# Aakshat Malhotra || 2021UCS1557

# Problem Statement:

To make a salary/payroll management System

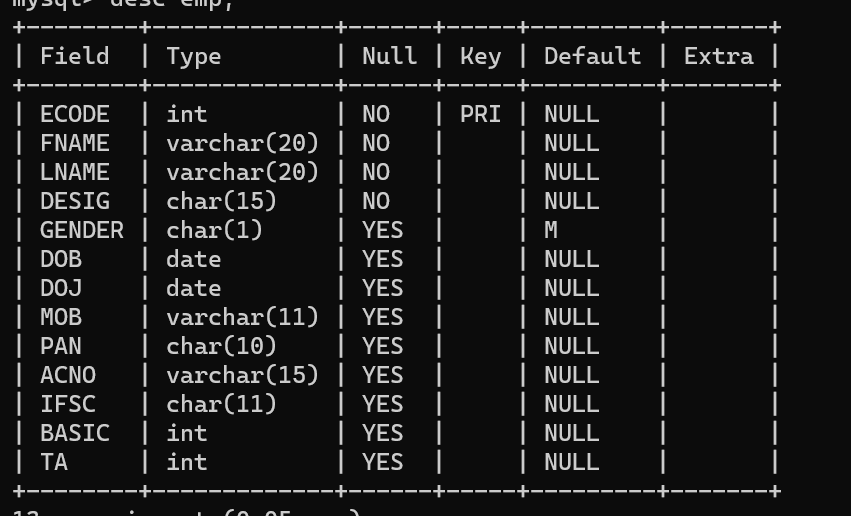
# Requirement Analysis

1. In this system we can add details about employees
2. We can update paybill on monthly basis.
3. We can remove employee and their payroll data
4. This system automatically calculates various allowances and deductions in the salary.

# Tables Used

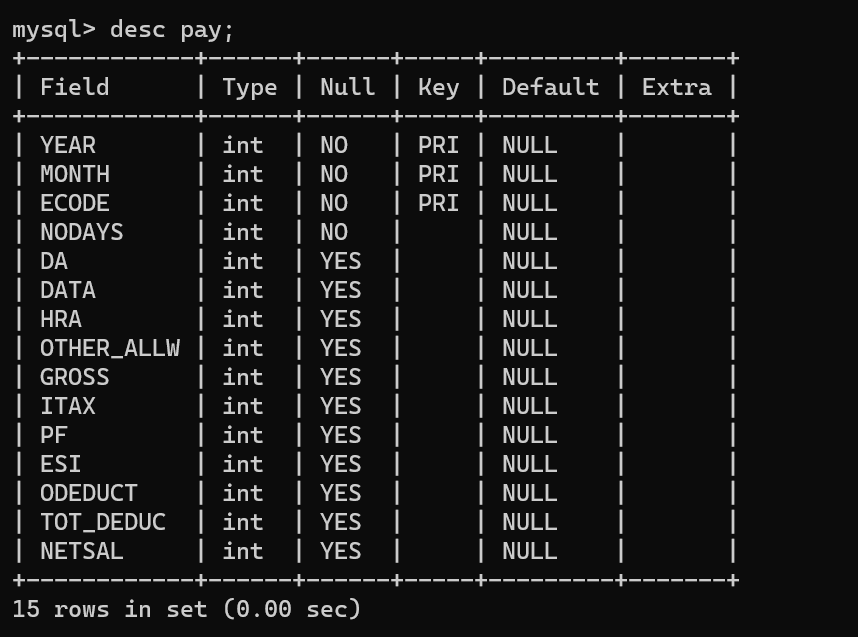
1. 3 tables are used
2. Employee table to store employee information

Structure of Employee Table:



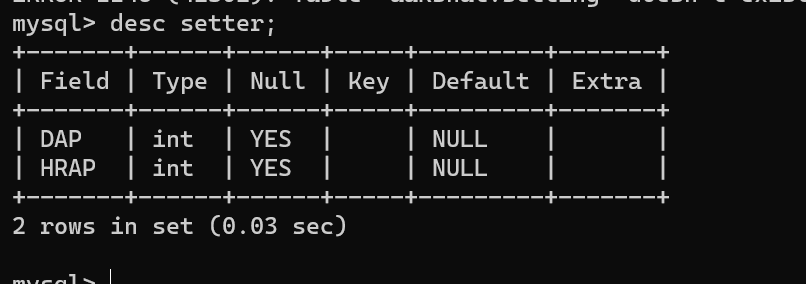
1. Pay table to store the paybill

Structure of Pay Table:

