Industrial Training Report

On

Python With SQL

Submitted in partial fulfilment of requirements for the award of the Degree of

Bachelor of Technology

In

Computer Science & Engineering

Submitted By

Devanshi Tegwal

(Enrollment Number:02211502720)

Under the guidance of

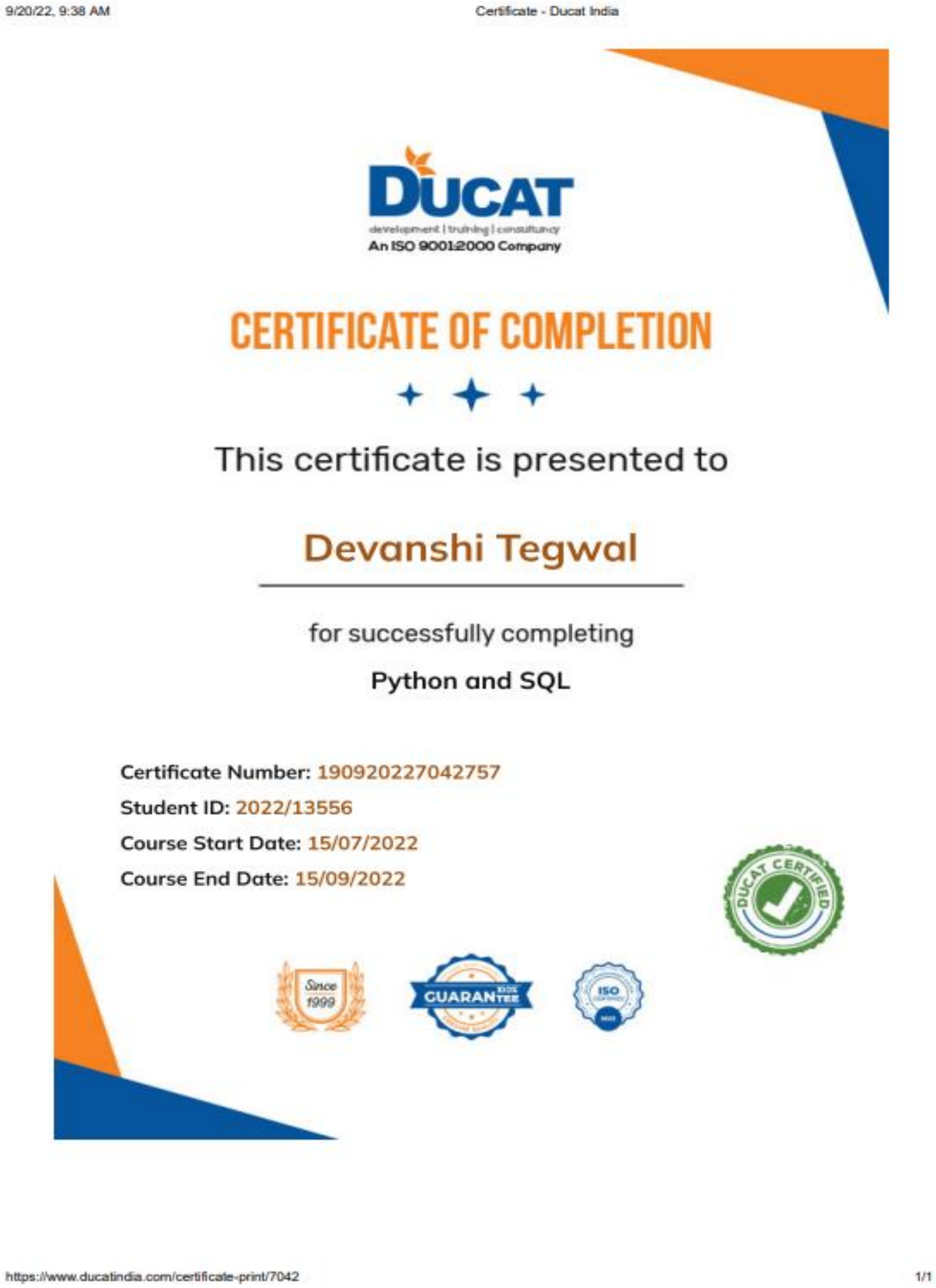
Mr. Tanay Kishore Mishra

(Product Manager, Infotech Pvt Ltd)

Department of Computer Science and Engineering

Bharati Vidyapeeth's College of Engineering, New Delhi – 110063, INDIA

September 2022



9/20/22, 9:38 AM Certificate - Ducat India

DUCAT

development | training | consultancy

An ISO 9001:2000 Company

CERTIFICATE OF COMPLETION

This certificate is presented to

Devanshi Tegwal

for successfully completing

Python and SQL

Certificate Number: 190920227042757

Student ID: 2022/13556

Course Start Date: 15/07/2022

Course End Date: 15/09/2022

DUCAT CERTIFIED

Since 1999

100% GUARANTEE

ISO CERTIFIED

https://www.ducatindia.com/certificate-print/7042 1/1

CANDIDATE’S DECLARATION

It is hereby certified that the work which is being presented in the B. Tech Industrial/In-house training Report entitled **"Python with SQL"** in partial fulfilment of the requirements for the award of the degree of **Bachelor of Technology** and submitted in the **Department of Computer Science & Engineering** of **BHARATI VIDYAPEETH’S COLLEGE OF ENGINEERING, New Delhi (Affiliated to Guru Gobind Singh Indraprastha University, Delhi)** is an authentic record of my own work carried out during a period from **July – September 2022** under the guidance of **Mr. Tanay Kishore Mishra, Product Manager, Infotech Pvt Ltd.**

The matter presented in the B. Tech Industrial/In-house training Report has not been submitted by me for the award of any other degree of this or any other Institute.

Student Name: Devanshi Tegwal
Enrollment No: 02211502720

This is to certify that the above statement made by the candidate is correct to the best of my knowledge. He/She/They are permitted to appear in the External Industrial/In-house training Examination.

Trainer Name	**Mr. Tanay Kishore Mishra**	**Dr. Pranav Dass**
Designation	**Product Manager**	**Head, CSE**
Company name	**Infotech Pvt Ltd**	

The B. Tech Industrial/In-house training Viva-Voce Examination of **Student name (Enrollment No.:………)** has been held on **……………………………….**

Industrial/In-house training Coordinator **(Signature of External Examiner)**

ACKNOWLEDGEMENT

The successful completion of this training marks the beginning of an ever going learning experience of converting ideas and concepts into real life and practical systems. This training was quite a learning experience for us at each step. At the same time, it has given us confidence to work in a professional setup. We feel the experience gained during the training will lead us to gain the bright prospect in the future.

