Employee Management System

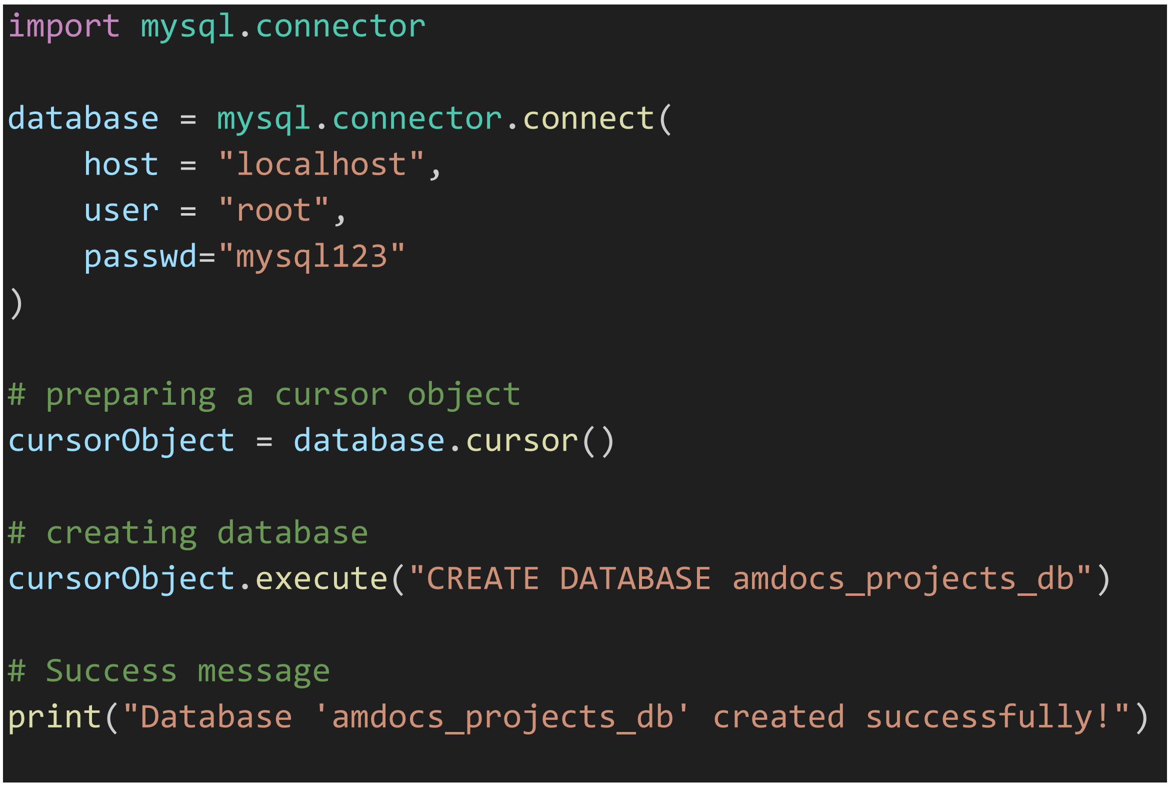
Introduction

The **Employee Management System** is a software solution that helps businesses manage employee data efficiently. It allows organizations to add, select, update, and delete employee information, such as personal details like name, date of birth, address, contact in a centralized database, MySQL.

The system streamlines HR processes like adding, updating, and removing employees, ensuring accurate record-keeping and improving operational efficiency.