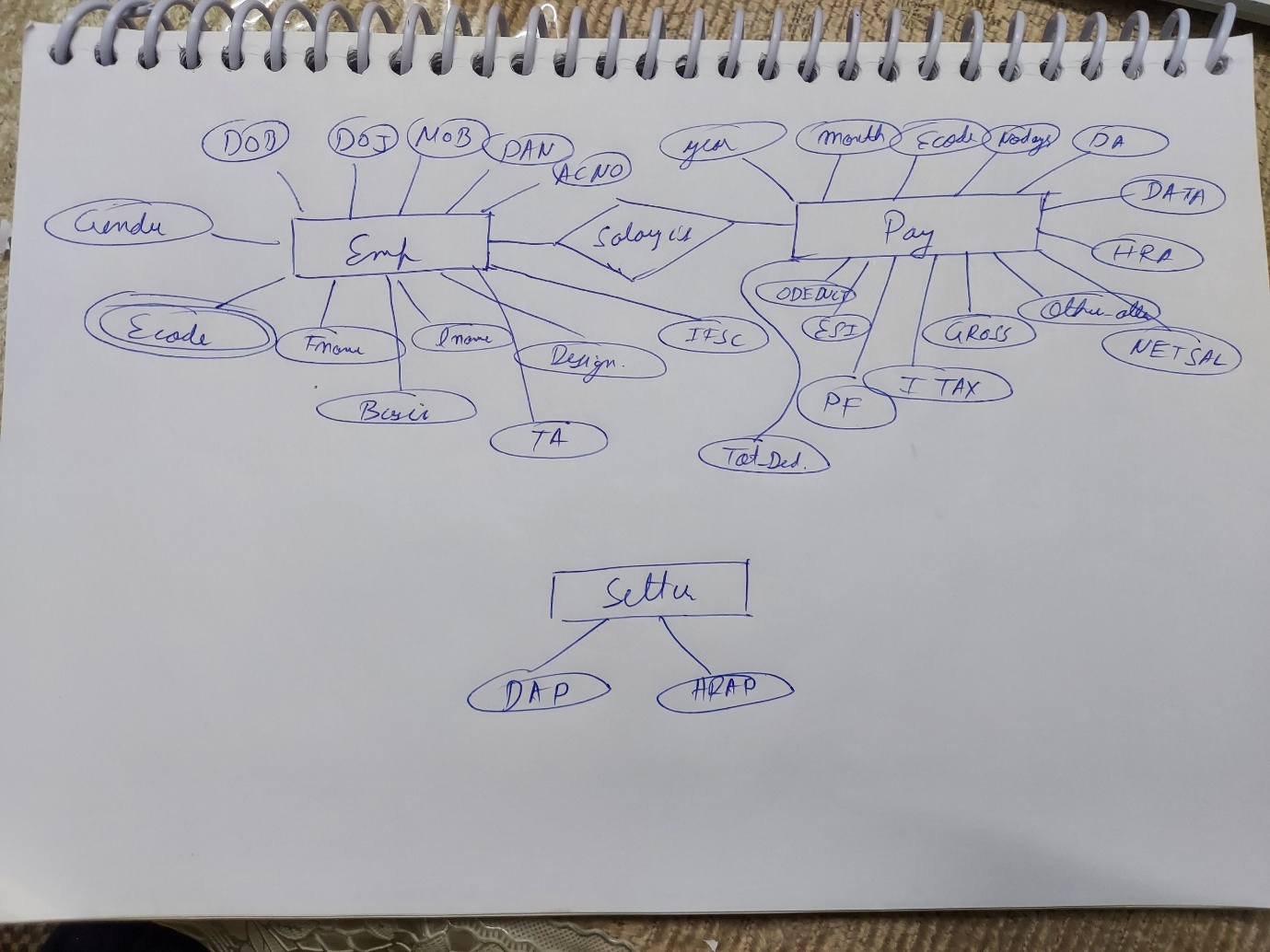


1. Setter to store the current DA and Hra percentage

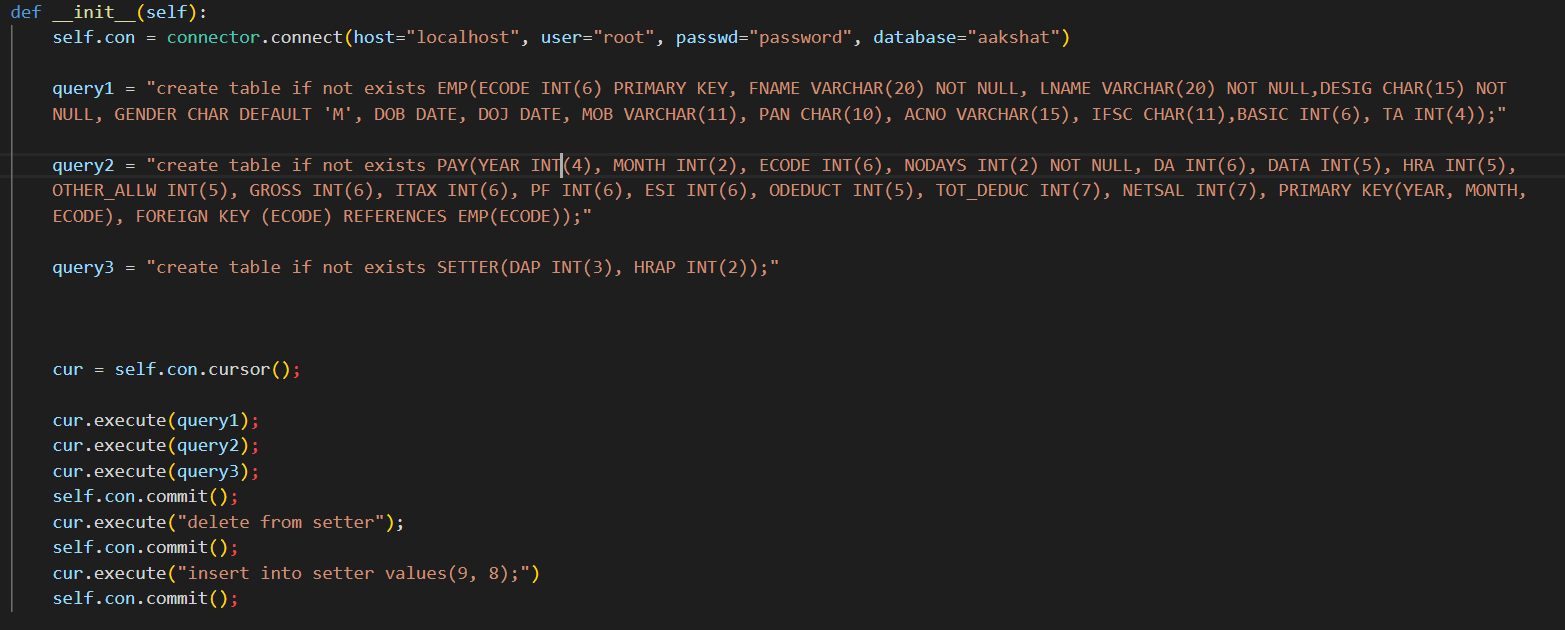
Structure of Setter Table:



# ER Diagram



# Create Table using queries in python



# Insert values queries in python

Values are used inserted by console

# Python Code

import mysql.connector as connector

class DBHelper:

    def \_\_init\_\_(self):

        self.con = connector.connect(host="localhost", user="root", passwd="password", database="aakshat")

        query1 = "create table if not exists EMP(ECODE INT(6) PRIMARY KEY, FNAME VARCHAR(20) NOT NULL, LNAME VARCHAR(20) NOT NULL,DESIG CHAR(15) NOT NULL, GENDER CHAR DEFAULT 'M', DOB DATE, DOJ DATE, MOB VARCHAR(11), PAN CHAR(10), ACNO VARCHAR(15), IFSC CHAR(11),BASIC INT(6), TA INT(4));"

