**))**

item **=** self**.**movies**.**horizontalHeaderItem**(**1**)**

item**.**setText**(**\_translate**(**"MainWindow"**,** "Genre"**))**

item **=** self**.**movies**.**horizontalHeaderItem**(**2**)**

item**.**setText**(**\_translate**(**"MainWindow"**,** "Rotten Tomatoes"**))**

item **=** self**.**movies**.**horizontalHeaderItem**(**3**)**

item**.**setText**(**\_translate**(**"MainWindow"**,** "Buy"**))**

\_\_sortingEnabled **=** self**.**movies**.**isSortingEnabled**()**

self**.**movies**.**setSortingEnabled**(False)**

self**.**movies**.**setSortingEnabled**(**\_\_sortingEnabled**)**

self**.**tabWidget**.**setTabText**(**self**.**tabWidget**.**indexOf**(**self**.**mov**),** \_translate**(**"MainWindow"**,** "Movies"**))**

self**.**music**.**setSortingEnabled**(True)**

item **=** self**.**music**.**horizontalHeaderItem**(**0**)**

item**.**setText**(**\_translate**(**"MainWindow"**,** "Music / Albums"**))**

item **=** self**.**music**.**horizontalHeaderItem**(**1**)**

item**.**setText**(**\_translate**(**"MainWindow"**,** "Artists"**))**

item **=** self**.**music**.**horizontalHeaderItem**(**2**)**

item**.**setText**(**\_translate**(**"MainWindow"**,** "Buy"**))**

\_\_sortingEnabled **=** self**.**music**.**isSortingEnabled**()**

self**.**music**.**setSortingEnabled**(False)**

self**.**music**.**setSortingEnabled**(**\_\_sortingEnabled**)**

self**.**tabWidget**.**setTabText**(**self**.**tabWidget**.**indexOf**(**self**.**mus**),** \_translate**(**"MainWindow"**,** "Music Records"**))**

self**.**games**.**setSortingEnabled**(True)**

item **=** self**.**games**.**horizontalHeaderItem**(**0**)**

item**.**setText**(**\_translate**(**"MainWindow"**,** "Games"**))**

item **=** self**.**games**.**horizontalHeaderItem**(**1**)**

item**.**setText**(**\_translate**(**"MainWindow"**,** "Developer"**))**

item **=** self**.**games**.**horizontalHeaderItem**(**2**)**

item**.**setText**(**\_translate**(**"MainWindow"**,** "Buy"**))**

\_\_sortingEnabled **=** self**.**games**.**isSortingEnabled**()**

self**.**games**.**setSortingEnabled**(False)**

self**.**games**.**setSortingEnabled**(**\_\_sortingEnabled**)**

self**.**tabWidget**.**setTabText**(**self**.**tabWidget**.**indexOf**(**self**.**gam**),** \_translate**(**"MainWindow"**,** "Games"**))**

self**.**software**.**setSortingEnabled**(True)**

item **=** self**.**software**.**horizontalHeaderItem**(**0**)**

item**.**setText**(**\_translate**(**"MainWindow"**,** "Softwares"**))**

item **=** self**.**software**.**horizontalHeaderItem**(**1**)**

item**.**setText**(**\_translate**(**"MainWindow"**,** "Company"**))**

item **=** self**.**software**.**horizontalHeaderItem**(**2**)**

item**.**setText**(**\_translate**(**"MainWindow"**,** "Buy"**))**

\_\_sortingEnabled **=** self**.**software**.**isSortingEnabled**()**

self**.**software**.**setSortingEnabled**(False)**

self**.**software**.**setSortingEnabled**(**\_\_sortingEnabled**)**

self**.**tabWidget**.**setTabText**(**self**.**tabWidget**.**indexOf**(**self**.**soft**),** \_translate**(**"MainWindow"**,** "Softwares"**))**

item **=** self**.**buy\_list**.**horizontalHeaderItem**(**0**)**

item**.**setText**(**\_translate**(**"MainWindow"**,** "Items"**))**

item **=** self**.**buy\_list**.**horizontalHeaderItem**(**1**)**

item**.**setText**(**\_translate**(**"MainWindow"**,** "Price"**))**

self**.**buy**.**setText**(**\_translate**(**"MainWindow"**,** "Buy"**))**

self**.**tabWidget**.**setTabText**(**self**.**tabWidget**.**indexOf**(**self**.**kart**),** \_translate**(**"MainWindow"**,** "MyBuy List"**))**

self**.**label\_3**.**setText**(**\_translate**(**"MainWindow"**,** "\* Double click on items to ADD or REMOVE them from

MyBuy."**))**

self**.**show\_btn**.**setText**(**\_translate**(**"MainWindow"**,** "View Collections"**))**

**if** \_\_name\_\_ **==** "\_\_main\_\_"**:**

**import** sys

app **=** QtWidgets**.**QApplication**(**sys**.**argv**)**

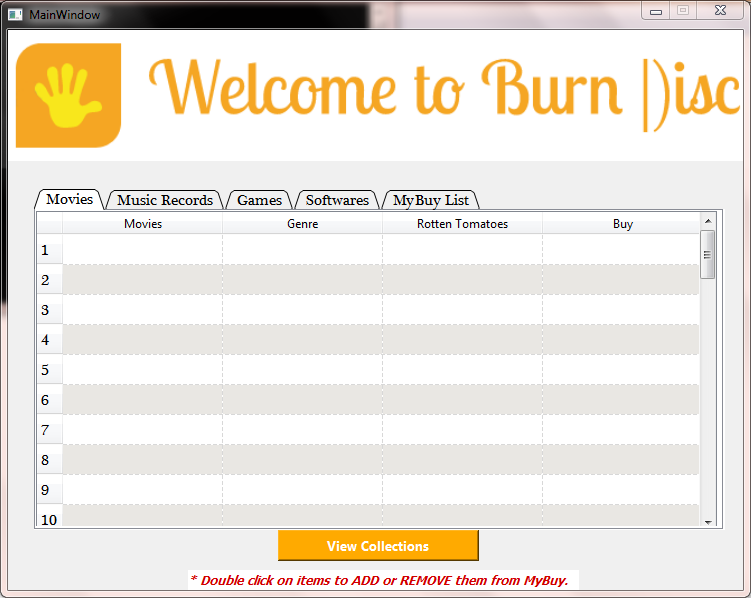
MainWindow **=** QtWidgets**.**QMainWindow**()**

ui **=** Ui\_MainWindow**()**

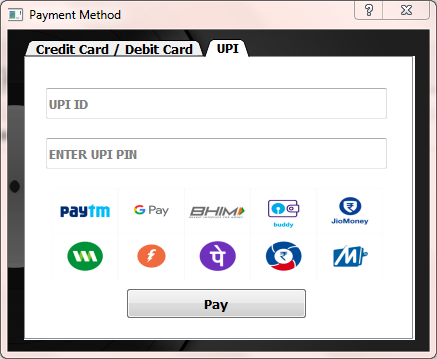
ui**.**setupUi**(**MainWindow**)**

MainWindow**.**show**()**

sys**.exit(**app**.**exec\_**())**

****

**Payment Window**

****

# Importing Modules and Connecting MYSQL

**from** PyQt5 **import** QtCore**,** QtGui**,** QtWidgets

**from** tkinter **import** messagebox

**import** mysql**.**connector **as** sql

**import** tkinter **as** Tk

root **=** Tk**.**Tk**()**  # For hiding tkinter root window

root**.**withdraw**()**

# Python Mysql Connection

con**=**sql**.**connect**(**host**=**"localhost"**,** user**=**"root"**,** password**=**"saieshagre"**)**

cur**=**con**.**cursor**(**buffered**=True)**

cur**.**execute**(**"USE project"**)**

# Graphical User Interface

**class** **Ui\_Dialog(object):**

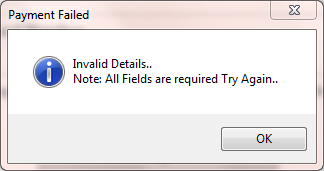
# Paying With Credit / Debit Card

**def** credit\_debit**(**self**):**

**if** **(len(**self**.**date**.**text**()** **and** self**.**holder\_name**.**text**()** **and** self**.**cvv**.**text**()** **and** self**.**card\_no**.**text**())==**0**)** **or** **(len(**self**.**card\_no**.**text**())!=**16**):**

messagebox**.**showinfo**(**"Payment Failed"**,** "Invalid Details or \n Note: All Fields are required Try Again.."**)**

**return**

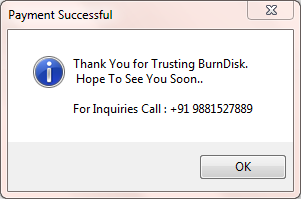


**else:**

messagebox**.**showinfo**(**"Payment Successful"**,**'''Thank You for Trusting BurnDisk.\n Hope To See You Soon..\n

