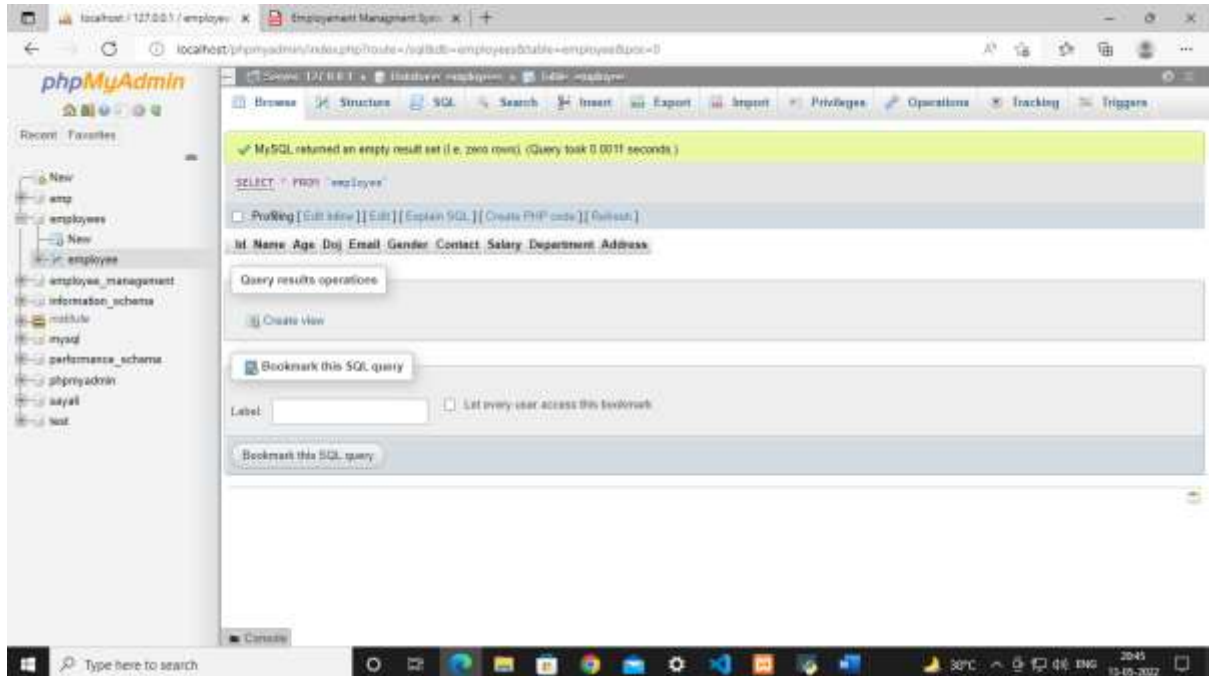# Employees Management

In our project, on "Employee Management", we have tried to show how the Data/information in organization is managed. Employee Management is a distributed application, developed to maintain the details of employees working in any organization. It maintains the information about the personal details of their employees which will store in MySQL database through python.

The project has been designed with the kind of features are, in that u can

- Add Employees Details
- Update Employees Details
- Delete Employees Details
- Display Employees Details

I. Add Employees Details :- Through Add Employees Details we can add the new employee details in Employees management.
II. Update Employees Details :- Through Update Employees Details we can update any of your Employee details in Employee management.
III. Delete Employees Details :- Through Delete Employees Delete we can delete the new employee details in Employee management.
IV. Display Employee Details :- Through Display Employees Details we Can display all of our Employee data which is stored in Employee management.

- Creating database and table for employees management.

The two script file of python which are frontend and backend of employees management are.