        query2 = "create table if not exists PAY(YEAR INT(4), MONTH INT(2), ECODE INT(6), NODAYS INT(2) NOT NULL, DA INT(6), DATA INT(5), HRA INT(5), OTHER\_ALLW INT(5), GROSS INT(6), ITAX INT(6), PF INT(6), ESI INT(6), ODEDUCT INT(5), TOT\_DEDUC INT(7), NETSAL INT(7), PRIMARY KEY(YEAR, MONTH, ECODE), FOREIGN KEY (ECODE) REFERENCES EMP(ECODE));"

        query3 = "create table if not exists SETTER(DAP INT(3), HRAP INT(2));"

        cur = self.con.cursor();

        cur.execute(query1);

        cur.execute(query2);

        cur.execute(query3);

        self.con.commit();

        cur.execute("delete from setter");

        self.con.commit();

        cur.execute("insert into setter values(9, 8);")

        self.con.commit();

    def input\_employee(self, ecode, fname, lname, desgination, gender, dob, doj, mob, pan, acno, ifsc, basic, ta):

        cur = self.con.cursor();

        query = f"insert into emp values({ecode}, '{fname}', '{lname}', '{desgination}', '{gender}', '{dob}','{doj}','{mob}','{pan}','{acno}','{ifsc}', {basic}, {ta})";

        cur.execute(query);

        self.con.commit();

        print("Data entered successfully")

    def display\_AllEmp(self):

        cur = self.con.cursor();

        query = "select \* from emp;";

        cur.execute(query);

        for row in cur:

            print()

            print(f"ECODE: {row[0]}")

            print(f"First Name: {row[1]}")

            print(f"Last Name: {row[2]}")

            print(f"Designation: {row[3]}")

            print(f"Gender: {row[4]}")

            print(f"DOB: {row[5]}")

            print(f"DOJ: {row[6]}")

            print(f"MOB: {row[7]}")

            print(f"PAN: {row[8]}")

            print(f"ACNO: {row[9]}")

            print(f"IFSC: {row[10]}")

            print(f"Basic Pay: {row[11]}")

            print(f"TA: {row[12]}")

            print()

            print()

    def display\_SpecificEmp(self, ecode):

        cur = self.con.cursor();

        query = f"select \* from emp where ecode = {ecode}";

        cur.execute(query);

        for row in cur:

            print();

            print(f"ECODE: {row[0]}")

            print(f"First Name: {row[1]}")

            print(f"Last Name: {row[2]}")

            print(f"Designation: {row[3]}")

            print(f"Gender: {row[4]}")

            print(f"DOB: {row[5]}")

            print(f"DOJ: {row[6]}")

            print(f"MOB: {row[7]}")

            print(f"PAN: {row[8]}")

            print(f"ACNO: {row[9]}")

            print(f"IFSC: {row[10]}")

            print(f"Basic Pay: {row[11]}")

            print(f"TA: {row[12]}")

            print()

    def percentage\_setter(self, daPercent, HRAPercent):

        cur = self.con.cursor();

        cur.execute("delete from setter")

        cur.execute(f"insert into setter values ({daPercent}, {HRAPercent})")

        self.con.commit();

    def show\_rates(self):

        cur = self.con.cursor();

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

        dap= 0

        hrp = 0

        for row in cur:

            dap = row[0]

            hrp = row[1]

        print(f"DA percentage is {dap} \nHRA percentage is {hrp}")

    def salary\_entryInd(self, year, month, ecode, no\_of\_days, da, DATA, hra, other\_allow, gross, itax, pf, esi, odeduct, tot\_deduc, netsal):

        cur = self.con.cursor();

        cur.execute(f"insert into pay values({year},{month}, {ecode}, {no\_of\_days}, {da}, {DATA}, {hra}, {other\_allow}, {gross}, {itax},  {pf}, {esi}, {odeduct}, {tot\_deduc}, {netsal})");

        self.con.commit();

    def salary\_entry(self):

        cur = self.con.cursor();

          # gettting percentages for da and hra

        cur.execute("select \* from setter;")

        da\_per= 0

        hra\_per = 0

        for row in cur:

            da\_per = row[0];

            hra\_per = row[1];

        year = int(input("Enter the year: "))

        month = int(input("Enter the month: "))

        cur.execute("select ecode, basic, TA from emp;")

        ls = [];

        for row in cur:

            ecode = row[0];

            basic = row[1];