For Inquiries Call : +91 9881527889 '''**)**

cur**.**execute**(**"delete from Buy;"**)**



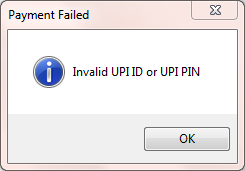
# Paying With UPI

**def** upi\_pay**(**self**):**

**if** **(**'@' **not** **in** self**.**upi\_id**.**text**())** **or** **len(**self**.**upi\_pin**.**text**())<**4**:**

messagebox**.**showinfo**(**"Payment Failed"**,** "Invalid UPI ID or UPI PIN"**)**

**return**



**else:**

messagebox**.**showinfo**(**"Payment Successful"**,**'''Thank You.\n Hope To See You Soon..\n

For Inquiries Call : +91 9881527889 '''**)**

cur**.**execute**(**"delete from Buy;"**)**

**def** setupUi**(**self**,** Dialog**):**

Dialog**.**setObjectName**(**"Dialog"**)**

Dialog**.**setFixedSize**(**421**,** 321**)**

self**.**label **=** QtWidgets**.**QLabel**(**Dialog**)**

self**.**label**.**setGeometry**(**QtCore**.**QRect**(-**4**,** **-**8**,** 431**,** 331**))**

self**.**label**.**setText**(**""**)**

self**.**label**.**setPixmap**(**QtGui**.**QPixmap**(**"Windows\\Icons\\Bg.png"**))**

self**.**label**.**setScaledContents**(True)**

self**.**label**.**setObjectName**(**"label"**)**

self**.**tabWidget **=** QtWidgets**.**QTabWidget**(**Dialog**)**

self**.**tabWidget**.**setGeometry**(**QtCore**.**QRect**(**16**,** 9**,** 391**,** 301**))**

font **=** QtGui**.**QFont**()**

font**.**setPointSize**(**10**)**

font**.**setBold**(True)**

font**.**setWeight**(**75**)**

self**.**tabWidget**.**setFont**(**font**)**

self**.**tabWidget**.**setTabShape**(**QtWidgets**.**QTabWidget**.**Triangular**)**

self**.**tabWidget**.**setTabBarAutoHide**(False)**

self**.**tabWidget**.**setObjectName**(**"tabWidget"**)**

self**.**credit **=** QtWidgets**.**QWidget**()**

self**.**credit**.**setObjectName**(**"credit"**)**

self**.**card\_no **=** QtWidgets**.**QLineEdit**(**self**.**credit**)**

self**.**card\_no**.**setGeometry**(**QtCore**.**QRect**(**20**,** 90**,** 351**,** 31**))**

font **=** QtGui**.**QFont**()**

font**.**setPointSize**(**11**)**

font**.**setBold**(True)**

font**.**setWeight**(**75**)**

self**.**card\_no**.**setFont**(**font**)**

self**.**card\_no**.**setObjectName**(**"card\_no"**)**

self**.**date **=** QtWidgets**.**QLineEdit**(**self**.**credit**)**

self**.**date**.**setGeometry**(**QtCore**.**QRect**(**20**,** 150**,** 171**,** 31**))**

font **=** QtGui**.**QFont**()**

font**.**setPointSize**(**11**)**

font**.**setBold**(True)**

font**.**setWeight**(**75**)**

self**.**date**.**setFont**(**font**)**

self**.**date**.**setObjectName**(**"date"**)**

self**.**holder\_name **=** QtWidgets**.**QLineEdit**(**self**.**credit**)**

self**.**holder\_name**.**setGeometry**(**QtCore**.**QRect**(**20**,** 20**,** 351**,** 31**))**

font **=** QtGui**.**QFont**()**

font**.**setPointSize**(**11**)**

font**.**setBold**(True)**

font**.**setWeight**(**75**)**

self**.**holder\_name**.**setFont**(**font**)**

self**.**holder\_name**.**setObjectName**(**"holder\_name"**)**

self**.**cvv **=** QtWidgets**.**QLineEdit**(**self**.**credit**)**

self**.**cvv**.**setGeometry**(**QtCore**.**QRect**(**210**,** 150**,** 161**,** 31**))**

font **=** QtGui**.**QFont**()**

font**.**setPointSize**(**11**)**

font**.**setBold**(True)**

font**.**setWeight**(**75**)**

self**.**cvv**.**setFont**(**font**)**

self**.**cvv**.**setEchoMode**(**QtWidgets**.**QLineEdit**.**Password**)**

self**.**cvv**.**setObjectName**(**"cvv"**)**

self**.**pay\_1 **=** QtWidgets**.**QPushButton**(**self**.**credit**)**

self**.**pay\_1**.**setGeometry**(**QtCore**.**QRect**(**74**,** 210**,** 241**,** 41**))**

self**.**pay\_1**.**clicked**.**connect**(**self**.**credit\_debit**)** # Click Signal

self**.**pay\_1**.**setObjectName**(**"pushButton"**)**

self**.**label\_2 **=** QtWidgets**.**QLabel**(**self**.**credit**)**

self**.**label\_2**.**setGeometry**(**QtCore**.**QRect**(**20**,** 60**,** 121**,** 31**))**

self**.**label\_2**.**setText**(**""**)**

self**.**label\_2**.**setPixmap**(**QtGui**.**QPixmap**(**"Windows\\Icons\\visa\_master.png"**))**

self**.**label\_2**.**setScaledContents**(True)**

self**.**label\_2**.**setObjectName**(**"label\_2"**)**

self**.**tabWidget**.**addTab**(**self**.**credit**,** ""**)**

self**.**upi **=** QtWidgets**.**QWidget**()**

self**.**upi**.**setObjectName**(**"upi"**)**

self**.**upi\_id **=** QtWidgets**.**QLineEdit**(**self**.**upi**)**

self**.**upi\_id**.**setGeometry**(**QtCore**.**QRect**(**20**,** 30**,** 341**,** 31**))**

self**.**upi\_id**.**setObjectName**(**"lineEdit\_2"**)**

self**.**upi\_pin **=** QtWidgets**.**QLineEdit**(**self**.**upi**)**

self**.**upi\_pin**.**setGeometry**(**QtCore**.**QRect**(**20**,** 80**,** 341**,** 31**))**

self**.**upi\_pin**.**setObjectName**(**"lineEdit\_3"**)**

self**.**label\_3 **=** QtWidgets**.**QLabel**(**self**.**upi**)**

self**.**label\_3**.**setGeometry**(**QtCore**.**QRect**(**26**,** 130**,** 331**,** 91**))**

self**.**label\_3**.**setText**(**""**)**

self**.**label\_3**.**setPixmap**(**QtGui**.**QPixmap**(**"Windows\\Icons\\upi.png"**))**

self**.**label\_3**.**setScaledContents**(True)**

self**.**label\_3**.**setObjectName**(**"label\_3"**)**

self**.**pay\_2 **=** QtWidgets**.**QPushButton**(**self**.**upi**)**

self**.**pay\_2**.**setGeometry**(**QtCore**.**QRect**(**100**,** 230**,** 181**,** 31**))**

self**.**pay\_2**.**clicked**.**connect**(**self**.**upi\_pay**)**# Click Signal

self**.**pay\_2**.**setObjectName**(**"pushButton\_2"**)**

self**.**tabWidget**.**addTab**(**self**.**upi**,** ""**)**

self**.**retranslateUi**(**Dialog**)**

self**.**tabWidget**.**setCurrentIndex**(**0**)**

QtCore**.**QMetaObject**.**connectSlotsByName**(**Dialog**)**

**def** retranslateUi**(**self**,** Dialog**):**

\_translate **=** QtCore**.**QCoreApplication**.**translate

Dialog**.**setWindowTitle**(**\_translate**(**"Dialog"**,** "Payment Method"**))**

self**.**card\_no**.**setPlaceholderText**(**\_translate**(**"Dialog"**,** "Card Number"**))**

self**.**date**.**setPlaceholderText**(**\_translate**(**"Dialog"**,** "Expiry Date"**))**

self**.**holder\_name**.**setPlaceholderText**(**\_translate**(**"Dialog"**,** "Card Holder\'s Name"**))**

self**.**cvv**.**setPlaceholderText**(**\_translate**(**"Dialog"**,** "CVV / Security Code"**))**

self**.**pay\_1**.**setText**(**\_translate**(**"Dialog"**,** "Pay"**))**

self**.**tabWidget**.**setTabText**(**self**.**tabWidget**.**indexOf**(**self**.**credit**),** \_translate**(**"Dialog"**,** "Credit Card / Debit Card"**))**

self**.**upi\_id**.**setPlaceholderText**(**\_translate**(**"Dialog"**,** "UPI ID"**))**

self**.**upi\_pin**.**setPlaceholderText**(**\_translate**(**"Dialog"**,** "ENTER UPI PIN"**))**

self**.**pay\_2**.**setText**(**\_translate**(**"Dialog"**,** "Pay"**))**

self**.**tabWidget**.**setTabText**(**self**.**tabWidget**.**indexOf**(**self**.**upi**),** \_translate**(**"Dialog"**,** "UPI"**)**