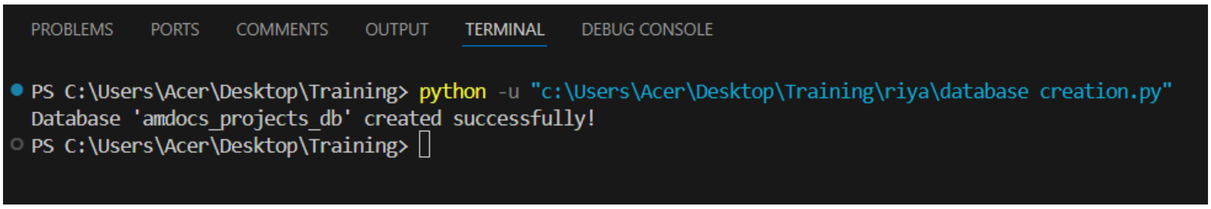
Codes and Outputs

1. **Database Creation:**



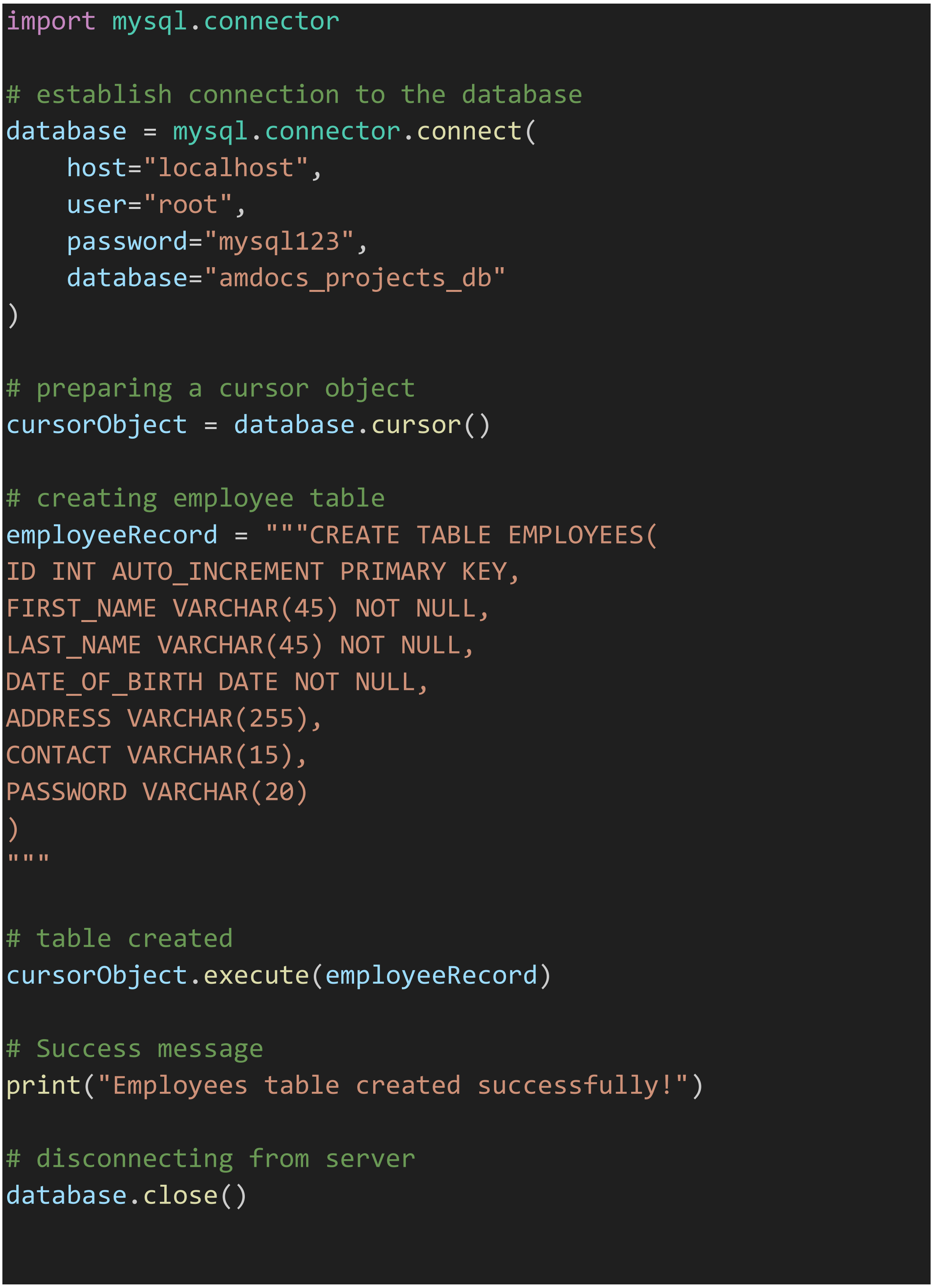
```python
import mysql.connector

database = mysql.connector.connect(
    host = "localhost",
    user = "root",
    passwd="mysql123"
)

# preparing a cursor object
cursorObject = database.cursor()

# creating database
cursorObject.execute("CREATE DATABASE amdocs_projects_db")

# Success message
print("Database 'amdocs_projects_db' created successfully!")
```

Output:



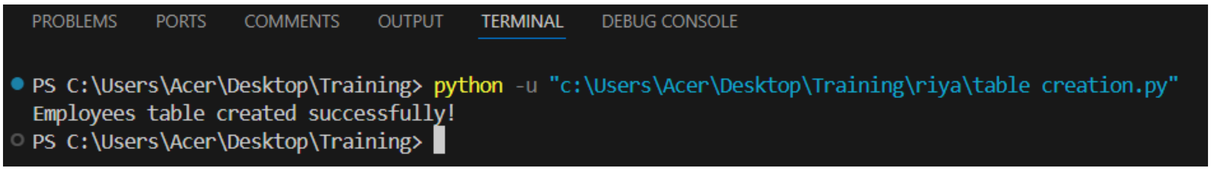
```
PS C:\Users\Acer\Desktop\Training> python -u "c:\Users\Acer\Desktop\Training\riya\database creation.py"
Database 'amdocs_projects_db' created successfully!
PS C:\Users\Acer\Desktop\Training> 
```