            TA = row[2];

            print(f"Details for Ecode:{ecode}")

            no\_of\_day = int(input("Enter the number of days worked: "))

            da = basic \* da\_per/100;

            data = da + TA;

            hra = basic \* hra\_per/100;

            other\_allow = int(input("Enter other allowances: "));

            gross = (basic)\*(no\_of\_day/30) + data + hra + other\_allow;

            itax = 12/100 \* gross;

            pf = 12/100 \* basic;

            esi = 1.75/100 \* gross;

            otherDeductions = int(input("Enter other deductions: "));

            totalDeductions = itax + pf + esi + otherDeductions;

            netsal = gross - totalDeductions;

            ls.append([year, month, ecode, no\_of\_day, da, data, hra, other\_allow, gross, itax, pf, esi, otherDeductions, totalDeductions, netsal]);

            print()

            print()

        for smallList in ls:

            self.salary\_entryInd(smallList[0], smallList[1], smallList[2], smallList[3], smallList[4], smallList[5], smallList[6], smallList[7], smallList[8], smallList[9], smallList[10] ,smallList[11], smallList[12], smallList[13], smallList[14]);

        print("Salary Updated successfully..")

    def show\_payBill(self, year):

        cur = self.con.cursor();

        cur.execute(f"select \* from pay where year = {year}")

        for row in cur:

            print();

            print(f"Year: {row[0]}")

            print(f"Month: {row[1]}")

            print(f"ECode: {row[2]}")

            print(f"No of Days: {row[3]}")

            print(f"DA: {row[4]}")

            print(f"HRA: {row[5]}")

            print(f"Other Allowance: {row[6]}")

            print(f"Gross: {row[7]}")

            print(f"ITax: {row[8]}")

            print(f"PF: {row[9]}")

            print(f"ESI: {row[10]}")

            print(f"Other Deductions: {row[11]}")

            print(f"Total Deductions: {row[12]}")

            print(f"Net Salary: {row[13]}")

            print()

            print();

    def removeEmployee(self, ecode):

        cur = self.con.cursor();

        cur.execute(f"delete from pay where ecode = {ecode}");

        cur.execute(f"delete from emp where ecode = {ecode}");

        self.con.commit();

        print("Data deleted successfully..");

from DBHelper import DBHelper;

import time

def main():

    helper = DBHelper();

    print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Welcome\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

    while (True):

        print("\nenter 0: to add employee details: ")

        print("\nenter 1: show employee details: ")

        print("\nenter 2: fix da and hra rates: ")

        print("\nenter 3: show current da and hra rates: ")

        print("\nenter 4: paybill entry ")

        print("\nenter 5: show paybill")

        print("\nenter 6: to delete employee details(both from payslip and emp database)")

        print("\nenter 7: exit\n")

        choic = int(input("Enter your choice: "))

        try:

            if choic == 0:

                print("Loading...\n")

                time.sleep(0.3)

                ecode = int(input("Enter ecode: "));

                fname = input("Enter full name: ");

                lname = input("Enter last name: ");

                desgination = input("Enter Designation: ");

                gender = input("Enter gender M/F: ");

                DOB = input("Enter dob: ");

                DOJ = input("Enter date of joining: ")

                mob = input("Enter mob: ");

                pan = input("Enter pan number: ");

                acno = input("Enter Account number: ");

                ifsc = input("Enter ifsc code: ");

                basic = int(input("Enter basic salary: "));

                ta = int(input("Enter ta: "))

                helper.input\_employee(ecode, fname, lname, desgination, gender, DOB, DOJ, mob, pan, acno, ifsc, basic, ta);

            elif choic == 1:

                print("Loading...\n")

                time.sleep(0.3)

                helper.display\_AllEmp();

            elif choic == 2:

                print("Loading...\n")

                time.sleep(0.3)

                daPercent = int(input("Enter da percentage: "))

                hraPercent = int(input("Enter HRA percentage: "))

                helper.percentage\_setter(daPercent, hraPercent);

            elif choic == 3:

                print("Loading...\n")

                time.sleep(0.3)

                helper.show\_rates();

            elif choic == 4:

                print("Loading...\n")

                time.sleep(0.3)

                helper.salary\_entry();

            elif choic == 5:

                print("Loading...\n")

                time.sleep(0.3)

                year = int(input("Enter the year: "))

                helper.show\_payBill(year);

            elif choic == 6:

                print("Loading...\n")

                time.sleep(0.3)

                ecode = int(input("Enter the ecode of the employee whose data u want to delete: "))

                helper.removeEmployee(ecode);

            elif choic == 7:

                print("Exiting...");

                break;

            else:

                print("Wrong choice.. try again..")