**if** \_\_name\_\_ **==** "\_\_main\_\_"**:**

**import** sys

app **=** QtWidgets**.**QApplication**(**sys**.**argv**)**

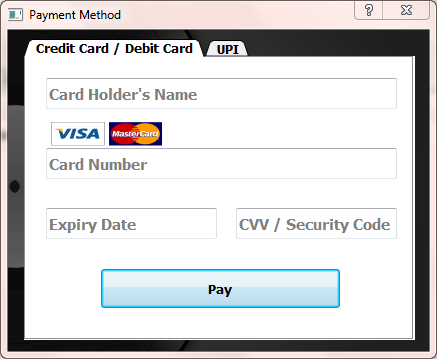
Dialog **=** QtWidgets**.**QDialog**()**

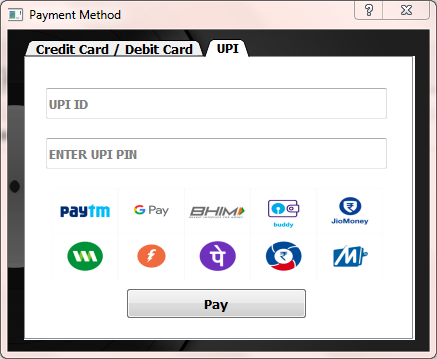
ui **=** Ui\_Dialog**()**

ui**.**setupUi**(**Dialog**)**

Dialog**.**show**()**

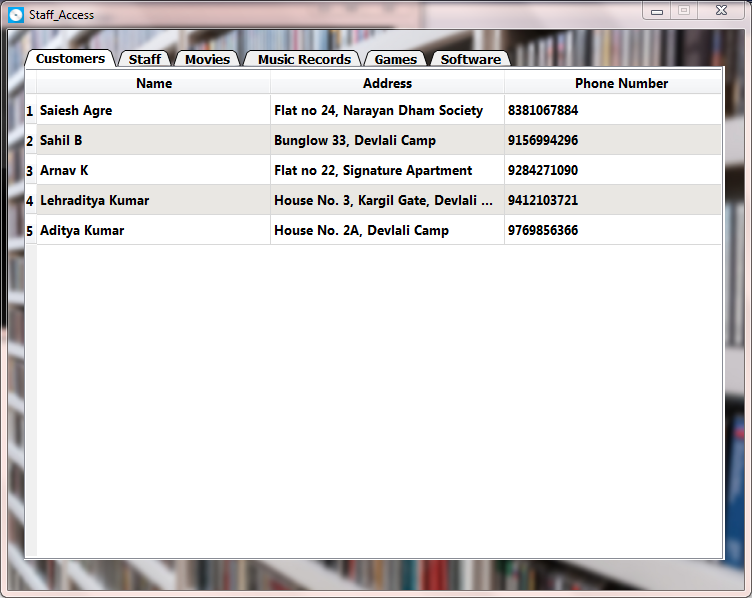
sys**.exit(**app**.**exec\_**())**

****

****

**Staff Window**

**(for Updating or Altering Collections)**

****

# Importing Modules and Connecting with MYSQL

**from** PyQt5 **import** QtCore**,** QtGui**,** QtWidgets

**import** mysql**.**connector **as** sql

**import** tkinter **as** Tk

**from** tkinter **import** messagebox

root **=** Tk**.**Tk**()** # For hiding tkinter root window

root**.**withdraw**()**

# Connecting to MYSQL and Selecting Database

con**=**sql**.**connect**(**host**=**"localhost"**,** user**=**"root"**,** password**=**"saieshagre"**)**

cur**=**con**.**cursor**(**buffered**=True)**

cur**.**execute**(**"USE project;"**)**

# Graphical User Intergace

**class** **Ui\_Register(object):**

# Adding New Records to Tables

# MOVIES TABLE

**def** add\_1**(**self**):**

a **=** self**.**lineEdit**.**text**()**

b **=** self**.**lineEdit\_2**.**text**()**

c **=** self**.**lineEdit\_3**.**text**()**

d **=** self**.**lineEdit\_4**.**text**()**

# Checking if records already exists already or entered records are empty

data **=** "select \* from movie where Movies=%s and Genre=%s"

cur**.**execute**(**data**,(**a**,** b**))**

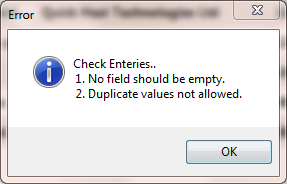
count **=** cur**.**rowcount

**if** **(len(**a **or** b **or** c **or** d**))==**0 **or** count**>=**1**:**

messagebox**.**showinfo**(**"Error"**,** "Check Enteries.. \n 1. No field should be empty, \n 2. Duplicate values not

allowed."**)**

**return**



**else:**

**pass**

# Inserting values after meeting requirements

**try:**

query **=** "insert into movie values(%s, %s, %s, %s)"

cur**.**execute**(**query**,(**a**,** b**,** c**,** d**))**

con**.**commit**()**

cur**.**execute**(**"select \* from movie"**)**

result **=** cur**.**fetchall**()**

count **=** cur**.**rowcount

self**.**Movies**.**setRowCount**(**count**)**

**for** row\_number**,** row\_data **in** **enumerate(**result**):**

**for** column\_number**,** data **in** **enumerate(**row\_data**):**

self**.**Movies**.**setItem**(**row\_number**,** column\_number**,** QtWidgets**.**QTableWidgetItem**(str(**data**)))**

**except:**

**pass**

# MUSIC TABLE

**def** add\_2**(**self**):**

a **=** self**.**lineEdit\_5**.**text**()**

b **=** self**.**lineEdit\_6**.**text**()**

c **=** self**.**lineEdit\_7**.**text**()**

# Checking if records already exists already or entered records are empty

data **=** "select \* from music where Music=%s and Artists=%s"

cur**.**execute**(**data**,(**a**,** b**))**

count **=** cur**.**rowcount

**if** **len(**a **or** b **or** c**)==**0 **or** count**>=**1**:**

messagebox**.**showinfo**(**"Error"**,** "Check Enteries.. \n 1. No field should be empty, \n 2. Duplicate values not

allowed."**)**

**return**

**else:**

**pass**

# Inserting values after meeting requirements

**try:**

query **=** "insert into music values(%s, %s, %s)"

cur**.**execute**(**query**,(**a**,** b**,** c**))**

con**.**commit**()**

cur**.**execute**(**"select \* from music"**)**

result **=** cur**.**fetchall**()**

count **=** cur**.**rowcount

self**.**music\_2**.**setRowCount**(**count**)**

**for** row\_number**,** row\_data **in** **enumerate(**result**):**

**for** column\_number**,** data **in** **enumerate(**row\_data**):**

self**.**music\_2**.**setItem**(**row\_number**,** column\_number**,** QtWidgets**.**QTableWidgetItem**(str(**data**)))**

**except:**

**pass**

# GAME TABLE

**def** add\_3**(**self**):**

a **=** self**.**lineEdit\_8**.**text**()**

b **=** self**.**lineEdit\_9**.**text**()**

c **=** self**.**lineEdit\_10**.**text**()**

# Checking if records already exists already or entered records are empty

data **=** "select \* from game where Games=%s and Category=%s"

cur**.**execute**(**data**,(**a**,** b**))**

count **=** cur**.**rowcount

**if** **len(**a **or** b **or** c**)==**0 **or** count**>=**1**:**

messagebox**.**showinfo**(**"Error"**,** "Check Enteries.. \n 1. No field should be empty, \n 2. Duplicate values not

allowed."**)**

**return**

**else:**

**pass**

# Inserting values after meeting requirements

**try:**

query **=** "insert into game values(%s, %s, %s)"

cur**.**execute**(**query**,(**a**,** b**,** c**))**

con**.**commit**()**

cur**.**execute**(**"select \* from game"**)**

result **=** cur**.**fetchall**()**

count **=** cur**.**rowcount

self**.**Games**.**setRowCount**(**count**)**

**for** row\_number**,** row\_data **in** **enumerate(**result**):**

**for** column\_number**,** data **in** **enumerate(**row\_data**):**

self**.**Games**.**setItem**(**row\_number**,** column\_number**,** QtWidgets**.**QTableWidgetItem**(str(**data**)))**

**except:**

**pass**

# SOFTWARE TABLE

**def** add\_4**(**self**):**

a **=** self**.**lineEdit\_11**.**text**()**

b **=** self**.**lineEdit\_12**.**text**()**

c **=** self**.**lineEdit\_13**.**text**()**

# Checking if records already exists already or entered records are empty

data **=** "select \* from software where Softwares=%s and Company=%s"

cur**.**execute**(**data**,(**a**,** b**))**

count **=** cur**.**rowcount

**if** **len(**a **or** b **or** c**)==**0 **or** count**>=**1**:**

messagebox**.**showinfo**(**"Error"**,** "Check Enteries..\n 1. No field should be empty.\n 2. Duplicate values not

allowed."**)**

**return**

**else:**

**pass**

# Inserting values after meeting requirements

**try:**

query **=** "insert into software values(%s, %s, %s)"

cur**.**execute**(**query**,(**a**,** b**,** c**))**

con**.**commit**()**

cur**.**execute**(**"select \* from software"**)**

result **=** cur**.**fetchall**()**

count **=** cur**.**rowcount

self**.**software\_2**.**setRowCount**(**count**)**

**for** row\_number**,** row\_data **in** **enumerate(**result**):**

**for** column\_number**,** data **in** **enumerate(**row\_data**):**

self**.**software\_2**.**setItem**(**row\_number**,** column\_number**,** QtWidgets**.**QTableWidgetItem**(str(**data**)))**