I express my deep gratitude to **Mr. Tanay Kishore Mishra, Product Manager, Infotech Pvt Ltd**, for his/her valuable guidance and suggestion throughout my training. We are also thankful to Dr. Deepika Varshney, Assistant Professor, Department of Computer Science and Engineering, BVCOE, for their valuable guidance.

Sign (Name of the Student)
Enrolment No. 02211502720

Company Profile

DUCAT is one of the leading industry-based training institutes in India, and it provides the students with the knowledge and hands-on training for the computers and IT-based fields. Started in the year 1999, DUCAT is a private institute that has been creating professionals for big MNC's and providing them with the job-specific workforce.

The main corporate office is in Noida, and the courses they offer include IT and Software, Animation, Business, and Management studies streams.

They provide a practical approach for all the training modules and a student gains an in-depth knowledge of the subject and also learn complex programming codes and structures more simply.

The major IT courses offered by the institute are - Language Programming, CCNA, MCSE, MCITP, and CCNP.

DUCAT provides three major types of services like software development for large and small organizations alike, on-campus training to the students who enrol in their technical courses. They also conduct several workshops and provide many placement services for students who are willing to work in the IT field in a well-reputed firm with significant career opportunities.

This institute has been in collaboration with global leaders such as HCL, Birla soft, Cognizant, Ericsson, Capgemini, and NIC which are all billion-dollar MNC's which have a vast reach in its customers and IT developmental tools. This has led to making DUCAT a well-known name in the IT industry for both hands-on training and B2B transactions.

CONTENTS

TECHNOLOGY LEARNT

Name	Sign	Description
MySQL	MySQL	MySQL is a tool used to manage databases and servers, so while it's not a database, it's widely used in relation to managing and organising data in databases.
Python		Python is commonly used for developing websites and software, task automation, data analysis, and data visualization.

.

1. Introduction

• Python

Python is a general purpose, dynamic, high level and interpreted programming language. It supports Object Oriented programming approach to develop applications. It is simple and easy to learn and provides lots of high-level data structures. Python is easy to learn yet powerful and versatile scripting language which makes it attractive for Application Development. Python’s syntax and dynamic typing with its interpreted nature, makes it an ideal language for scripting and rapid application development. Python supports multiple programming pattern, including object oriented, imperative and functional or procedural programming styles intended to work on special area such as web programming. That is why it is known as multipurpose because it can be used with web, enterprise, 3D CAD etc.

It is known for its general purpose nature that makes it applicable in almost each domain of software development. Python as a whole can be used in any sphere of development.

Here, we are specifying applications areas with their frameworks or libraries for which python can be used.

1. Web Applications

We can use Python to develop web applications. It provides libraries to handle internet protocols such as HTML and XML, JSON, Email processing, request, beautiful Soup, Feed parser etc. It also provides Frameworks such as Django, Pyramid, Flask etc. to design and develop web based applications. Some important developments are: Python Wiki Engines, Pocoo, Python Blog Software etc.

2. Desktop GUI Applications

Python provides Tk GUI library to develop user interface in python based application. Some other useful toolkits wxWidgets, Kivy, pyqt that are useable on several platforms. The Kivy is popular for writing multitouch applications.

3. Software Development

Python is helpful for software development process. It works as a support language and can be used for build control and management, testing etc.

4. Scientific and Numeric

Python is popular and widely used in scientific and numeric computing. Some useful library and package are SciPy, Pandas, IPython etc. SciPy is group of packages of engineering, science and mathematics.

5. Business Applications

Python is used to build Business applications like ERP and e-commerce systems. Tryton is a high level application platform.

6. Audio or Video based Applications

Python is awesome to perform multiple tasks and can be used to develop multimedia applications. Some of real applications are: TimPlayer, cplay etc.

NEED OF FRAMEWORK FOR PYTHON

Python is an open-source object-oriented programming language. Python is one of the most popular programming languages in the world. It has a well-organized source code, and it is used for web applications and numerous other things.

Python is also easy to interface with frameworks. There are several frameworks that allow for easier customization and less coding. They help developers get the job done faster. With these frameworks, games, apps, educational programs, and other software can be built faster and better.

1. CherryPy

CherryPy is a minimalist Python web framework. It works on the most recent versions of Python, and it even runs perfectly on Android. It has been in use for almost ten years. CherryPy allows developers to run various HTTP servers at the same time and it allows for profiling. It also has a plugin system that is flexible. Appoptics is one of the best way to check the application performance score.

2. Flask

Flask is a framework that is highly suitable for developers that want to make a standalone app. Flask comes with the Jinja template engine by default, but developers can choose any template engine or ORM that is perfect for them. Flask is very useful for writing APIs, endpoints, or RESTful services. Flask is considered open-ended to unique systems. Developers can create backend systems any way they want, but it was designed for more open-ended applications.

3. Django

Django is the most popular Python framework. It is an open-source framework that makes things very fast and scalable. The framework is regularly updated to match the latest versions of Python. Django supports a lot of database engines and is used by popular websites like Instagram, Pinterest, The Washington Post and others. Popular comment internet application, Disqus was made using Django. Django is easy to pick up for beginners. The first release of Django was in 2006, and it has been expanding since then. Django can be used to build almost any kind of website from social networks to CMS to wikis to news sites. With Django, you can deliver content in JSON, RSS feeds XML, HTML, and almost any other format.

4. Pyramid

Pyramid is a Python framework that supports authentication and routing. It is great for developing large web applications, like CMSs, and it is useful for prototyping a concept and for developers working on API projects. Pyramid is flexible and can be used for both simple and complex projects.

5. TurboGears

TurboGears is an innovative Python framework. It is a single file app that can be run on Python and also scaled to a full stack solution. It has a flexible ORM with multi-database support. It can be used for all forms of simple and advanced projects. The documentation is simple and easy to use. With TurboGears, you can create a ready-to-extend application in mere minutes.

6. Pylons

Pylons is a framework that makes the tracking of errors easy thanks to its online debugger. It is an open source framework supported by a lot of talented and experienced developers.

7. Web2py

Web2py is one of the most portable, flexible, and easy to use Python frameworks available. It doesn’t require any installation or configuration and it uses LDAP for its authentication system. Web2py can be run off a simple USB flash drive. It supports various database engines, and it has a built-in ticketing system to help it manage errors. Web2py doesn't support older versions of Python.