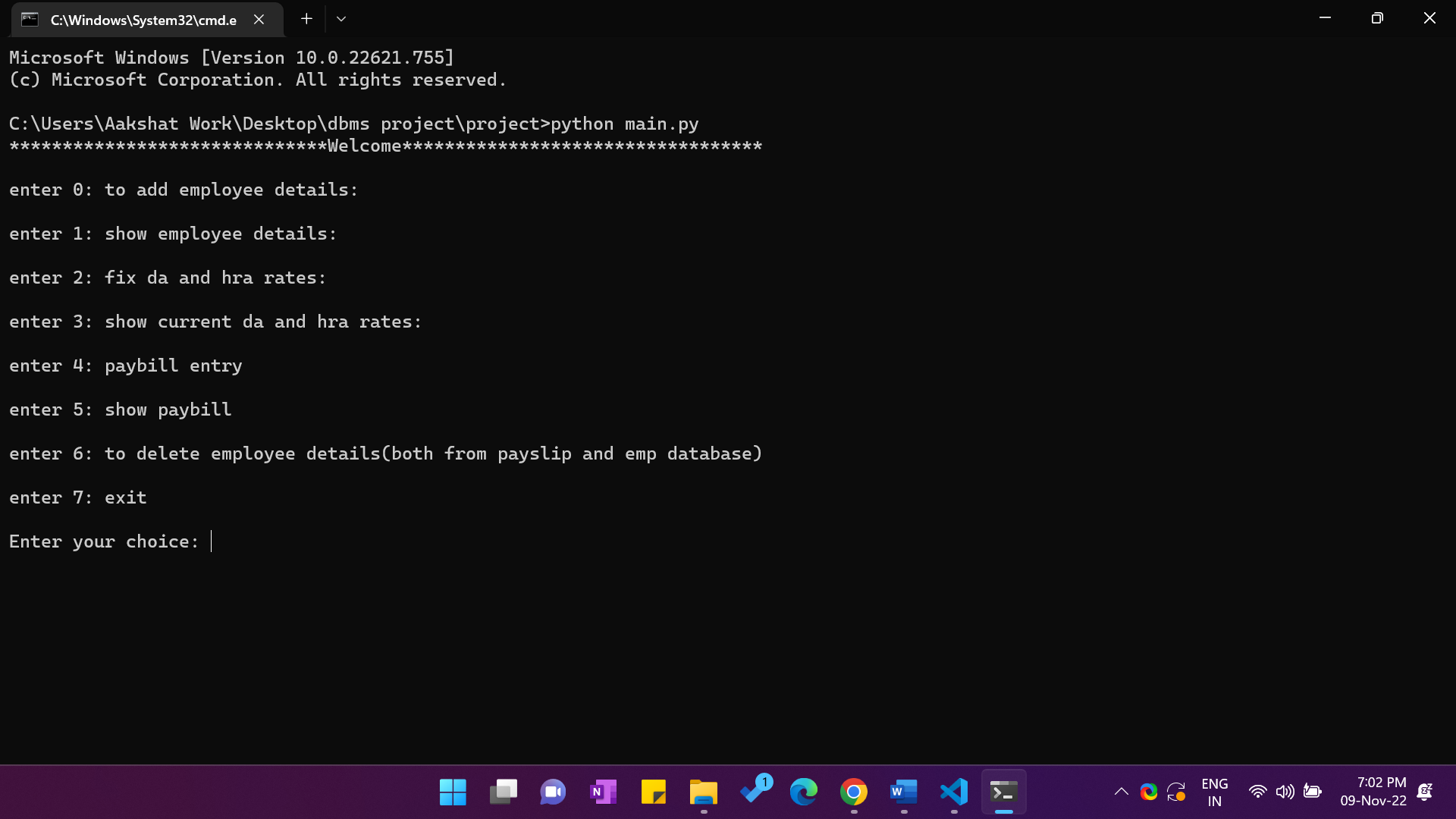
        except Exception as e:

            print(e);