**except:**

**pass**

# Deleting Records from Tables

# MOVIE TABLE

**def** del\_1**(**self**):**

a **=** self**.**lineEdit**.**text**()**

b **=** self**.**lineEdit\_2**.**text**()**

query **=** 'delete from movie where movies = %s and Genre = %s'

cur**.**execute**(**query**,** **(**a**,** b**))**

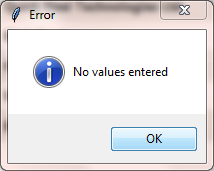
con**.**commit**()**

# Checking if entered records are empty

**if** **len(**a **or** b**)==**0**:**

messagebox**.**showinfo**(**"Error"**,** "No values entered"**)**

**return**



**else:**

**pass**

#Auto Refreshing

cur**.**execute**(**"select \* from movie"**)**

result **=** cur**.**fetchall**()**

count **=** cur**.**rowcount

self**.**Movies**.**setRowCount**(**count**)**

**for** row\_number**,** row\_data **in** **enumerate(**result**):**

**for** column\_number**,** data **in** **enumerate(**row\_data**):**

self**.**Movies**.**setItem**(**row\_number**,** column\_number**,** QtWidgets**.**QTableWidgetItem**(str(**data**)))**

# MUSIC TABLE

**def** del\_2**(**self**):**

a **=** self**.**lineEdit\_5**.**text**()**

b **=** self**.**lineEdit\_6**.**text**()**

query **=** "delete from music where Music = %s and Artists = %s"

cur**.**execute**(**query**,(**a**,** b**))**

con**.**commit**()**

# Checking if entered records are empty

**if** **len(**a **or** b**)==**0**:**

messagebox**.**showinfo**(**"Error"**,** "No values entered"**)**

**return**

**else:**

**pass**

#Auto Refreshing

cur**.**execute**(**"select \* from music"**)**

result **=** cur**.**fetchall**()**

count **=** cur**.**rowcount

self**.**music\_2**.**setRowCount**(**count**)**

**for** row\_number**,** row\_data **in** **enumerate(**result**):**

**for** column\_number**,** data **in** **enumerate(**row\_data**):**

self**.**music\_2**.**setItem**(**row\_number**,** column\_number**,** QtWidgets**.**QTableWidgetItem**(str(**data**)))**

# GAMES TABLE

**def** del\_3**(**self**):**

a **=** self**.**lineEdit\_8**.**text**()**

b **=** self**.**lineEdit\_9**.**text**()**

query **=** "delete from game where Games=%s and Developers = %s"

cur**.**execute**(**query**,(**a**,** b**))**

con**.**commit**()**

# Checking if entered records are empty

**if** **len(**a **or** b**)==**0**:**

messagebox**.**showinfo**(**"Error"**,** "No values entered"**)**

**return**

**else:**

**pass**

#Auto Refreshing

cur**.**execute**(**"select \* from game"**)**

result **=** cur**.**fetchall**()**

count **=** cur**.**rowcount

self**.**Games**.**setRowCount**(**count**)**

**for** row\_number**,** row\_data **in** **enumerate(**result**):**

**for** column\_number**,** data **in** **enumerate(**row\_data**):**

self**.**Games**.**setItem**(**row\_number**,** column\_number**,** QtWidgets**.**QTableWidgetItem**(str(**data**)))**

# SOFTWARE TABLE

**def** del\_4**(**self**):**

a **=** self**.**lineEdit\_11**.**text**()**

b **=** self**.**lineEdit\_12**.**text**()**

query **=** "delete from software where Softwares=%s and Company = %s"

cur**.**execute**(**query**,(**a**,** b**))**

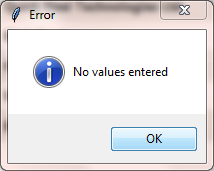
con**.**commit**()**

# Checking if entered records are empty

**if** **len(**a **or** b**)==**0**:**

messagebox**.**showinfo**(**"Error"**,** "No values entered"**)**

**return**



**else:**

**pass**

#Auto Refreshing

cur**.**execute**(**"select \* from software"**)**

result **=** cur**.**fetchall**()**

count **=** cur**.**rowcount

self**.**software\_2**.**setRowCount**(**count**)**

**for** row\_number**,** row\_data **in** **enumerate(**result**):**

**for** column\_number**,** data **in** **enumerate(**row\_data**):**

self**.**software\_2**.**setItem**(**row\_number**,** column\_number**,** QtWidgets**.**QTableWidgetItem**(str(**data**)))**

# Showing Data from MYSQL Database Table to Python Table

**def** show\_data**(**self**):**

# ---- For Customer Table ----

self**.**refresh**.**hide**()**

cur**.**execute**(**"select Username, Address, PhoneNo from cust\_table"**)**

result **=** cur**.**fetchall**()**

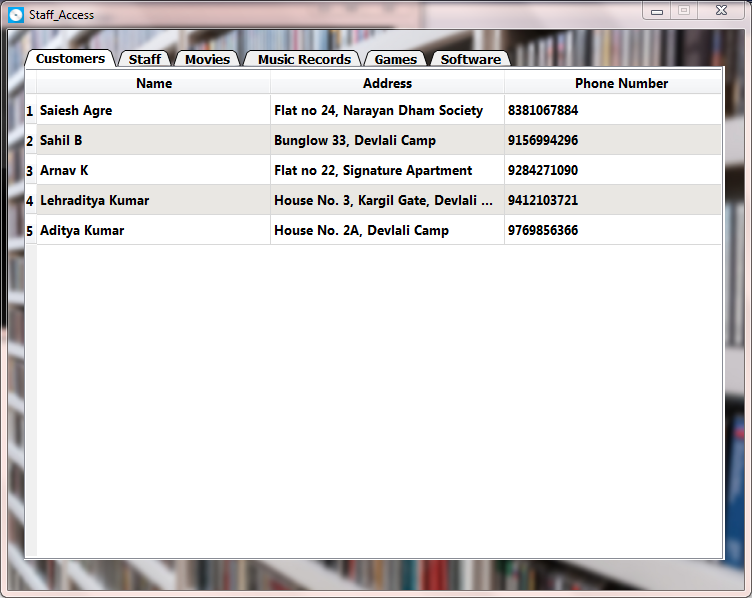
count **=** cur**.**rowcount

self**.**Customer**.**setRowCount**(**count**)**

**for** row\_number**,** row\_data **in** **enumerate(**result**):**

**for** column\_number**,** data **in** **enumerate(**row\_data**):**

self**.**Customer**.**setItem**(**row\_number**,** column\_number**,** QtWidgets**.**QTableWidgetItem**(str(**data**)))**



Mom

# ---- For Staff Table ----

cur**.**execute**(**"select StaffName, Address, PhoneNo from staf\_table"**)**

result **=** cur**.**fetchall**()**

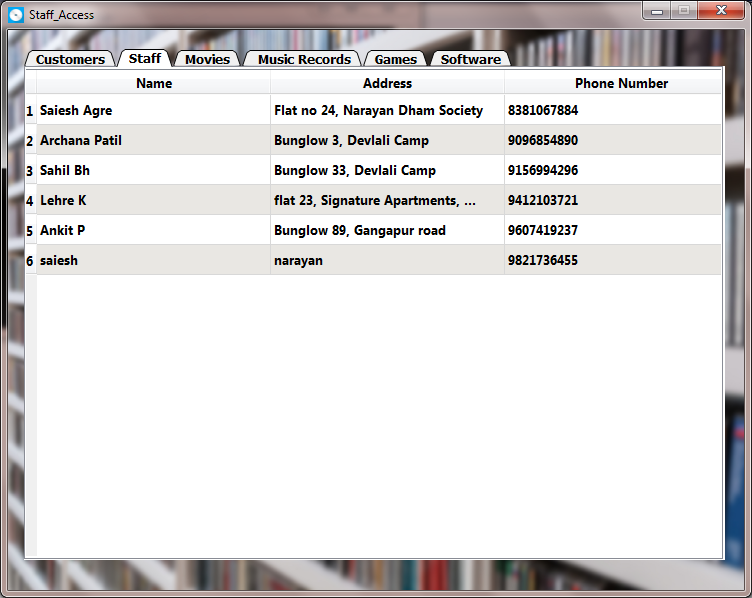
count **=** cur**.**rowcount

self**.**Staff**.**setRowCount**(**count**)**

**for** row\_number**,** row\_data **in** **enumerate(**result**):**

**for** column\_number**,** data **in** **enumerate(**row\_data**):**

self**.**Staff**.**setItem**(**row\_number**,** column\_number**,** QtWidgets**.**QTableWidgetItem**(str(**data**)))**