2. Employee table creation:



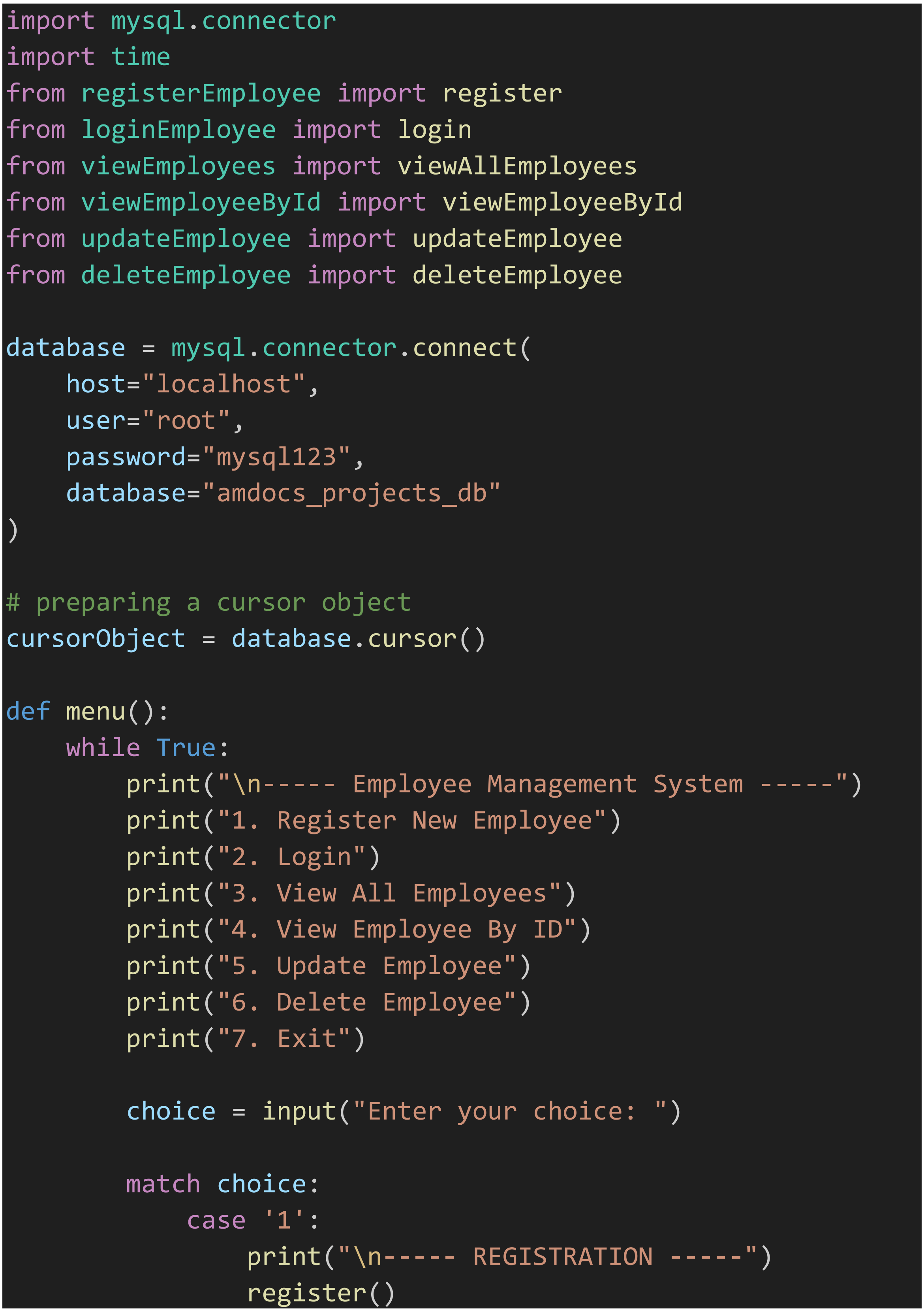
```python
import mysql.connector

# establish connection to the database
database = mysql.connector.connect(
    host="localhost",
    user="root",
    password="mysql123",
    database="amdocs_projects_db"
)

# preparing a cursor object
cursorObject = database.cursor()

# creating employee table
employeeRecord = """CREATE TABLE EMPLOYEES(
ID INT AUTO_INCREMENT PRIMARY KEY,
FIRST_NAME VARCHAR(45) NOT NULL,
LAST_NAME VARCHAR(45) NOT NULL,
DATE_OF_BIRTH DATE NOT NULL,
ADDRESS VARCHAR(255),
CONTACT VARCHAR(15),
PASSWORD VARCHAR(20)
)
"""

# table created
cursorObject.execute(employeeRecord)

# Success message
print("Employees table created successfully!")

# disconnecting from server
database.close()
```