Famous Python Libraries

1. TensorFlow

If you are currently working on a machine learning project in Python, then you may have heard about this popular open source library known as TensorFlow.

This library was developed by Google in collaboration with Brain Team. TensorFlow is a part of almost every Google application for machine learning.

TensorFlow works like a computational library for writing new algorithms that involve a large number of tensor operations, since neural networks can be easily expressed as computational graphs they can be implemented using TensorFlow as a series of operations on Tensors. Plus, tensors are N-dimensional matrices which represent your data.

2. Scikit-Learn

It is a Python library is associated with NumPy and SciPy. It is considered as one of the best libraries for working with complex data.

There are a lot of changes being made in this library. One modification is the cross-validation feature, providing the ability to use more than one metric. Lots of training methods like logistics regression and nearest neighbors have received some little improvements.

Features Of Scikit-Learn

1. Cross-validation: There are various methods to check the accuracy of supervised models on unseen data.

2. Unsupervised learning algorithms: Again there is a large spread of algorithms in the offering – starting from clustering, factor analysis, principal component analysis to unsupervised neural networks.

3. Feature extraction: Useful for extracting features from images and text (e.g. Bag of words)

3. Numpy

Numpy is considered as one of the most popular machine learning library in Python.

TensorFlow and other libraries uses Numpy internally for performing multiple operations on Tensors. Array interface is the best and the most important feature of Numpy.

Features Of Numpy

1. **Interactive:** Numpy is very interactive and easy to use.
2. **Mathematics:** Makes complex mathematical implementations very

simple.

3. **Intuitive:** Makes coding real easy and grasping the concepts is easy.
4. **Lot of Interaction:** Widely used, hence a lot of open source contribution.

4. Pandas

Pandas is a machine learning library in Python that provides data structures of high-level and a wide variety of tools for analysis. One of the great feature of this library is the ability to translate complex operations with data using one or two commands. Pandas have so many inbuilt methods for grouping, combining data, and filtering, as well as time-series functionality.

Features Of Pandas

Pandas make sure that the entire process of manipulating data will be easier. Support for operations such as Re-indexing, Iteration, Sorting, Aggregations, Concatenations and Visualizations are among the feature highlights of Pandas.

• My SQL

MySQL is a relational database management system based on the Structured Query Language, which is the popular language for accessing and managing the records in the database.

It is open-source database software, which is supported by Oracle Company. It is fast, scalable, and easy to use database management system in comparison with Microsoft SQL Server and Oracle Database. It is commonly used in conjunction with PHP scripts for creating powerful and dynamic server-side or web-based enterprise applications.

MySQL that provides many things, which are as follows:

- It allows us to implement database operations on tables, rows, columns, and indexes.
- It defines the database relationship in the form of tables (collection of rows

and columns), also known as relations.

- It provides the Referential Integrity between rows or columns of various tables.
- It allows us to updates the table indexes automatically.
- It uses many SQL queries and combines useful information from multiple tables for the end-users.

MySql Structured Query Language(SQL) as we all know is the database language by the use of which we can perform certain operations on the existing database and also we can use this language to create a database. SQL uses certain commands like Create, Drop, Insert, etc. to carry out the required tasks.

These SQL commands are mainly categorized into four categories as:

DDL – Data Definition Language

Data Definition Language actually consists of the SQL commands that can be used to define the database schema. It simply deals with descriptions of the database schema and is used to create and modify the structure of database objects in the database.

- **CREATE**: This command is used to create the database or its objects (like table, index, function, views, store procedure, and triggers).
- **DROP**: This command is used to delete objects from the database.
- **ALTER:** This is used to alter the structure of the database.
- **TRUNCATE:** This is used to remove all records from a table, including all spaces allocated for the records are removed.
- **COMMENT**: This is used to add comments to the data dictionary.
- **RENAME:** This is used to rename an object existing in the database

DQL – Data Query Language

DQL statements are used for performing queries on the data within schema objects. The purpose of the DQL Command is to get some schema relation based on the query passed to it.

- **SELECT:** It is used to retrieve data from the database.

DML – Data Manipulation Language

The SQL commands that deals with the manipulation of data present in the database belong to DML or Data Manipulation Language and this includes most of the SQL statements. It is the component of the SQL statement that controls access to data and to the database.

- **INSERT** : It is used to insert data into a table.
- **UPDATE:** It is used to update existing data within a table.
- **DELETE** : It is used to delete records from a database table.
- **LOCK:** Table control concurrency.

DCL – Data Control Language

DCL includes commands such as GRANT and REVOKE which mainly deal with the rights, permissions, and other controls of the database system.

- **GRANT:** This command gives users access privileges to the database.
- **REVOKE:** This command withdraws the user’s access privileges given by using the GRANT command.

1. Motivation For Work

Python is a great language for new and experienced programmers. Its syntax is small and simple, so you can learn it relatively quickly, but you can write enormous production quality software with it as well. Many companies use Python in parts of their codebases, hence there is a high demand for Python programmers in the job market.

Python is also a well designed language. For example, whitespace and indentation are meaningful, which means it is always clear where code blocks begin and end, without the clutter of brackets in many other languages. This makes for more readable, clean code.

You can write Python in a variety of styles. There is a rich set of object orientation capabilities, but you don’t need to create a class if you don’t need to, and instead write in a more procedural or functional style.

Python comes with a large, well maintained standard library that includes useful tools such as random number generation and regular expressions, and it is easy to install external tools from a collection of tens of thousands of open source packages.

MySQL is globally renowned for being the most secure and reliable database management system used in popular web applications like WordPress, Drupal, Joomla, Facebook and Twitter.

The data security and support for transactional processing that accompany the recent version of MySQL, can greatly benefit any business especially if it is an eCommerce business that involves frequent money transfers. It offers unmatched scalability to facilitate the management of deeply embedded apps using a smaller footprint even in massive warehouses that stack terabytes of data. On-demand flexibility is the star feature of MySQL. It features a distinct storage-engine framework that facilitates system administrators to configure the MySQL database server for a flawless performance.

MySQL comes with the assurance of 24X7 uptime and offers a wide range of high availability solutions like specialized cluster servers and master/slave replication configurations. All the fears and worries that arise in an open source solution can be brought to an end with My SQL’s round-the-clock support and enterprise indemnification. The secure processing and trusted software of MySQL combine to provide effective transactions for large volume projects. It makes maintenance, debugging and upgrades fast and easy while enhancing the end-user experience.