# ---- For Movie Table ----

cur**.**execute**(**"select \* from movie"**)**

result **=** cur**.**fetchall**()**

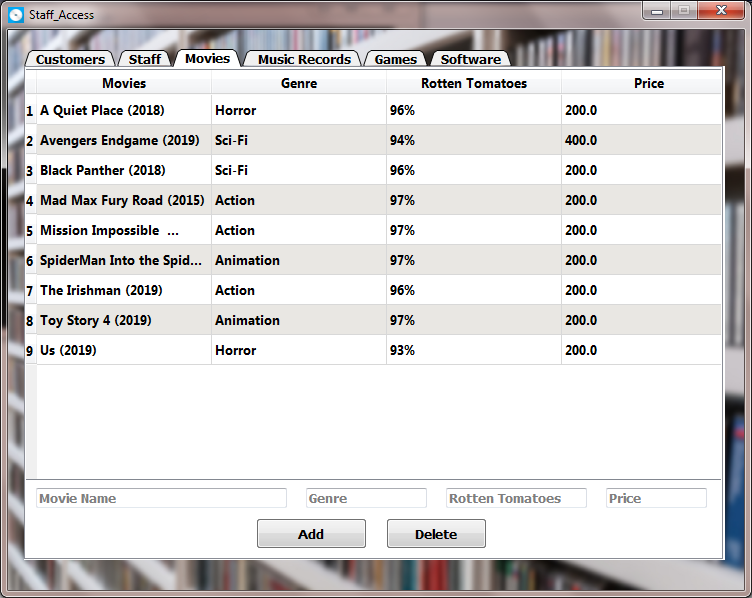
count **=** cur**.**rowcount

self**.**Movies**.**setRowCount**(**count**)**

**for** row\_number**,** row\_data **in** **enumerate(**result**):**

**for** column\_number**,** data **in** **enumerate(**row\_data**):**

self**.**Movies**.**setItem**(**row\_number**,** column\_number**,** QtWidgets**.**QTableWidgetItem**(str(**data**)))**



# ---- For Music Table ----

cur**.**execute**(**"select \* from music"**)**

result **=** cur**.**fetchall**()**

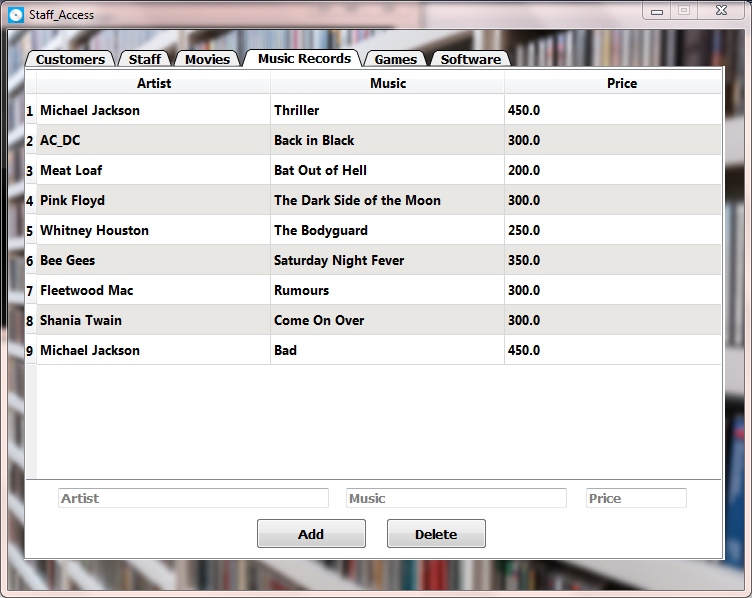
count **=** cur**.**rowcount

self**.**music\_2**.**setRowCount**(**count**)**

**for** row\_number**,** row\_data **in** **enumerate(**result**):**

**for** column\_number**,** data **in** **enumerate(**row\_data**):**

self**.**music\_2**.**setItem**(**row\_number**,** column\_number**,** QtWidgets**.**QTableWidgetItem**(str(**data**)))**



# ---- For Game Table ----

cur**.**execute**(**"select \* from game"**)**

result **=** cur**.**fetchall**()**

count **=** cur**.**rowcount

self**.**Games**.**setRowCount**(**count**)**

**for** row\_number**,** row\_data **in** **enumerate(**result**):**

**for** column\_number**,** data **in** **enumerate(**row\_data**):**

self**.**Games**.**setItem**(**row\_number**,** column\_number**,** QtWidgets**.**QTableWidgetItem**(str(**data**)))**

# ---- For Software Table ----

cur**.**execute**(**"select \* from software"**)**

result **=** cur**.**fetchall**()**

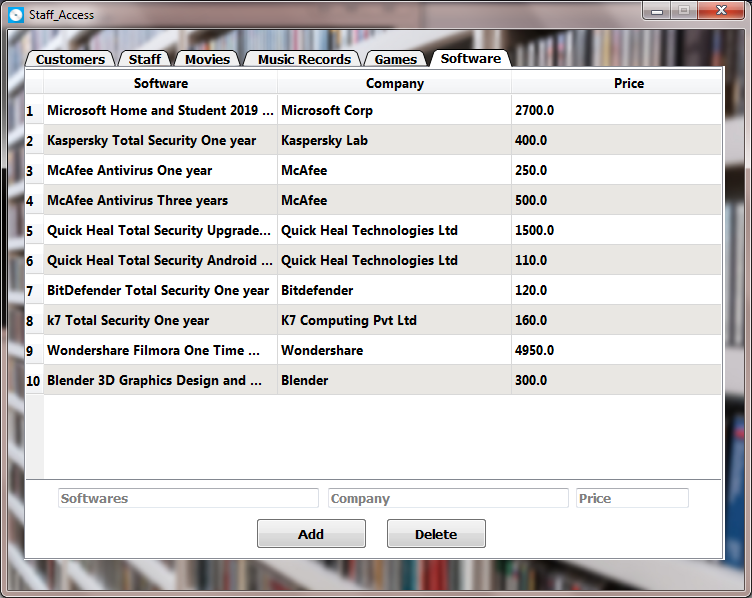
count **=** cur**.**rowcount

self**.**software\_2**.**setRowCount**(**count**)**

**for** row\_number**,** row\_data **in** **enumerate(**result**):**

**for** column\_number**,** data **in** **enumerate(**row\_data**):**

self**.**software\_2**.**setItem**(**row\_number**,** column\_number**,** QtWidgets**.**QTableWidgetItem**(str(**data**)))**



**def** setupUi**(**self**,** Register**):**

Register**.**setObjectName**(**"Register"**)**

Register**.**setWindowModality**(**QtCore**.**Qt**.**WindowModal**)**

Register**.**setFixedSize**(**736**,** 560**)**

icon **=** QtGui**.**QIcon**()**

icon**.**addPixmap**(**QtGui**.**QPixmap**(**"Windows\\Icons\\Mw.png"**),** QtGui**.**QIcon**.**Normal**,** QtGui**.**QIcon**.**Off**)**

Register**.**setWindowIcon**(**icon**)**

self**.**centralwidget **=** QtWidgets**.**QWidget**(**Register**)**

self**.**centralwidget**.**setObjectName**(**"centralwidget"**)**

self**.**label **=** QtWidgets**.**QLabel**(**self**.**centralwidget**)**

self**.**label**.**setGeometry**(**QtCore**.**QRect**(-**4**,** **-**8**,** 741**,** 571**))**

font **=** QtGui**.**QFont**()**

font**.**setFamily**(**"Aharoni"**)**

font**.**setBold**(True)**

font**.**setWeight**(**75**)**

self**.**label**.**setFont**(**font**)**

self**.**label**.**setText**(**""**)**

self**.**label**.**setPixmap**(**QtGui**.**QPixmap**(**"Windows\\Icons\\Bg2.jpg"**))**

self**.**label**.**setScaledContents**(True)**

self**.**label**.**setObjectName**(**"label"**)**

self**.**tab **=** QtWidgets**.**QTabWidget**(**self**.**centralwidget**)**

self**.**tab**.**setGeometry**(**QtCore**.**QRect**(**16**,** 19**,** 701**,** 511**))**

font **=** QtGui**.**QFont**()**

font**.**setPointSize**(**10**)**

font**.**setBold**(True)**

font**.**setWeight**(**75**)**

self**.**tab**.**setFont**(**font**)**

self**.**tab**.**setTabShape**(**QtWidgets**.**QTabWidget**.**Triangular**)**

self**.**tab**.**setObjectName**(**"tab"**)**

self**.**customer **=** QtWidgets**.**QWidget**()**

self**.**customer**.**setObjectName**(**"customer"**)**

self**.**Customer **=** QtWidgets**.**QTableWidget**(**self**.**customer**)**

self**.**Customer**.**setGeometry**(**QtCore**.**QRect**(-**5**,** 1**,** 701**,** 491**))**

self**.**Customer**.**setHorizontalScrollBarPolicy**(**QtCore**.**Qt**.**ScrollBarAlwaysOff**)**

self**.**Customer**.**setEditTriggers**(**QtWidgets**.**QAbstractItemView**.**NoEditTriggers**)**

self**.**Customer**.**setAlternatingRowColors**(True)**

self**.**Customer**.**setSelectionMode**(**QtWidgets**.**QAbstractItemView**.**SingleSelection**)**