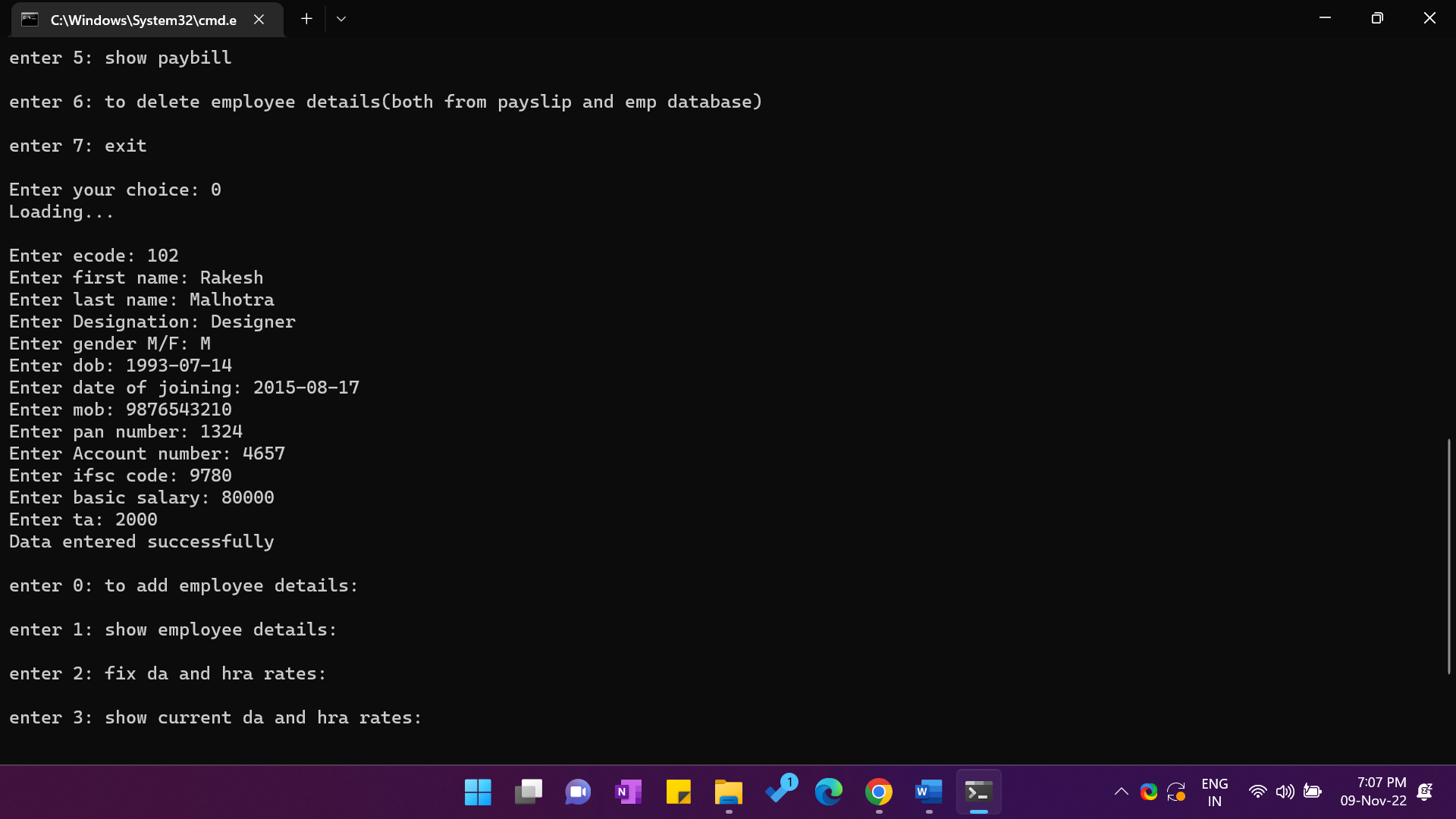
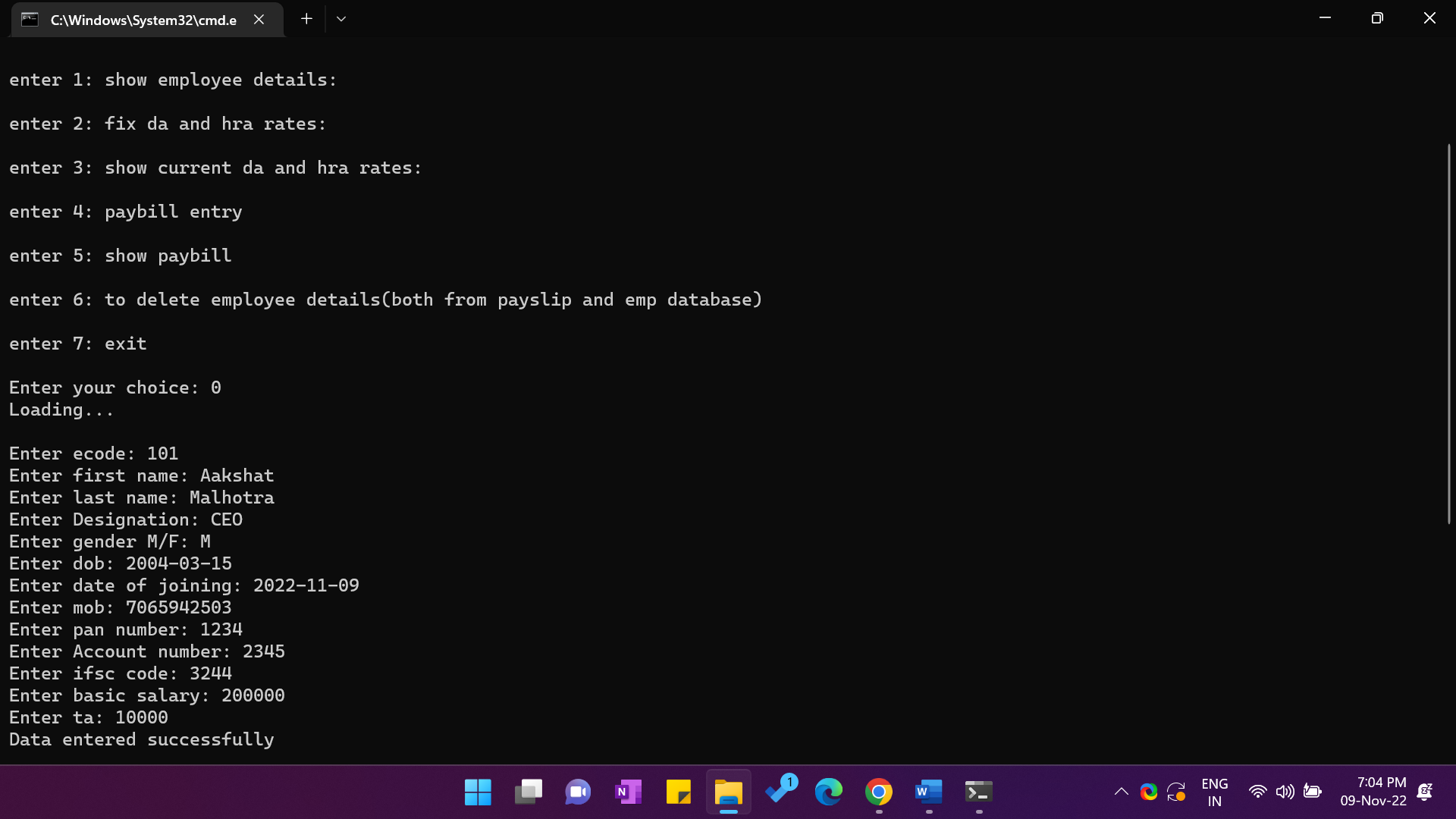
if (\_\_name\_\_ == "\_\_main\_\_"):

    main();

# Working Screenshots



## Entering employee details



## Showing Employee Details

## Setting DA and HRA percentages

## Showing DA and HRA percentage

## Printing paybill for a particular year

## Deleting employee details