Output:

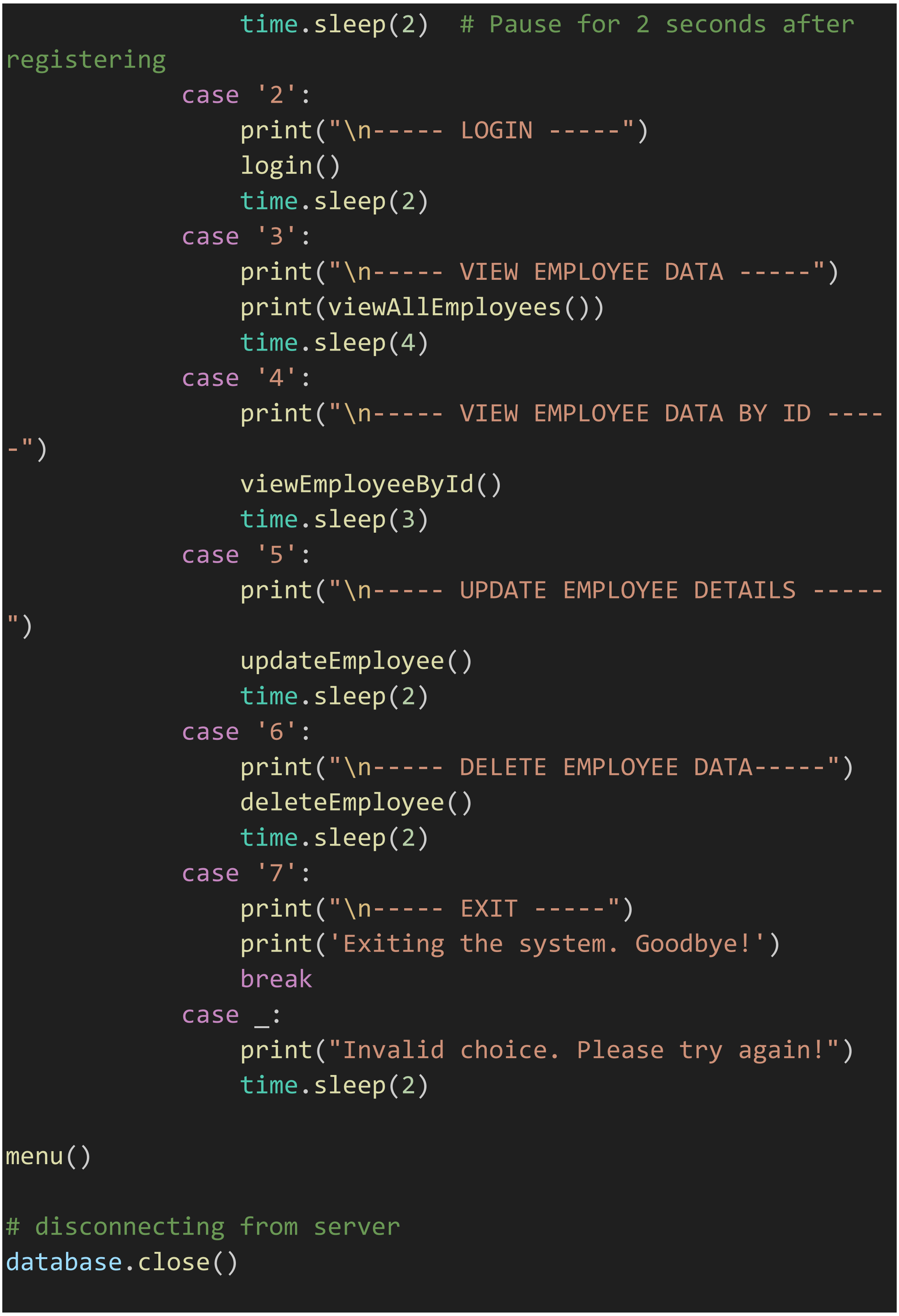


```
● PS C:\Users\Acer\Desktop\Training> python -u "c:\Users\Acer\Desktop\Training\riya\table creation.py"
  Employees table created successfully!
○ PS C:\Users\Acer\Desktop\Training> █
```