2. Problem Statement

Aim of this report is to understand python, MySQL and then making a project for storing database by taking input from user by CLI and GUI using Ktinker.

2. Methodology

2.1 MySQL Python Connector

MySQL Connector/Python enables Python programs to access MySQL databases, using an API that is compliant with the Python Database API Specification v2.0 (PEP 249). It is written in pure Python and does not have any dependencies except for the Python Standard Library.

Installing mysql-connector module

This module does not come built-in with Python. To install it type the below command in the terminal.

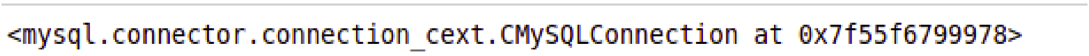
```
pip install mysql-connector-python
```

2.2 Connecting MySQL with Python

Code to Connect

```
import mysql.connector


mysql.connector.connect(host='localhost',
                        database='database',
                        user='root',
                        password='your password')
```

Output



```
<mysql.connector.connection_cext.CMySQLConnection at 0x7f55f6799978>
```

2.3 Making database using Python with MySQL

2.3.1 File name mydatabase.py

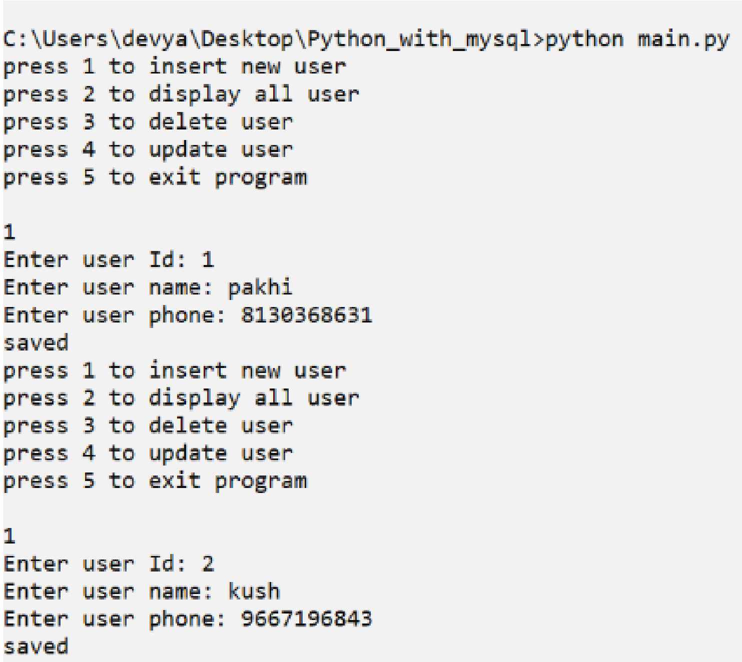
```
import mysql.connector as connector


class myDB: #made class
    def __init__(self):  #made contructor
        #made opject and used connect method to make a connection, then made requirements to connect
        self.con = connector.connect(host='localhost',
                                     database='registration',
                                     user='root',
                                     password='pakhi')
        query = '''create table if not exists
                    user(userId int primary key,
                    userName varchar(100),
                    phone varchar(12))'''
        #A cursor keeps track of the position in the result set, and allows you to perform multiple operations row by row.
        cur = self.con.cursor()
        cur.execute(query)
        # print("created")
    #insert
    #to insert we will make a method(func)
    def insert_user(self,userId,userName,phone):
        query = "insert into user(userId,userName,phone)
values({},'{}','{}')".format(userId,userName,phone)
        # print(query)
        cur = self.con.cursor()
        cur.execute(query)
        self.con.commit()
        print("saved")

    #fetch all
    def fetch_all(self):
        query="select * from user"
        cur = self.con.cursor()
        cur.execute(query)
        for row in cur:
            print("User Id : ", row[0])
            print("User Name : ", row[1])
            print("User Phone : ", row[2])
            print()

    #delete user
    def delete_user(self,userId):
        query = "delete from user where userId = {}".format(userId)
        print(query)
        c = self.con.cursor()
        c.execute(query)
        self.con.commit()
        print("deleted")

    #update
    def update_user(self,userId,newName,newPhone):
        query = "update user set userName='{}', phone={} where
userId='{}'".format(newName,newPhone,userId)
        print(query)
        c = self.con.cursor()
        c.execute(query)
        self.con.commit()
        print("updated")
```

2.3.2 File name main.py

```
from mydatabase import myDB

def main():
    db = myDB()
    while True:
        print("press 1 to insert new user")
        print("press 2 to display all user")
        print("press 3 to delete user")
        print("press 4 to update user")
        print("press 5 to exit program")
        print()

        try:
            choice = int(input())
            if(choice==1):
                #insert user
                userId = int(input("Enter user Id: "))
                userName = input("Enter user name: ")
                userPhone = input("Enter user phone: ")
                db.insert_user(userId,userName,userPhone)

            elif choice == 2:
                #display user
                db.fetch_all()

            elif choice == 3:
                #delete user
                userid = int(input("enter user id id you want to delete"))
                db.delete_user(userid)

            elif choice == 4:
                #update user
                userId = int(input("Enter user Id: "))
                userName = input("new name: ")
                userPhone = input("new phone: ")
                db.update_user()

            elif choice == 5:
                break
            else:
                print("Invalid input! Try again")

        except Exception as e:
            print(e)
            print("Invalid Details ! Try again")

if __name__ == "__main__":
    main()
```