1. db.py   (Database.py)
2. Main.py

db.py :

In this we have connect our database with python through importing mysql.connector.

```python
from os import curdir
from sqlite3 import Cursor
import mysql.connector
myDatabase=mysql.connector.connect(host="localhost",      user="root",
database="employees" )
cursor=myDatabase.cursor()
def insert(Name,Age,Doj,Email,Gender,Contact,Salary,Department,Address):
```

```python
    global cursor
    global myDatabase
    sql="insert into employee
(Name,Age,Doj,Email,Gender,Contact,Salary,Department,Address) values
(%s,%s,%s,%s,%s,%s,%s,%s,%s)"
    val=(Name,Age,Doj,Email,Gender,Contact,Salary,Department,Address)
    cursor.execute(sql,val)
    myDatabase.commit()

def update(id,Name,Age,Doj,Email,Gender,Contact,Salary,Department,Address):
    sql="update employee set
Name=%s,Age=%s,Doj=%s,Email=%s,Gender=%s,Contact=%s,Salary=%s,Department=%s,Ad
dress=%s where id=%s"
    val=(Name,Age,Doj,Email,Gender,Contact,Salary,Department,Address,id)

    cursor.execute(sql,val)
    myDatabase.commit()

def remove(id):
        sql="delete from employee where id=%s"
        val=(id,)
        cursor.execute(sql,val)
        myDatabase.commit()

def fetch():
    sql="SELECT * from employee"
    cursor.execute(sql)
    rows=cursor.fetchall()
        # print(rows)
    return rows
```

Main.py :