3. **Project menu**

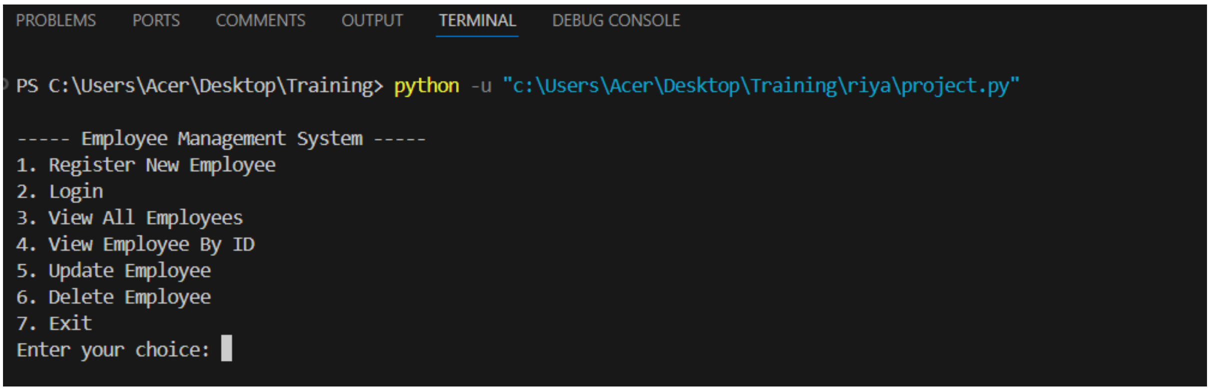


```
import mysql.connector
import time
from registerEmployee import register
from loginEmployee import login
from viewEmployees import viewAllEmployees
from viewEmployeeById import viewEmployeeById
from updateEmployee import updateEmployee
from deleteEmployee import deleteEmployee

database = mysql.connector.connect(
    host="localhost",
    user="root",
    password="mysql123",
    database="amdocs_projects_db"
)

# preparing a cursor object
cursorObject = database.cursor()

def menu():
    while True:
        print("\n----- Employee Management System -----")
        print("1. Register New Employee")
        print("2. Login")
        print("3. View All Employees")
        print("4. View Employee By ID")
        print("5. Update Employee")
        print("6. Delete Employee")
        print("7. Exit")

        choice = input("Enter your choice: ")

        match choice:
            case '1':
                print("\n----- REGISTRATION -----")
                register()
```



```
                time.sleep(2)  # Pause for 2 seconds after 
registering
            case '2':
                print("\n----- LOGIN -----")
                login()
                time.sleep(2)
            case '3':
                print("\n----- VIEW EMPLOYEE DATA -----")
                print(viewAllEmployees())
                time.sleep(4)
            case '4':
                print("\n----- VIEW EMPLOYEE DATA BY ID ----
-")
                viewEmployeeById()
                time.sleep(3)
            case '5':
                print("\n----- UPDATE EMPLOYEE DETAILS -----
")
                updateEmployee()
                time.sleep(2)
            case '6':
                print("\n----- DELETE EMPLOYEE DATA-----")
                deleteEmployee()
                time.sleep(2)
            case '7':
                print("\n----- EXIT -----")
                print('Exiting the system. Goodbye!')
                break
            case _:
                print("Invalid choice. Please try again!")
                time.sleep(2)

menu()

# disconnecting from server
database.close()
```

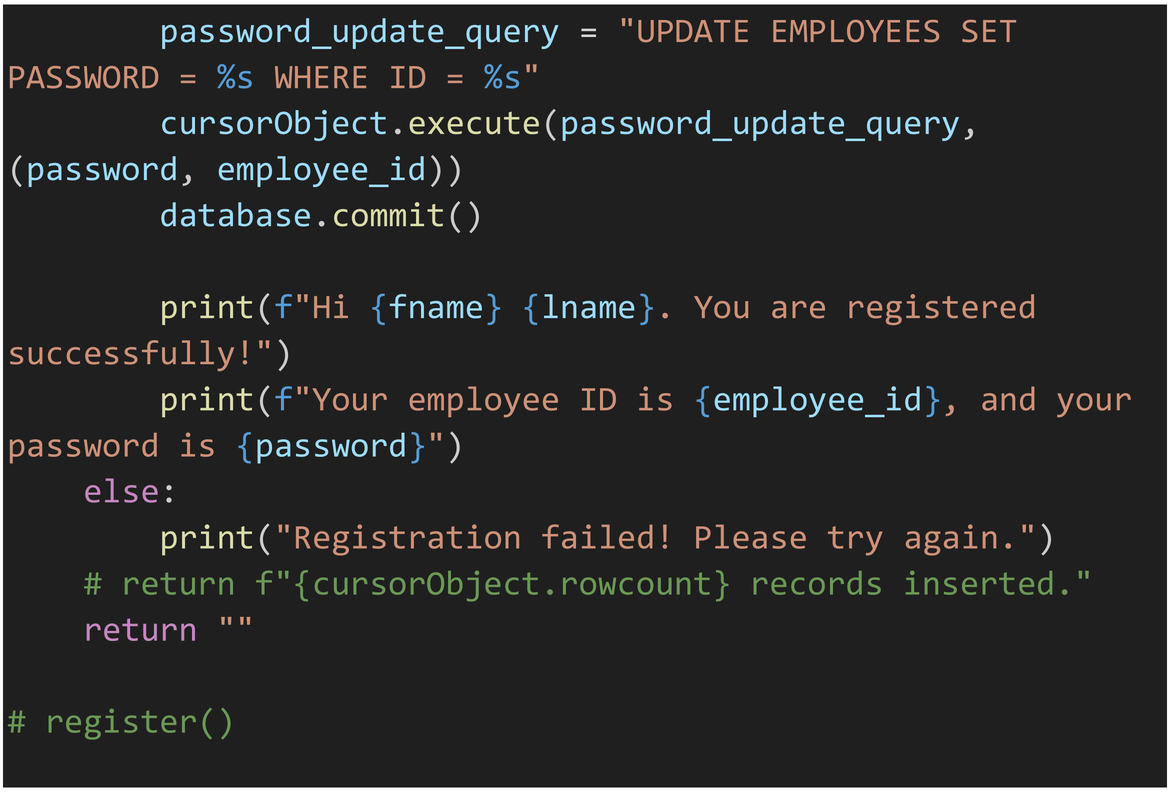
Output:



```
PS C:\Users\Acer\Desktop\Training> python -u "c:\Users\Acer\Desktop\Training\riya\project.py"

----- Employee Management System -----
1. Register New Employee
2. Login
3. View All Employees
4. View Employee By ID
5. Update Employee
6. Delete Employee
7. Exit
Enter your choice: 
```