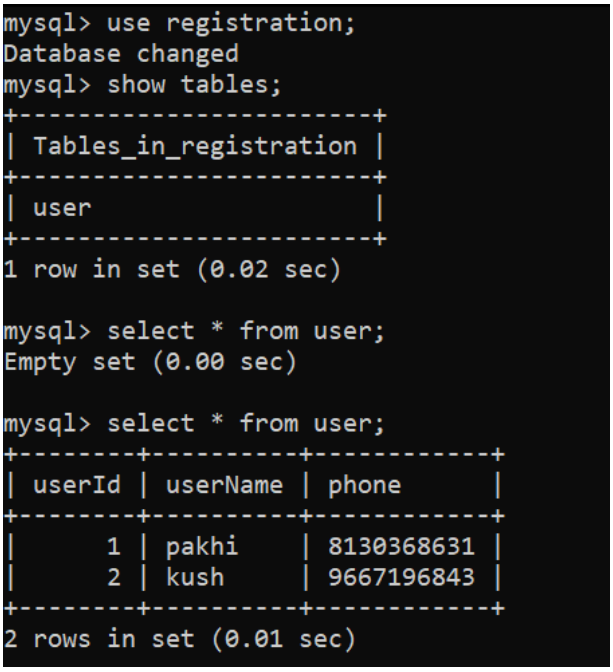
2.3.4 Code Output

CLI



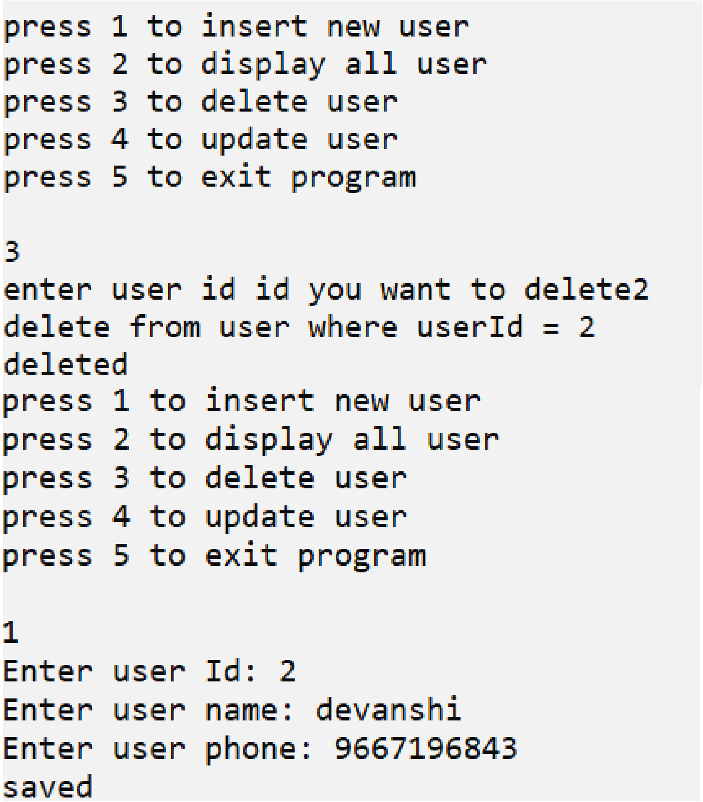
```
C:\Users\devya\Desktop\Python_with_mysql>python main.py
press 1 to insert new user
press 2 to display all user
press 3 to delete user
press 4 to update user
press 5 to exit program

1
Enter user Id: 1
Enter user name: pakhi
Enter user phone: 8130368631
saved
press 1 to insert new user
press 2 to display all user
press 3 to delete user
press 4 to update user
press 5 to exit program

1
Enter user Id: 2
Enter user name: kush
Enter user phone: 9667196843
saved
```

MySQL CLI



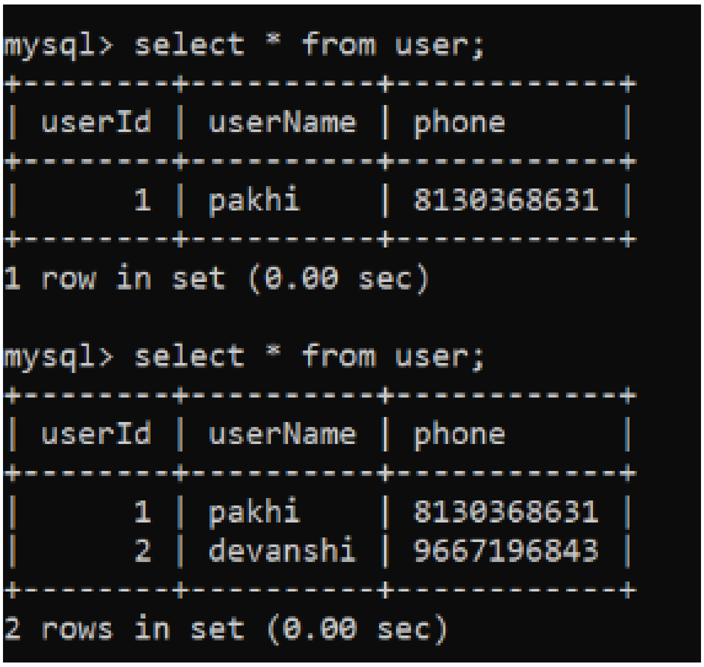
```
mysql> use registration;
Database changed
mysql> show tables;
+------------------------+
| Tables_in_registration |
+------------------------+
| user                   |
+------------------------+
1 row in set (0.02 sec)

mysql> select * from user;
Empty set (0.00 sec)

mysql> select * from user;
+--------+----------+------------+
| userId | userName | phone      |
+--------+----------+------------+
|      1 | pakhi    | 8130368631 |
|      2 | kush     | 9667196843 |
+--------+----------+------------+
2 rows in set (0.01 sec)
```

CLI



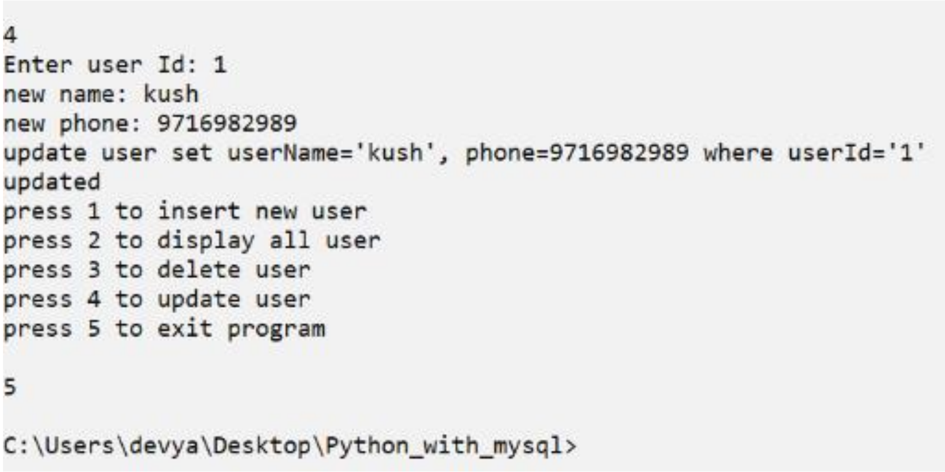
```
press 1 to insert new user
press 2 to display all user
press 3 to delete user
press 4 to update user
press 5 to exit program

3
enter user id id you want to delete2
delete from user where userId = 2
deleted
press 1 to insert new user
press 2 to display all user
press 3 to delete user
press 4 to update user
press 5 to exit program

1
Enter user Id: 2
Enter user name: devanshi
Enter user phone: 9667196843
saved
```