self**.**Customer**.**setColumnCount**(**3**)**

self**.**Customer**.**setObjectName**(**"Customer"**)**

self**.**Customer**.**setRowCount**(**0**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**Customer**.**setHorizontalHeaderItem**(**0**,** item**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**Customer**.**setHorizontalHeaderItem**(**1**,** item**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**Customer**.**setHorizontalHeaderItem**(**2**,** item**)**

self**.**Customer**.**horizontalHeader**().**setDefaultSectionSize**(**234**)**

self**.**Customer**.**horizontalHeader**().**setStretchLastSection**(True)**

self**.**tab**.**addTab**(**self**.**customer**,** ""**)**

self**.**staff **=** QtWidgets**.**QWidget**()**

self**.**staff**.**setObjectName**(**"staff"**)**

self**.**Staff **=** QtWidgets**.**QTableWidget**(**self**.**staff**)**

self**.**Staff**.**setGeometry**(**QtCore**.**QRect**(-**5**,** 1**,** 701**,** 491**))**

self**.**Staff**.**setHorizontalScrollBarPolicy**(**QtCore**.**Qt**.**ScrollBarAlwaysOff**)**

self**.**Staff**.**setEditTriggers**(**QtWidgets**.**QAbstractItemView**.**NoEditTriggers**)**

self**.**Staff**.**setAlternatingRowColors**(True)**

self**.**Staff**.**setSelectionMode**(**QtWidgets**.**QAbstractItemView**.**SingleSelection**)**

self**.**Staff**.**setObjectName**(**"Staff"**)**

self**.**Staff**.**setColumnCount**(**3**)**

self**.**Staff**.**setRowCount**(**0**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**Staff**.**setHorizontalHeaderItem**(**0**,** item**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**Staff**.**setHorizontalHeaderItem**(**1**,** item**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**Staff**.**setHorizontalHeaderItem**(**2**,** item**)**

self**.**Staff**.**horizontalHeader**().**setDefaultSectionSize**(**234**)**

self**.**Staff**.**horizontalHeader**().**setStretchLastSection**(True)**

self**.**tab**.**addTab**(**self**.**staff**,** ""**)**

self**.**movies **=** QtWidgets**.**QWidget**()**

self**.**movies**.**setObjectName**(**"movies"**)**

self**.**Movies **=** QtWidgets**.**QTableWidget**(**self**.**movies**)**

self**.**Movies**.**setGeometry**(**QtCore**.**QRect**(-**5**,** 1**,** 701**,** 411**))**

self**.**Movies**.**setHorizontalScrollBarPolicy**(**QtCore**.**Qt**.**ScrollBarAlwaysOff**)**

self**.**Movies**.**setEditTriggers**(**QtWidgets**.**QAbstractItemView**.**NoEditTriggers**)**

self**.**Movies**.**setDragEnabled**(True)**

self**.**Movies**.**setAlternatingRowColors**(True)**

self**.**Movies**.**setSelectionMode**(**QtWidgets**.**QAbstractItemView**.**SingleSelection**)**

self**.**Movies**.**setObjectName**(**"Movies"**)**

self**.**Movies**.**setColumnCount**(**4**)**

self**.**Movies**.**setRowCount**(**0**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**Movies**.**setHorizontalHeaderItem**(**0**,** item**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**Movies**.**setHorizontalHeaderItem**(**1**,** item**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**Movies**.**setHorizontalHeaderItem**(**2**,** item**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**Movies**.**setHorizontalHeaderItem**(**3**,** item**)**

self**.**Movies**.**horizontalHeader**().**setDefaultSectionSize**(**175**)**

self**.**Movies**.**horizontalHeader**().**setStretchLastSection**(True)**

self**.**lineEdit **=** QtWidgets**.**QLineEdit**(**self**.**movies**)**

self**.**lineEdit**.**setGeometry**(**QtCore**.**QRect**(**10**,** 420**,** 251**,** 20**))**

self**.**lineEdit**.**setObjectName**(**"lineEdit"**)**

self**.**lineEdit\_2 **=** QtWidgets**.**QLineEdit**(**self**.**movies**)**

self**.**lineEdit\_2**.**setGeometry**(**QtCore**.**QRect**(**280**,** 420**,** 121**,** 20**))**

self**.**lineEdit\_2**.**setObjectName**(**"lineEdit\_2"**)**

self**.**lineEdit\_3 **=** QtWidgets**.**QLineEdit**(**self**.**movies**)**

self**.**lineEdit\_3**.**setGeometry**(**QtCore**.**QRect**(**420**,** 420**,** 141**,** 20**))**

self**.**lineEdit\_3**.**setObjectName**(**"lineEdit\_3"**)**

self**.**lineEdit\_4 **=** QtWidgets**.**QLineEdit**(**self**.**movies**)**

self**.**lineEdit\_4**.**setGeometry**(**QtCore**.**QRect**(**580**,** 420**,** 101**,** 20**))**

font **=** QtGui**.**QFont**()**

font**.**setBold**(True)**

font**.**setWeight**(**75**)**

self**.**lineEdit\_4**.**setFont**(**font**)**

self**.**lineEdit\_4**.**setStyleSheet**(**""**)**

self**.**lineEdit\_4**.**setObjectName**(**"lineEdit\_4"**)**

self**.**pushButton\_1 **=** QtWidgets**.**QPushButton**(**self**.**movies**)**

self**.**pushButton\_1**.**setGeometry**(**QtCore**.**QRect**(**230**,** 450**,** 111**,** 31**))**

self**.**pushButton\_1**.**clicked**.**connect**(**self**.**add\_1**)** # Click Signal

self**.**pushButton\_1**.**setObjectName**(**"pushButton\_1"**)**

self**.**pushButton\_2 **=** QtWidgets**.**QPushButton**(**self**.**movies**)**

self**.**pushButton\_2**.**setGeometry**(**QtCore**.**QRect**(**360**,** 450**,** 101**,** 31**))**

self**.**pushButton\_2**.**clicked**.**connect**(**self**.**del\_1**)** # Click Signal

self**.**pushButton\_2**.**setObjectName**(**"pushButton\_2"**)**

self**.**tab**.**addTab**(**self**.**movies**,** ""**)**

self**.**music **=** QtWidgets**.**QWidget**()**

self**.**music**.**setObjectName**(**"music"**)**

self**.**music\_2 **=** QtWidgets**.**QTableWidget**(**self**.**music**)**

self**.**music\_2**.**setGeometry**(**QtCore**.**QRect**(-**5**,** 1**,** 701**,** 411**))**

self**.**music\_2**.**setHorizontalScrollBarPolicy**(**QtCore**.**Qt**.**ScrollBarAlwaysOff**)**

self**.**music\_2**.**setEditTriggers**(**QtWidgets**.**QAbstractItemView**.**NoEditTriggers**)**

self**.**music\_2**.**setDragEnabled**(True)**

self**.**music\_2**.**setAlternatingRowColors**(True)**

self**.**music\_2**.**setSelectionMode**(**QtWidgets**.**QAbstractItemView**.**SingleSelection**)**

self**.**music\_2**.**setObjectName**(**"music\_2"**)**

self**.**music\_2**.**setColumnCount**(**3**)**

self**.**music\_2**.**setRowCount**(**0**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**music\_2**.**setHorizontalHeaderItem**(**0**,** item**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**music\_2**.**setHorizontalHeaderItem**(**1**,** item**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**music\_2**.**setHorizontalHeaderItem**(**2**,** item**)**

self**.**music\_2**.**horizontalHeader**().**setDefaultSectionSize**(**234**)**

self**.**music\_2**.**horizontalHeader**().**setStretchLastSection**(True)**

self**.**lineEdit\_5 **=** QtWidgets**.**QLineEdit**(**self**.**music**)**

self**.**lineEdit\_5**.**setGeometry**(**QtCore**.**QRect**(**32**,** 420**,** 271**,** 20**))**

self**.**lineEdit\_5**.**setObjectName**(**"lineEdit\_5"**)**

self**.**lineEdit\_6 **=** QtWidgets**.**QLineEdit**(**self**.**music**)**

self**.**lineEdit\_6**.**setGeometry**(**QtCore**.**QRect**(**320**,** 420**,** 221**,** 20**))**

self**.**lineEdit\_6**.**setObjectName**(**"lineEdit\_6"**)**

self**.**lineEdit\_7 **=** QtWidgets**.**QLineEdit**(**self**.**music**)**

self**.**lineEdit\_7**.**setGeometry**(**QtCore**.**QRect**(**560**,** 420**,** 101**,** 20**))**

self**.**lineEdit\_7**.**setObjectName**(**"lineEdit\_7"**)**

self**.**pushButton\_3 **=** QtWidgets**.**QPushButton**(**self**.**music**)**

self**.**pushButton\_3**.**setGeometry**(**QtCore**.**QRect**(**230**,** 450**,** 111**,** 31**))**

self**.**pushButton\_3**.**clicked**.**connect**(**self**.**add\_2**)** # Click Signal