In this we have have import our backend file and create our GUI application for employees management using tkinter. Tkinter is a standard GUI library for python. Python when combined with Tkinter provides a fast and easy way to create GUI applications. After that we have designed our application.

```python
from tkinter import *
from tkinter import ttk
```

```python
from tkinter import messagebox
import db

root = Tk()
root.title("Employees Management ")
root.geometry("1920x1080+0+0")
root.config(bg="#000000")
root.state("zoomed")

name = StringVar()
age = StringVar()
doj = StringVar()
gender = StringVar()
email = StringVar()
contact = StringVar()
salary = StringVar()
Department = StringVar()

entries_frame = Frame(root, bg="#000000")
entries_frame.pack(side=TOP, fill=X)
title = Label(entries_frame, text="Employees Management", font=("Rockwell
Extra Bold", 19, "bold"), bg="#000000", fg="white")
title.grid(row=0, columnspan=2, padx=10, pady=20, sticky="w")

lblName = Label(entries_frame, text="Name", font=("Bodoni MT", 16),
bg="#000000", fg="white")
lblName.grid(row=1, column=0, padx=10, pady=10, sticky="w")
txtName = Entry(entries_frame, textvariable=name, font=("Bodoni MT", 16),
width=30)
txtName.grid(row=1, column=1, padx=10, pady=10, sticky="w")

lblAge = Label(entries_frame, text="Age", font=("Bodoni MT", 16),
bg="#000000", fg="white")
lblAge.grid(row=2, column=0, padx=10, pady=10, sticky="w")
txtAge = Entry(entries_frame, textvariable=age, font=("Bodoni MT", 16),
width=30)
txtAge.grid(row=2, column=1, padx=10, pady=10, sticky="w")

lbldoj = Label(entries_frame, text="D.O.J", font=("Bodoni MT", 16),
bg="#000000", fg="white")
lbldoj.grid(row=3, column=0, padx=10, pady=10, sticky="w")
txtDoj = Entry(entries_frame, textvariable=doj, font=("Bodoni MT", 16),
width=30)
txtDoj.grid(row=3, column=1, padx=10, pady=10, sticky="w")
```

```python
lblEmail = Label(entries_frame, text="Email", font=("Bodoni MT", 16),
bg="#000000", fg="white")
lblEmail.grid(row=4, column=0, padx=10, pady=10, sticky="w")
txtEmail = Entry(entries_frame, textvariable=email, font=("Bodoni MT", 16),
width=30)
txtEmail.grid(row=4, column=1, padx=10, pady=10, sticky="w")

lblGender = Label(entries_frame, text="Gender", font=("Bodoni MT", 16),
bg="#000000", fg="white")
lblGender.grid(row=5, column=0, padx=10, pady=10, sticky="w")
comboGender = ttk.Combobox(entries_frame, font=("Bodoni MT", 16), width=28,
textvariable=gender, state="readonly")
comboGender['values'] = ("Male", "Female", "Other")
comboGender.grid(row=5, column=1, padx=10, sticky="w")

lblContact = Label(entries_frame, text="Contact No", font=("Bodoni MT", 16),
bg="#000000", fg="white")
lblContact.grid(row=6, column=0, padx=10, pady=10, sticky="w")
txtContact = Entry(entries_frame, textvariable=contact, font=("Bodoni MT",
16), width=30)
txtContact.grid(row=6, column=1, padx=10, sticky="w")


lblSalary = Label(entries_frame, text="Salary", font=("Bodoni MT", 16),
bg="#000000", fg="white")
lblSalary.grid(row=1, column=2, padx=10, pady=10, sticky="w")
txtSalary = Entry(entries_frame, textvariable=salary, font=("Bodoni MT", 16),
width=30)
txtSalary.grid(row=1, column=3, padx=10, pady=10, sticky="w")

lblDepartment = Label(entries_frame, text="Department", font=("Bodoni MT",
16), bg="#000000", fg="white")
lblDepartment.grid(row=2, column=2, padx=10, pady=10, sticky="w")
txtDepartment = Entry(entries_frame, textvariable=Department, font=("Bodoni
MT", 16), width=30)
txtDepartment.grid(row=2, column=3, padx=10, pady=10, sticky="w")

lblAddress = Label(entries_frame, text="Address", font=("Bodoni MT", 16),
bg="#000000", fg="white")
lblAddress.grid(row=3, column=2, padx=10, pady=10, sticky="w")

txtAddress = Text(entries_frame, width=20, height=2, font=("Bodoni MT", 16))
txtAddress.grid(row=3, column=3, columnspan=2, padx=10, sticky="w")

def getData(event):
    selected_row = tv.focus()
```

```python
        data = tv.item(selected_row)