MySQL CLI



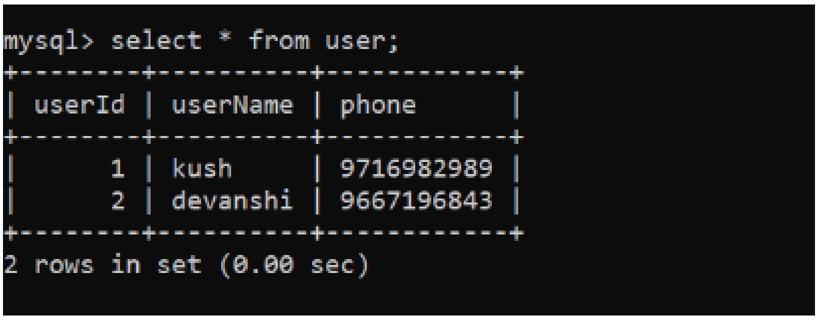
```
mysql> select * from user;
+--------+----------+------------+
| userId | userName | phone      |
+--------+----------+------------+
|      1 | pakhi    | 8130368631 |
+--------+----------+------------+
1 row in set (0.00 sec)

mysql> select * from user;
+--------+----------+------------+
| userId | userName | phone      |
+--------+----------+------------+
|      1 | pakhi    | 8130368631 |
|      2 | devanshi | 9667196843 |
+--------+----------+------------+
2 rows in set (0.00 sec)
```

CLI



```
4
Enter user Id: 1
new name: kush
new phone: 9716982989
update user set userName='kush', phone=9716982989 where userId='1'
updated
press 1 to insert new user
press 2 to display all user
press 3 to delete user
press 4 to update user
press 5 to exit program

5

C:\Users\devya\Desktop\Python_with_mysql>
```

MySQL CLI



```
mysql> select * from user;
+--------+----------+------------+
| userId | userName | phone      |
+--------+----------+------------+
|      1 | kush     | 9716982989 |
|      2 | devanshi | 9667196843 |
+--------+----------+------------+
2 rows in set (0.00 sec)
```

2.4 Ktinker

Tkinter is the standard GUI library for Python. Python when combined with Tkinter provides a fast and easy way to create GUI applications. Tkinter provides a powerful object-oriented interface to the Tk GUI toolkit.

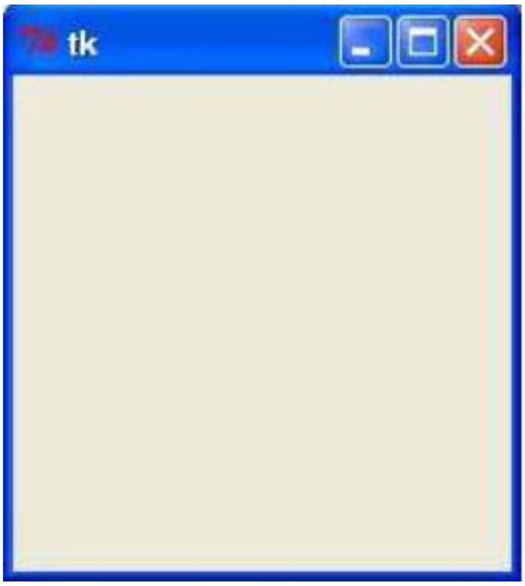
Creating a GUI application using Tkinter is an easy task. All you need to do is perform the following steps −

3. Import the *Tkinter* module.
4. Create the GUI application main window.
5. Add one or more of the above-mentioned widgets to the GUI application.
6. Enter the main event loop to take action against each event triggered by the user.

Example

```
#!/usr/bin/python


import Tkinter
top = Tkinter.Tk()
# Code to add widgets will go here...
top.mainloop()
```

This would create a following window –



Tkinter Widgets

Tkinter provides various controls, such as buttons, labels and text boxes used in a GUI application. These controls are commonly called widgets.

There are currently 15 types of widgets in Tkinter. We present these widgets as well as a brief description in the following table –

Sr.No.	Operator & Description

1	Button The Button widget is used to display buttons in your application.
2	Canvas The Canvas widget is used to draw shapes, such as lines, ovals, polygons and rectangles, in your application.
3	Checkbutton The Checkbutton widget is used to display a number of options as checkboxes. The user can select multiple options at a time.
4	Entry The Entry widget is used to display a single-line text field for accepting values from a user.
5	Frame The Frame widget is used as a container widget to organize other widgets.
6	Label The Label widget is used to provide a single-line caption for other widgets. It can also contain images.
7	Listbox The Listbox widget is used to provide a list of options to a user.
8	Menubutton The Menubutton widget is used to display menus in your application.
9	Menu

	The Menu widget is used to provide various commands to a user. These commands are contained inside Menubutton.
10	Message The Message widget is used to display multiline text fields for accepting values from a user.
11	Radiobutton The Radiobutton widget is used to display a number of options as radio buttons. The user can select only one option at a time.
12	Scale The Scale widget is used to provide a slider widget.
13	Scrollbar The Scrollbar widget is used to add scrolling capability to various widgets, such as list boxes.
14	Text The Text widget is used to display text in multiple lines.
15	Toplevel The Toplevel widget is used to provide a separate window container.
16	Spinbox The Spinbox widget is a variant of the standard Tkinter Entry widget, which can be used to select from a fixed number of values.
17	PanedWindow

	A PanedWindow is a container widget that may contain any number of panes, arranged horizontally or vertically.
18	LabelFrame A labelframe is a simple container widget. Its primary purpose is to act as a spacer or container for complex window layouts.
19	tkMessageBox This module is used to display message boxes in your applications.

Standard attributes

Let us take a look at how some of their common attributes.such as sizes, colors and fonts are specified.

- Dimensions
- Colors
- Fonts
- Anchors
- Relief styles
- Bitmaps
- Cursors

Geometry Management

All Tkinter widgets have access to specific geometry management methods, which have the purpose of organizing widgets throughout the parent widget area. Tkinter exposes the following geometry manager classes: pack, grid, and place.