self**.**pushButton\_3**.**setObjectName**(**"pushButton\_3"**)**

self**.**pushButton\_4 **=** QtWidgets**.**QPushButton**(**self**.**music**)**

self**.**pushButton\_4**.**setGeometry**(**QtCore**.**QRect**(**360**,** 450**,** 101**,** 31**))**

self**.**pushButton\_4**.**clicked**.**connect**(**self**.**del\_2**)** # Click Signal

self**.**pushButton\_4**.**setObjectName**(**"pushButton\_4"**)**

self**.**tab**.**addTab**(**self**.**music**,** ""**)**

self**.**game **=** QtWidgets**.**QWidget**()**

self**.**game**.**setObjectName**(**"game"**)**

self**.**Games **=** QtWidgets**.**QTableWidget**(**self**.**game**)**

self**.**Games**.**setGeometry**(**QtCore**.**QRect**(-**5**,** 1**,** 701**,** 411**))**

self**.**Games**.**setHorizontalScrollBarPolicy**(**QtCore**.**Qt**.**ScrollBarAlwaysOff**)**

self**.**Games**.**setEditTriggers**(**QtWidgets**.**QAbstractItemView**.**NoEditTriggers**)**

self**.**Games**.**setDragEnabled**(True)**

self**.**Games**.**setAlternatingRowColors**(True)**

self**.**Games**.**setSelectionMode**(**QtWidgets**.**QAbstractItemView**.**SingleSelection**)**

self**.**Games**.**setObjectName**(**"Games"**)**

self**.**Games**.**setColumnCount**(**3**)**

self**.**Games**.**setRowCount**(**0**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**Games**.**setHorizontalHeaderItem**(**0**,** item**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**Games**.**setHorizontalHeaderItem**(**1**,** item**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**Games**.**setHorizontalHeaderItem**(**2**,** item**)**

self**.**Games**.**horizontalHeader**().**setDefaultSectionSize**(**234**)**

self**.**Games**.**horizontalHeader**().**setStretchLastSection**(True)**

self**.**lineEdit\_8 **=** QtWidgets**.**QLineEdit**(**self**.**game**)**

self**.**lineEdit\_8**.**setGeometry**(**QtCore**.**QRect**(**30**,** 420**,** 251**,** 20**))**

self**.**lineEdit\_8**.**setObjectName**(**"lineEdit\_8"**)**

self**.**lineEdit\_9 **=** QtWidgets**.**QLineEdit**(**self**.**game**)**

self**.**lineEdit\_9**.**setGeometry**(**QtCore**.**QRect**(**300**,** 420**,** 231**,** 20**))**

self**.**lineEdit\_9**.**setObjectName**(**"lineEdit\_9"**)**

self**.**lineEdit\_10 **=** QtWidgets**.**QLineEdit**(**self**.**game**)**

self**.**lineEdit\_10**.**setGeometry**(**QtCore**.**QRect**(**550**,** 420**,** 113**,** 20**))**

self**.**lineEdit\_10**.**setObjectName**(**"lineEdit\_10"**)**

self**.**pushButton\_5 **=** QtWidgets**.**QPushButton**(**self**.**game**)**

self**.**pushButton\_5**.**setGeometry**(**QtCore**.**QRect**(**230**,** 450**,** 111**,** 31**))**

self**.**pushButton\_5**.**clicked**.**connect**(**self**.**add\_3**)** # Click Signal

self**.**pushButton\_5**.**setObjectName**(**"pushButton\_5"**)**

self**.**pushButton\_6 **=** QtWidgets**.**QPushButton**(**self**.**game**)**

self**.**pushButton\_6**.**setGeometry**(**QtCore**.**QRect**(**360**,** 450**,** 101**,** 31**))**

self**.**pushButton\_6**.**clicked**.**connect**(**self**.**del\_3**)** # Click Signal

self**.**pushButton\_6**.**setObjectName**(**"pushButton\_6"**)**

self**.**tab**.**addTab**(**self**.**game**,** ""**)**

self**.**software **=** QtWidgets**.**QWidget**()**

self**.**software**.**setObjectName**(**"software"**)**

self**.**software\_2 **=** QtWidgets**.**QTableWidget**(**self**.**software**)**

self**.**software\_2**.**setGeometry**(**QtCore**.**QRect**(-**5**,** 1**,** 701**,** 411**))**

self**.**software\_2**.**setHorizontalScrollBarPolicy**(**QtCore**.**Qt**.**ScrollBarAlwaysOff**)**

self**.**software\_2**.**setEditTriggers**(**QtWidgets**.**QAbstractItemView**.**NoEditTriggers**)**

self**.**software\_2**.**setDragEnabled**(True)**

self**.**software\_2**.**setAlternatingRowColors**(True)**

self**.**software\_2**.**setSelectionMode**(**QtWidgets**.**QAbstractItemView**.**SingleSelection**)**

self**.**software\_2**.**setObjectName**(**"software\_2"**)**

self**.**software\_2**.**setColumnCount**(**3**)**

self**.**software\_2**.**setRowCount**(**0**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**software\_2**.**setHorizontalHeaderItem**(**0**,** item**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**software\_2**.**setHorizontalHeaderItem**(**1**,** item**)**

item **=** QtWidgets**.**QTableWidgetItem**()**

self**.**software\_2**.**setHorizontalHeaderItem**(**2**,** item**)**

self**.**software\_2**.**horizontalHeader**().**setDefaultSectionSize**(**234**)**

self**.**software\_2**.**horizontalHeader**().**setStretchLastSection**(True)**

self**.**lineEdit\_11 **=** QtWidgets**.**QLineEdit**(**self**.**software**)**

self**.**lineEdit\_11**.**setGeometry**(**QtCore**.**QRect**(**32**,** 420**,** 261**,** 20**))**

self**.**lineEdit\_11**.**setObjectName**(**"lineEdit\_11"**)**

self**.**lineEdit\_12 **=** QtWidgets**.**QLineEdit**(**self**.**software**)**

self**.**lineEdit\_12**.**setGeometry**(**QtCore**.**QRect**(**302**,** 420**,** 241**,** 20**))**

self**.**lineEdit\_12**.**setObjectName**(**"lineEdit\_12"**)**

self**.**lineEdit\_13 **=** QtWidgets**.**QLineEdit**(**self**.**software**)**

self**.**lineEdit\_13**.**setGeometry**(**QtCore**.**QRect**(**550**,** 420**,** 113**,** 20**))**

self**.**lineEdit\_13**.**setObjectName**(**"lineEdit\_13"**)**

self**.**pushButton\_7 **=** QtWidgets**.**QPushButton**(**self**.**software**)**

self**.**pushButton\_7**.**setGeometry**(**QtCore**.**QRect**(**230**,** 450**,** 111**,** 31**))**

self**.**pushButton\_7**.**clicked**.**connect**(**self**.**add\_4**)** # Click Signal

self**.**pushButton\_7**.**setObjectName**(**"pushButton\_9"**)**

self**.**pushButton\_8 **=** QtWidgets**.**QPushButton**(**self**.**software**)**

self**.**pushButton\_8**.**setGeometry**(**QtCore**.**QRect**(**360**,** 450**,** 101**,** 31**))**

self**.**pushButton\_8**.**clicked**.**connect**(**self**.**del\_4**)** # Click Signal

self**.**pushButton\_8**.**setObjectName**(**"pushButton\_8"**)**

self**.**tab**.**addTab**(**self**.**software**,** ""**)**

self**.**refresh **=** QtWidgets**.**QPushButton**(**self**.**centralwidget**)**

self**.**refresh**.**setGeometry**(**QtCore**.**QRect**(**250**,** 530**,** 231**,** 31**))**

self**.**refresh**.**clicked**.**connect**(**self**.**show\_data**)**# Click Signal

font **=** QtGui**.**QFont**()**

font**.**setFamily**(**"Corbel"**)**

font**.**setPointSize**(**16**)**

font**.**setBold**(True)**

font**.**setWeight**(**75**)**

self**.**refresh**.**setFont**(**font**)**

self**.**refresh**.**setFocusPolicy**(**QtCore**.**Qt**.**ClickFocus**)**

self**.**refresh**.**setStyleSheet**(**'''background-color: qlineargradient(spread:pad, x1:0, y1:0, x2:1, y2:0,

stop:0 rgba(255, 216, 0, 255), stop:1 rgba(255, 255, 255, 255));'''**)**

self**.**refresh**.**setObjectName**(**"refresh"**)**

Register**.**setCentralWidget**(**self**.**centralwidget**)**

self**.**retranslateUi**(**Register**)**

self**.**tab**.**setCurrentIndex**(**0**)**

QtCore**.**QMetaObject**.**connectSlotsByName**(**Register**)**

**def** retranslateUi**(**self**,** Register**):**

\_translate **=** QtCore**.**QCoreApplication**.**translate

Register**.**setWindowTitle**(**\_translate**(**"Register"**,** "Staff\_Access"**))**