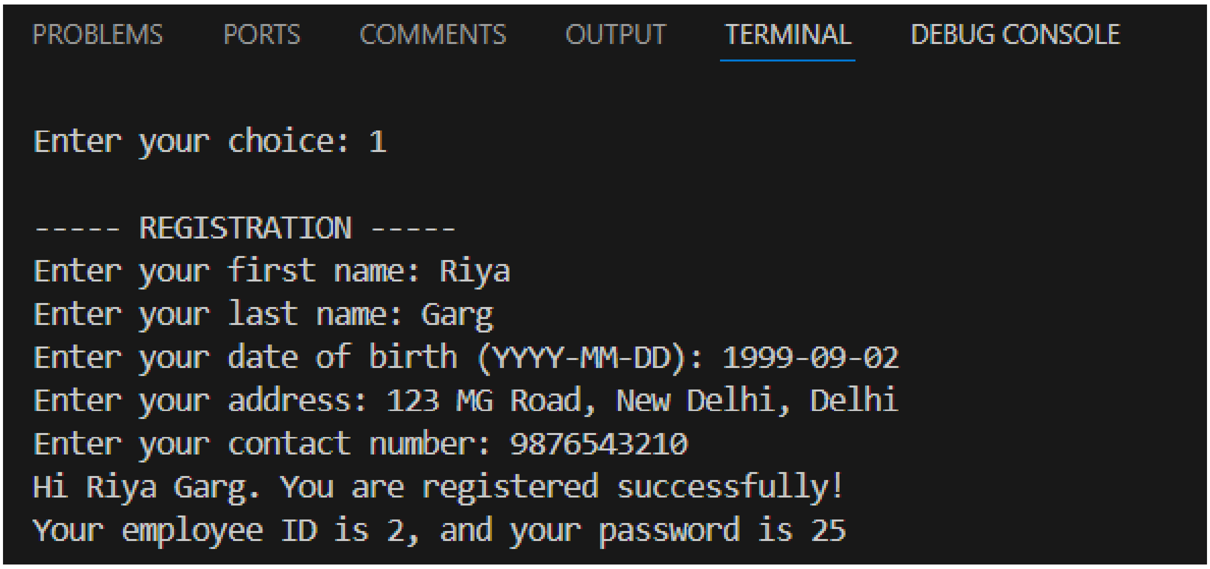
4. Register new employee

```
import mysql.connector

database = mysql.connector.connect(
    host="localhost",
    user="root",
    password="mysql123",
    database="amdocs_projects_db"
)

# preparing a cursor object
cursorObject = database.cursor()

def register():
    # Get employee data from user input
    fname = input("Enter your first name: ")
    lname = input("Enter your last name: ")
    dob = input("Enter your date of birth (YYYY-MM-DD): ")
    address = input("Enter your address: ")
    contact = input("Enter your contact number: ")

    # SQL query for inserting data into the EMPLOYEES table
    sql = "INSERT INTO EMPLOYEES(FIRST_NAME, LAST_NAME, 
DATE_OF_BIRTH, ADDRESS, CONTACT) \
    VALUES(%s, %s, %s, %s, %s)"
    val = (fname, lname, dob, address, contact)

    # execute the query and commit changes
    cursorObject.execute(sql,val)
    database.commit()

    if cursorObject.rowcount:
        # retrieving the employee id of the newly inserted 
employee
        employee_id = cursorObject.lastrowid
        password = employee_id * 10 + 5

        # Updating the EMPLOYEES table with the generated 
password
```



```
        password_update_query = "UPDATE EMPLOYEES SET
PASSWORD = %s WHERE ID = %s"
        cursorObject.execute(password_update_query,
(password, employee_id))
        database.commit()

        print(f"Hi {fname} {lname}. You are registered
successfully!")
        print(f"Your employee ID is {employee_id}, and your
password is {password}")
    else:
        print("Registration failed! Please try again.")
    # return f"{cursorObject.rowcount} records inserted."
    return ""

# register()
```