        global row
        row = data["values"]
        name.set(row[1])
        age.set(row[2])
        doj.set(row[3])
        email.set(row[4])
        gender.set(row[5])
        contact.set(row[6])
        salary.set(row[7])
        Department.set(row[8])
        txtAddress.delete(1.0, END)
        txtAddress.insert(END, row[9])

def dispalyAll():
    tv.delete(*tv.get_children())
    for row in Database.fetch():
        tv.insert("", END, values=row)


def add_employee():
    if txtName.get() == "" or txtAge.get() == "" or txtDoj.get() == "" or
txtEmail.get() == "" or comboGender.get() == "" or txtContact.get() == "" or
txtSalary.get() == "" or txtDepartment.get() == "" or txtAddress.get(
            1.0, END) == "":
        messagebox.showerror("Erorr in Input", "Please Fill All the Details")
        return
    db.insert(txtName.get(),txtAge.get(), txtDoj.get() , txtEmail.get()
,comboGender.get(), txtContact.get(),
txtSalary.get(),txtDepartment.get(),txtAddress.get(
            1.0, END))
    messagebox.showinfo("Success", "Record Inserted")
    clearAll()
    dispalyAll()



def update_employee():
    if txtName.get() == "" or txtAge.get() == "" or txtDoj.get() == "" or
txtEmail.get() == "" or comboGender.get() == "" or txtContact.get() == "" or
txtSalary.get() == "" or txtDepartment.get() == "" or txtAddress.get(
            1.0, END) == "":
        messagebox.showerror("Erorr in Input", "Please Fill All the Details")
        return
    db.update(row[0],txtName.get(), txtAge.get(), txtDoj.get(),
txtEmail.get(), comboGender.get(), txtContact.get(), txtSalary.get(),
txtDepartment.get(),
```

```python
                txtAddress.get(
                    1.0, END))
    messagebox.showinfo("Success", "Record Update")
    clearAll()
    dispalyAll()


def delete_employee():
    db.remove(row[0])
    clearAll()
    dispalyAll()


def clearAll():
    name.set("")
    age.set("")
    doj.set("")
    gender.set("")
    email.set("")
    contact.set("")
    salary.set("")
    Department.set("")
    txtAddress.delete(1.0, END)



btn_frame = Frame(entries_frame, bg="#000000")
btn_frame.grid(row=8, column=0, columnspan=4, padx=10, pady=10, sticky="w")
btnAdd = Button(btn_frame, command=add_employee, text="Add Details", width=15,
font=("Rockwell Condensed", 16, "bold"), fg="white",
                bg="#c0392b", bd=0).grid(row=0, column=0)
btnEdit = Button(btn_frame, command=update_employee, text="Update Details",
width=15, font=("Rockwell Condensed", 16, "bold"),
                fg="white", bg="#16a085",
                bd=0).grid(row=0, column=1, padx=10)
btnClear = Button(btn_frame, command=clearAll, text="Clear Details", width=15,
font=("Rockwell Condensed", 16, "bold"), fg="white",
                bg="#29809b",
                bd=0).grid(row=0, column=3, padx=10)
btnDelete = Button(btn_frame, command=delete_employee, text="Delete Details",
width=15, font=("Rockwell Condensed", 16, "bold"),
                fg="white", bg="#f39c12",
                bd=0).grid(row=0, column=2, padx=10)



tree_frame = Frame(root, bg="#ecf0f1")
tree_frame.place(x=0, y=480, width=1400, height=420)
style = ttk.Style()
```

```
                rowheight=50)
style.configure("mystyle.Treeview.Heading", font=('Calibri', 18))
tv = ttk.Treeview(tree_frame, columns=(1,2,3,4,5,6,7,8,9,10),
style="mystyle.Treeview")
tv.heading("1", text="ID")
tv.column("1", width=5)
tv.heading("2", text="Name")
tv.heading("3", text="Age")
tv.column("3", width=5)
tv.heading("4", text="D.O.J")
tv.column("4", width=45)
tv.heading("5", text="Email")
tv.heading("6", text="Gender")
tv.column("6", width=30)
tv.heading("7", text="Contact")
tv.column("7", width=90)
tv.heading("8",text="salary")
tv.column("8", width=25)
tv.heading("9",text="Department")
tv.column("9", width=90)
tv.heading("10", text="Address")
tv['show'] = 'headings'
tv.bind("<ButtonRelease-1>", getData)
tv.pack(fill=X)

dispalyAll()
root.mainloop()
```