self**.**Customer**.**setSortingEnabled**(True)**

item **=** self**.**Customer**.**horizontalHeaderItem**(**0**)**

item**.**setText**(**\_translate**(**"Register"**,** "Name"**))**

item **=** self**.**Customer**.**horizontalHeaderItem**(**1**)**

item**.**setText**(**\_translate**(**"Register"**,** "Address"**))**

item **=** self**.**Customer**.**horizontalHeaderItem**(**2**)**

item**.**setText**(**\_translate**(**"Register"**,** "Phone Number"**))**

self**.**tab**.**setTabText**(**self**.**tab**.**indexOf**(**self**.**customer**),** \_translate**(**"Register"**,** "Customers"**))**

self**.**Staff**.**setSortingEnabled**(True)**

item **=** self**.**Staff**.**horizontalHeaderItem**(**0**)**

item**.**setText**(**\_translate**(**"Register"**,** "Name"**))**

item **=** self**.**Staff**.**horizontalHeaderItem**(**1**)**

item**.**setText**(**\_translate**(**"Register"**,** "Address"**))**

item **=** self**.**Staff**.**horizontalHeaderItem**(**2**)**

item**.**setText**(**\_translate**(**"Register"**,** "Phone Number"**))**

self**.**tab**.**setTabText**(**self**.**tab**.**indexOf**(**self**.**staff**),** \_translate**(**"Register"**,** "Staff"**))**

self**.**Movies**.**setSortingEnabled**(True)**

item **=** self**.**Movies**.**horizontalHeaderItem**(**0**)**

item**.**setText**(**\_translate**(**"Register"**,** "Movies"**))**

item **=** self**.**Movies**.**horizontalHeaderItem**(**1**)**

item**.**setText**(**\_translate**(**"Register"**,** "Genre"**))**

item **=** self**.**Movies**.**horizontalHeaderItem**(**2**)**

item**.**setText**(**\_translate**(**"Register"**,** "Rotten Tomatoes"**))**

item **=** self**.**Movies**.**horizontalHeaderItem**(**3**)**

item**.**setText**(**\_translate**(**"Register"**,** "Price"**))**

self**.**lineEdit**.**setPlaceholderText**(**\_translate**(**"Register"**,** "Movie Name"**))**

self**.**lineEdit\_2**.**setPlaceholderText**(**\_translate**(**"Register"**,** "Genre"**))**

self**.**lineEdit\_3**.**setPlaceholderText**(**\_translate**(**"Register"**,** "Rotten Tomatoes"**))**

self**.**lineEdit\_4**.**setPlaceholderText**(**\_translate**(**"Register"**,** "Price"**))**

self**.**pushButton\_1**.**setText**(**\_translate**(**"Register"**,** "Add"**))**

self**.**pushButton\_2**.**setText**(**\_translate**(**"Register"**,** "Delete"**))**

self**.**tab**.**setTabText**(**self**.**tab**.**indexOf**(**self**.**movies**),** \_translate**(**"Register"**,** "Movies"**))**

self**.**music\_2**.**setSortingEnabled**(True)**

item **=** self**.**music\_2**.**horizontalHeaderItem**(**0**)**

item**.**setText**(**\_translate**(**"Register"**,** "Artist"**))**

item **=** self**.**music\_2**.**horizontalHeaderItem**(**1**)**

item**.**setText**(**\_translate**(**"Register"**,** "Music"**))**

item **=** self**.**music\_2**.**horizontalHeaderItem**(**2**)**

item**.**setText**(**\_translate**(**"Register"**,** "Price"**))**

self**.**lineEdit\_5**.**setPlaceholderText**(**\_translate**(**"Register"**,** "Artist"**))**

self**.**lineEdit\_6**.**setPlaceholderText**(**\_translate**(**"Register"**,** "Music"**))**

self**.**lineEdit\_7**.**setPlaceholderText**(**\_translate**(**"Register"**,** "Price"**))**

self**.**pushButton\_3**.**setText**(**\_translate**(**"Register"**,** "Add"**))**

self**.**pushButton\_4**.**setText**(**\_translate**(**"Register"**,** "Delete"**))**

self**.**tab**.**setTabText**(**self**.**tab**.**indexOf**(**self**.**music**),** \_translate**(**"Register"**,** " Music Records"**))**

self**.**Games**.**setSortingEnabled**(True)**

item **=** self**.**Games**.**horizontalHeaderItem**(**0**)**

item**.**setText**(**\_translate**(**"Register"**,** " Games"**))**

item **=** self**.**Games**.**horizontalHeaderItem**(**1**)**

item**.**setText**(**\_translate**(**"Register"**,** "Developer"**))**

item **=** self**.**Games**.**horizontalHeaderItem**(**2**)**

item**.**setText**(**\_translate**(**"Register"**,** "Price"**))**

self**.**pushButton\_5**.**setText**(**\_translate**(**"Register"**,** "Add"**))**

self**.**pushButton\_6**.**setText**(**\_translate**(**"Register"**,** "Delete"**))**

self**.**tab**.**setTabText**(**self**.**tab**.**indexOf**(**self**.**game**),** \_translate**(**"Register"**,** "Games"**))**

self**.**software\_2**.**setSortingEnabled**(True)**

item **=** self**.**software\_2**.**horizontalHeaderItem**(**0**)**

item**.**setText**(**\_translate**(**"Register"**,** "Software"**))**

item **=** self**.**software\_2**.**horizontalHeaderItem**(**1**)**

item**.**setText**(**\_translate**(**"Register"**,** "Company"**))**

item **=** self**.**software\_2**.**horizontalHeaderItem**(**2**)**

item**.**setText**(**\_translate**(**"Register"**,** "Price"**))**

self**.**pushButton\_7**.**setText**(**\_translate**(**"Register"**,** "Add"**))**

self**.**pushButton\_8**.**setText**(**\_translate**(**"Register"**,** "Delete"**))**

self**.**tab**.**setTabText**(**self**.**tab**.**indexOf**(**self**.**software**),** \_translate**(**"Register"**,** "Software"**))**

self**.**refresh**.**setText**(**\_translate**(**"Register"**,** "Refresh"**))**

self**.**lineEdit\_8**.**setPlaceholderText**(**\_translate**(**"Register"**,** "Game Name"**))**

self**.**lineEdit\_9**.**setPlaceholderText**(**\_translate**(**"Register"**,** "Developers"**))**

self**.**lineEdit\_10**.**setPlaceholderText**(**\_translate**(**"Register"**,** "Price"**))**

self**.**lineEdit\_11**.**setPlaceholderText**(**\_translate**(**"Register"**,** "Softwares"**))**

self**.**lineEdit\_12**.**setPlaceholderText**(**\_translate**(**"Register"**,** "Company"**))**

self**.**lineEdit\_13**.**setPlaceholderText**(**\_translate**(**"Register"**,** "Price"**))**

**if** \_\_name\_\_ **==** "\_\_main\_\_"**:**

**import** sys

app **=** QtWidgets**.**QApplication**(**sys**.**argv**)**

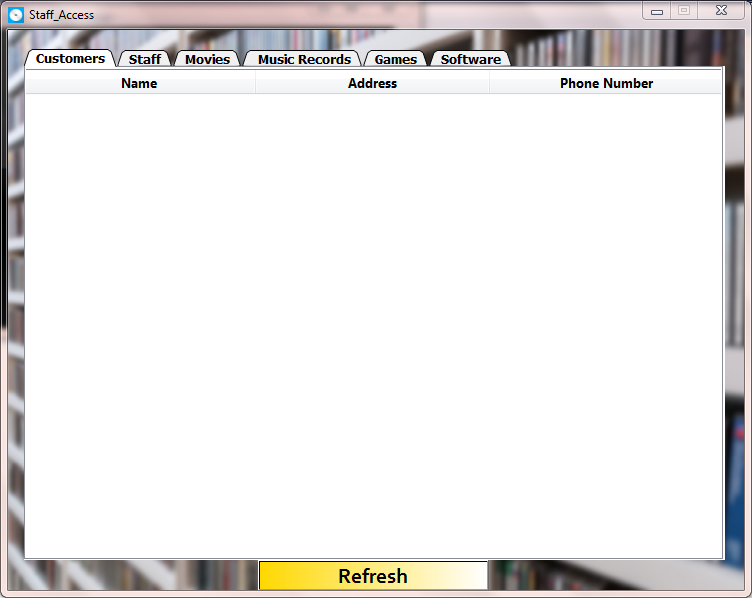
Register **=** QtWidgets**.**QMainWindow**()**

ui **=** Ui\_Register**()**

ui**.**setupUi**(**Register**)**

Register**.**show**()**

sys**.exit(**app**.**exec\_**())**

****

**Bibliography**

* **Reference Books**
* Informatics Practices by Sumita Arora (Class XI)
* Informatics Practices by Sumita Arora (Class XII)
* **Reference Websites**
* [www.geeksforgeeks.org](http://www.geeksforgeeks.org)
* [www.stackoverflow.com](http://www.stackoverflow.com)
* [www.w3schools.com](http://www.w3schools.com)
* [www.tutorialspoint.com](http://www.tutorialspoint.com)
* [www.github.com](http://www.github.com)
* [www.programiz.com](http://www.programiz.com)
* [www.twilio.com](http://www.twilio.com)