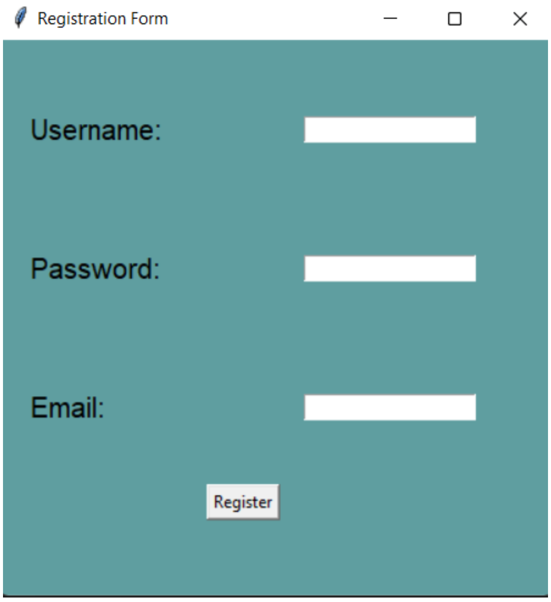
- The *pack()* Method − This geometry manager organizes widgets in blocks before placing them in the parent widget.
- The *grid()* Method − This geometry manager organizes widgets in a table-like structure in the parent widget.

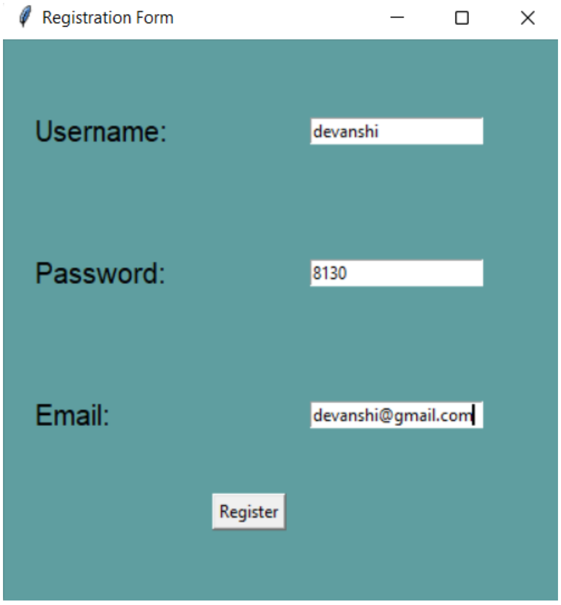
- The *place()* Method − This geometry manager organizes widgets by placing them in a specific position in the parent widget.

2.5 Making GUI using KTinker

```
from tkinter import *
import tkinter.messagebox as msg

#built-in function of Tkinter which is allowing you to create a window for our
application.  we are giving the window name root
root = Tk()
# built-in attribute of tkinter to provide fix w*h
root.geometry('400x400')

root.title("Registration Form")

# to make changes in the window
root.config(bg="cadetblue")

def register_data():
    import mysql.connector as cnct
    mydb =
cnct.connect(host='localhost',user='root',password='pakhi',database='registration_for
m')

    # it is used to check whether all this configuration passed is correct or not.
    # or it is used to use all the sql command in python
    cur = mydb.cursor()
    username = username_entry.get()

    password = password_entry.get()
    email = email_entry.get()
    query = "insert into registration
values('{}','{}','{}')".format(username,password,email)

    cur.execute(query)
    mydb.commit()
    #to show the message
    msg.showinfo("Registraion","Registered Successfully")

# to insert any text in the window we use Label attribute.
username_label = Label(root,text="Username:",bg="cadetblue",font=('bold','15'))
username_label.place(x=20,y=50) #place for the text

password_label = Label(root,text = "Password:", bg = "cadetblue", font = ('bold',15))
password_label.place(x=20,y=150)

email_label = Label(root,text = "Email:", bg = "cadetblue", font = ('bold',15))
email_label.place(x=20,y=250)

username_entry = Entry()
username_entry.place(x=220,y=55)

password_entry = Entry()
password_entry.place(x=220,y=155)

email_entry = Entry()
email_entry.place(x=220,y=255)

register_button = Button(root,text="Register", command=register_data)
register_button.place(x=150,y=320)

root.mainloop() # we are using it to access that particular window on our screen.
```