After the execution of the code the output will display the GUI part
Shown below.

Adding Employee :

Here we can see the details has been added in both frontend and backend.

Update Details :

Through update we can make changes in data of employees.

Here we have updated the salary in both frontend and backend.

Employee Management
Project by – Rizwan.
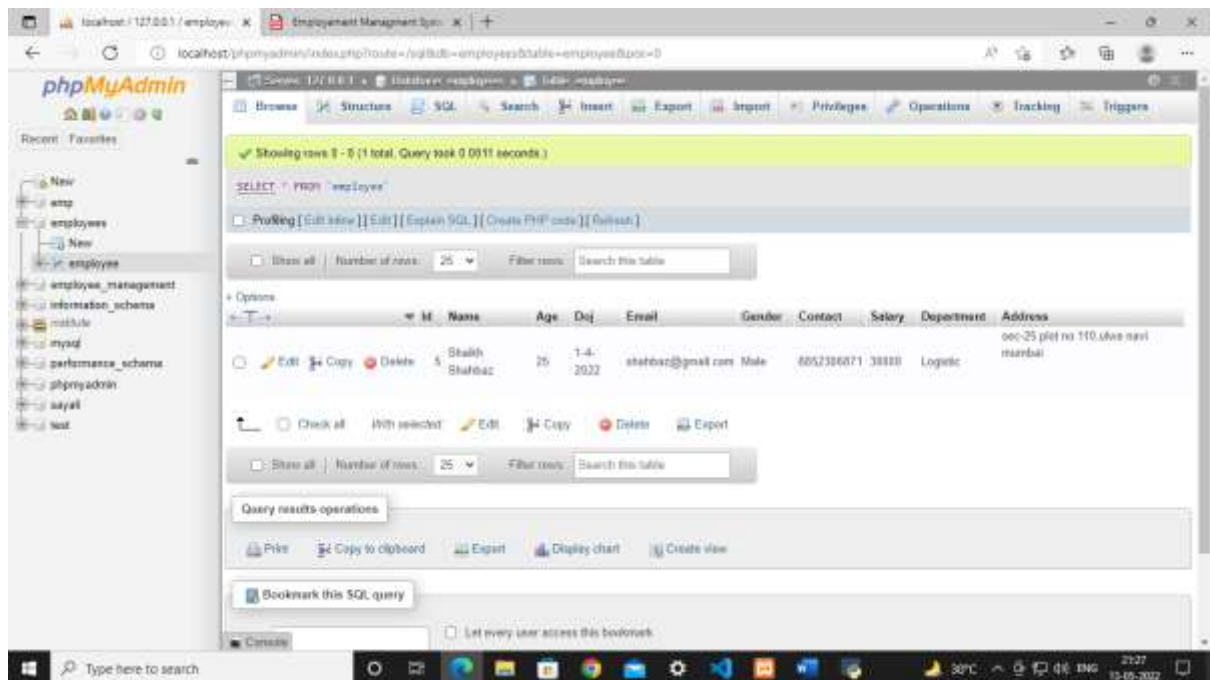
Delete :