2.6 Taking Input Using GUI

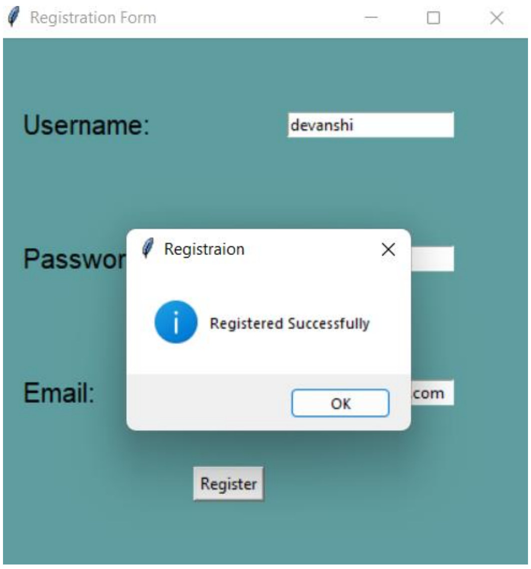




Username: devanshi

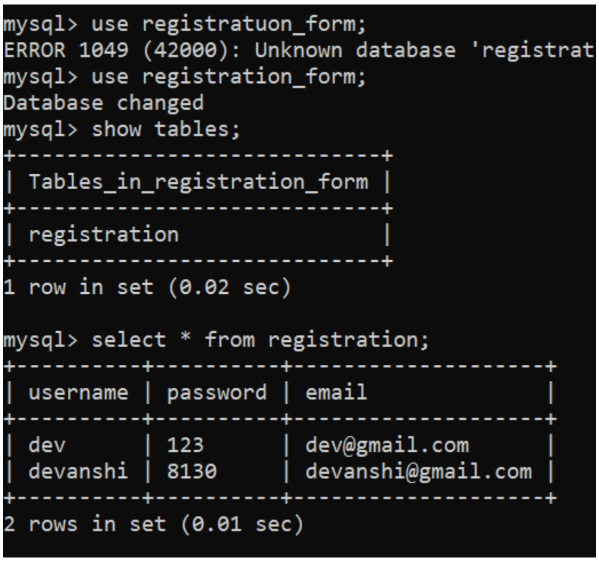
Password: 8130

Email: devanshi@gmail.com

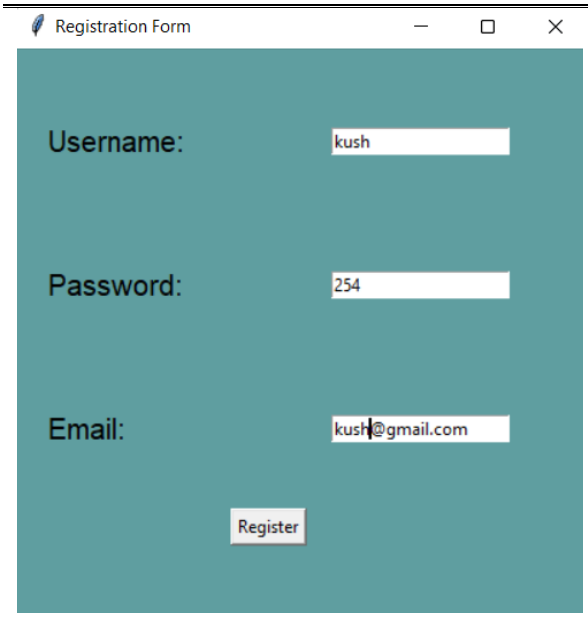


Username: devanshi

Registered Successfully



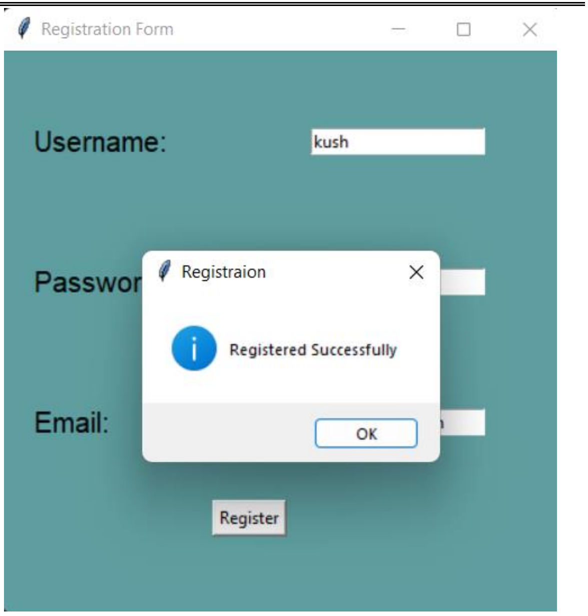
```
mysql> use registratuon_form;
ERROR 1049 (42000): Unknown database 'registrat
mysql> use registration_form;
Database changed
mysql> show tables;
+----------------------------+
| Tables_in_registration_form |
+----------------------------+
| registration               |
+----------------------------+
1 row in set (0.02 sec)

mysql> select * from registration;
+----------+----------+-------------------+
| username | password | email             |
+----------+----------+-------------------+
| dev      | 123      | dev@gmail.com     |
| devanshi | 8130     | devanshi@gmail.com |
+----------+----------+-------------------+
2 rows in set (0.01 sec)
```


Registration Form

Username: kush

Password: 254

Email: kush@gmail.com


Registration Form

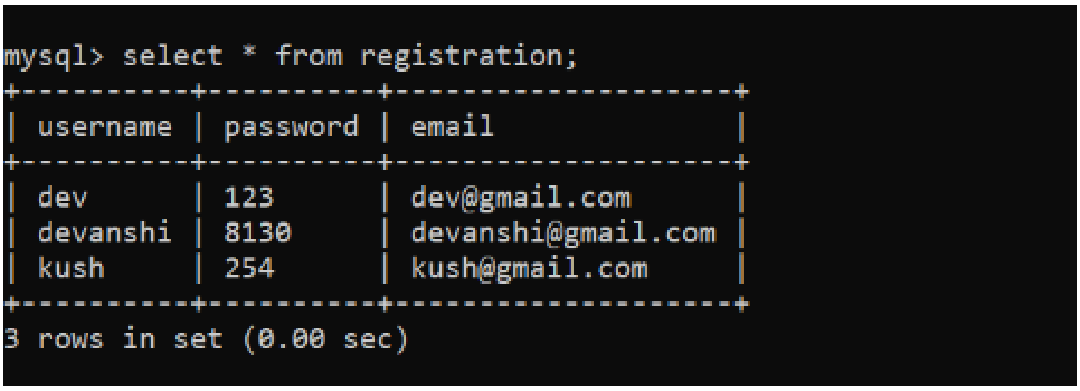
Username: kush

Passwor

Email:

Registraion

Registered Successfully


```
mysql> select * from registration;
+----------+----------+--------------------+
| username | password | email              |
+----------+----------+--------------------+
| dev      | 123      | dev@gmail.com      |
| devanshi | 8130     | devanshi@gmail.com |
| kush     | 254      | kush@gmail.com     |
+----------+----------+--------------------+
3 rows in set (0.00 sec)
```

3. SYSTEM REQUIREMENTS

3.1 Hardware Requirement

	Minimum	Recommended
CPU	64bit x86	Multi Core 64bit x86
RAM	4 GB	8 GB or higher
Display	1024×768	1920×1200 or higher

3.2 Software Requirement

a. Python(IDLE)

b. VS code

c. MySQL

d. IDLE

e. ANACONDA

f. SPYDER

4. RESULT

Both Code are running successfully as they are taking input from the user and storing into the databases.

5. Conclusion

The industrial training that I want through within the few days brought in new technology and expanded my knowledge in the IT industry.

I got the opportunity to put known concepts into practice in real project and to learn new concepts through applying them.

Meeting with deadlines, keeping the code quality, trying various approaches to determine the best method and finally completing the industrial training with success were experience that gained throughout the training period.

I know best. Also even if you think your good in something there is always something new that you haven’t learnt.

PYHTON developer think about your future in python development. PYTHON is great tool for general purpose programming. It is user friendly one can easily understand the programming techniques.

6. **References**

[1] https://www.tutorialspoint.com/python/python_quick_guide.htm

[2] https://www.tutorialspoint.com/python/python_gui_programming.htm

[3] https://medium.datadriveninvestor.com/140-python-projects-with-source-code-fa12c9e2aeac

[4] https://www.geeksforgeeks.org/mysql-connector-python-module-in-python/?id=discuss

[5] https://www.python.org/

[6] https://www.mysqltutorial.org/

[7] https://en.wikipedia.org/wiki/Python_(programming_language)