Select the employee data which you want to delete and press delete details and it will data from both frontend and backend.
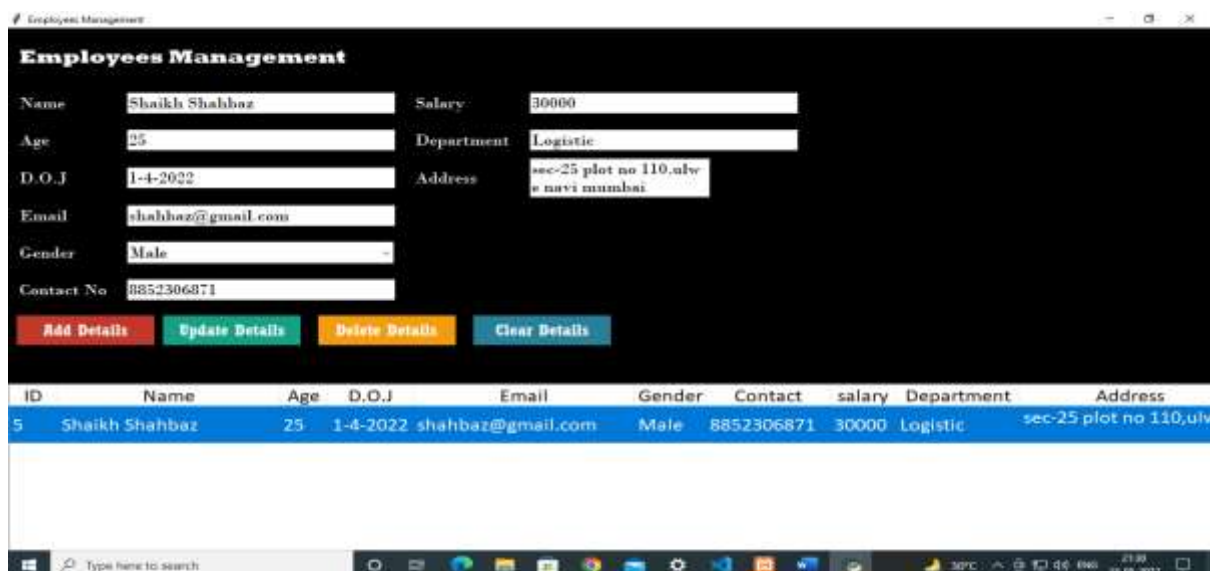
Employee Management
Project by – Rizwan.
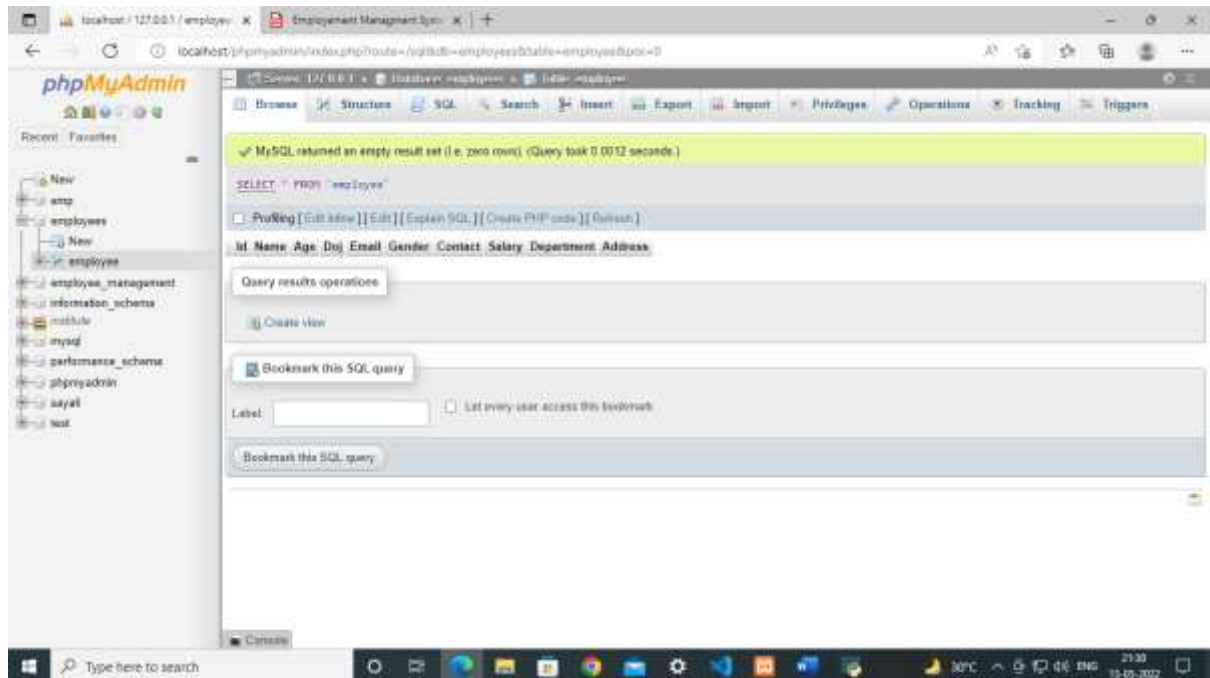
Employee Management
Project by – Rizwan.

Display :

Here is the Employees details shown through Tkinter tree view.

ID	Name	Age	D.O.J	Email	Gender	Contact	salary	Department	Address
6	Sohail	21	1-1-22	sohail@gmail.com	Male	9890076512	15000	calling	cheeta camp trombay
7	Sam	22	9-1-22	sam@gmail.com	Male	9890154391	18000	mailing	Pailipada trombay
8	Mohit	23	1-5-22	Mohit@gmail.com	Male	8907154391	20000	Desktop Supp	trombay koliwada mu