Output:

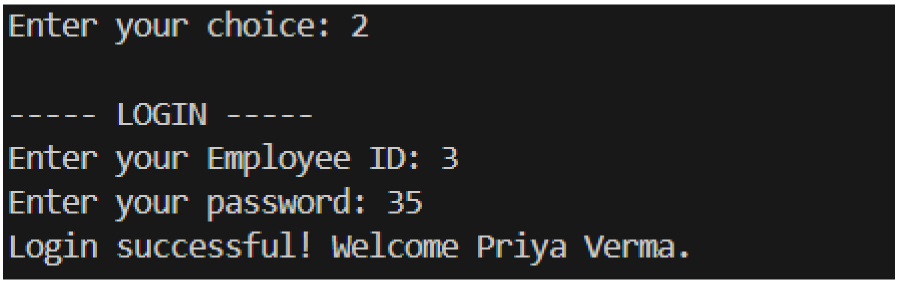


```
Enter your choice: 1

----- REGISTRATION -----
Enter your first name: Riya
Enter your last name: Garg
Enter your date of birth (YYYY-MM-DD): 1999-09-02
Enter your address: 123 MG Road, New Delhi, Delhi
Enter your contact number: 9876543210
Hi Riya Garg. You are registered successfully!
Your employee ID is 2, and your password is 25
```

5. Login employee

```python
import mysql.connector

# Establishing the database connection
database = mysql.connector.connect(
    host="localhost",
    user="root",
    password="mysql123",
    database="amdocs_projects_db"
)

# preparing a cursor object
cursorObject = database.cursor()

def login():
    employee_id = input("Enter your Employee ID: ")
    password = input("Enter your password: ")

    # query to check if id and password match
    query = "SELECT * FROM EMPLOYEES WHERE ID = %s AND PASSWORD = %s"
    cursorObject.execute(query, (employee_id, password))

    result = cursorObject.fetchone()

    if result:
        # Assuming result[1] is first name and result[2] is last name
        print(f"Login successful! Welcome {result[1]} {result[2]}.")
    else:
        print("Invalid Employee ID or Password. Please try again!")

    return ""

# login()
```

Output:

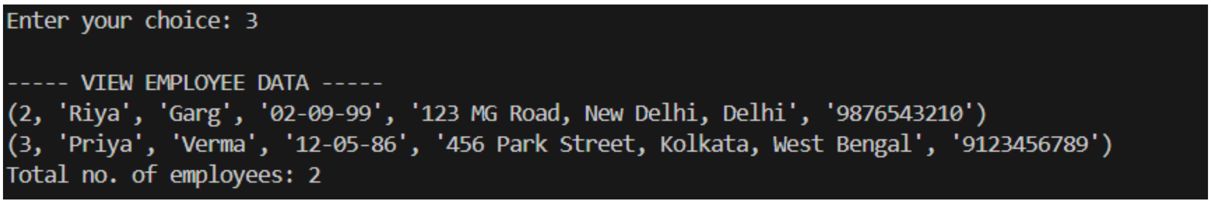


```
Enter your choice: 2

----- LOGIN -----
Enter your Employee ID: 3
Enter your password: 35
Login successful! Welcome Priya Verma.
```

6. **View all employees**

```
import mysql.connector

database = mysql.connector.connect(
    host="localhost",
    user="root",
    password="mysql123",
    database="amdocs_projects_db"
)

# preparing a cursor object
cursorObject = database.cursor()

def viewAllEmployees():
    # SQL query to read data from the EMPLOYEES table
    query = "SELECT * FROM EMPLOYEES"
    cursorObject.execute(query)

    result = cursorObject.fetchall()

    for row in result:
        # Formatting the date to 'dd-mm-yyyy'
        formatted_dob = row[3].strftime("%d-%m-%y")
        print((row[0], row[1], row[2], formatted_dob,
row[4], row[5]))

    totalEmployees = len(result)
    return f"Total no. of employees: {totalEmployees}"

# totalEmployees = viewAllEmployees()
# print(totalEmployees)
```

Output:

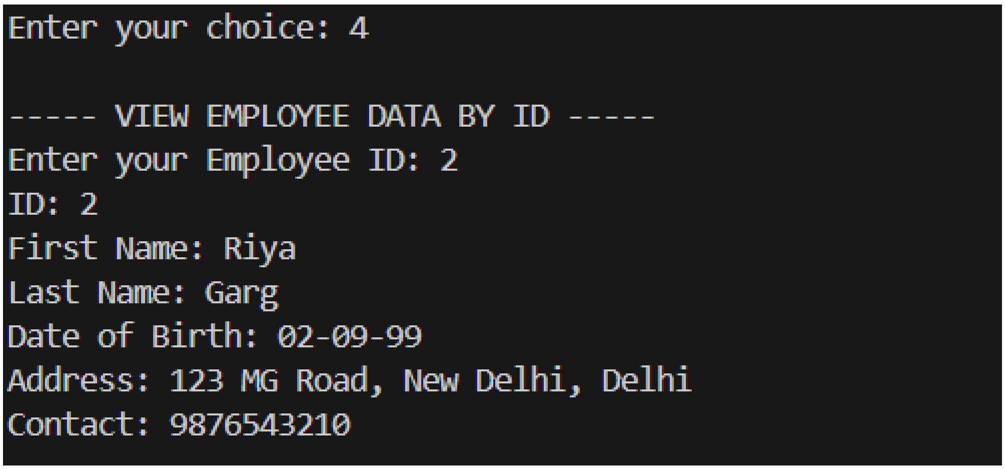


```
Enter your choice: 3

----- VIEW EMPLOYEE DATA -----
(2, 'Riya', 'Garg', '02-09-99', '123 MG Road, New Delhi, Delhi', '9876543210')
(3, 'Priya', 'Verma', '12-05-86', '456 Park Street, Kolkata, West Bengal', '9123456789')
Total no. of employees: 2
```

7. **View employee by id**

```
import mysql.connector

database = mysql.connector.connect(
    host="localhost",
    user="root",
    password="mysql123",
    database="amdocs_projects_db"
)

# preparing a cursor object
cursorObject = database.cursor()

def viewEmployeeById():

    employee_id = input("Enter your Employee ID: ")
    # SQL query to read data from the EMPLOYEES table based
on employee ID
    query = "SELECT * FROM EMPLOYEES WHERE ID = %s"
    cursorObject.execute(query, (employee_id,))

    #  fetch the result
    result = cursorObject.fetchone()

    if result:
        formatted_dob = result[3].strftime("%d-%m-%y")
        # Displaying employee details
        print(f"ID: {result[0]}")
        print(f"First Name: {result[1]}")
        print(f"Last Name: {result[2]}")
        print(f"Date of Birth: {formatted_dob}")
        print(f"Address: {result[4]}")
        print(f"Contact: {result[5]}")
    else:
        print(f"No employee found with ID: {employee_id}")

    return ""

# viewEmployeeById()
```

Output:

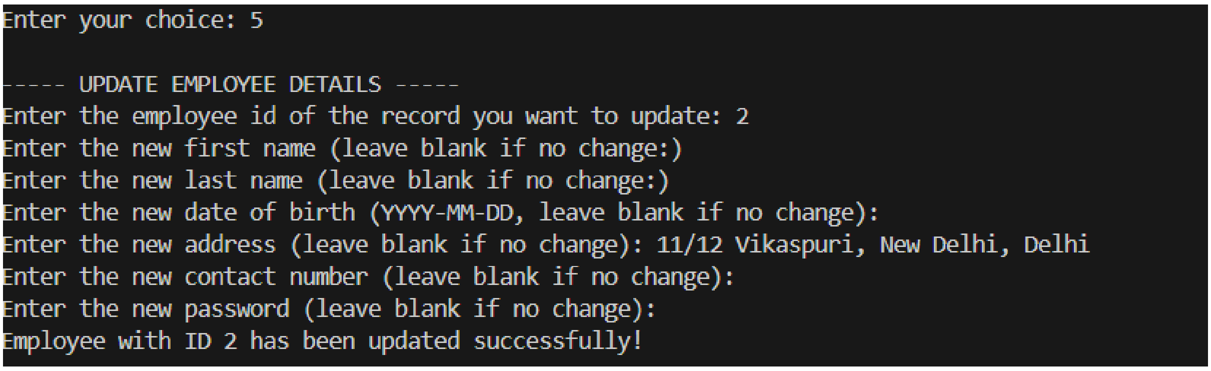


```
Enter your choice: 4

----- VIEW EMPLOYEE DATA BY ID -----
Enter your Employee ID: 2
ID: 2
First Name: Riya
Last Name: Garg
Date of Birth: 02-09-99
Address: 123 MG Road, New Delhi, Delhi
Contact: 9876543210
```

8. **Update employee details**

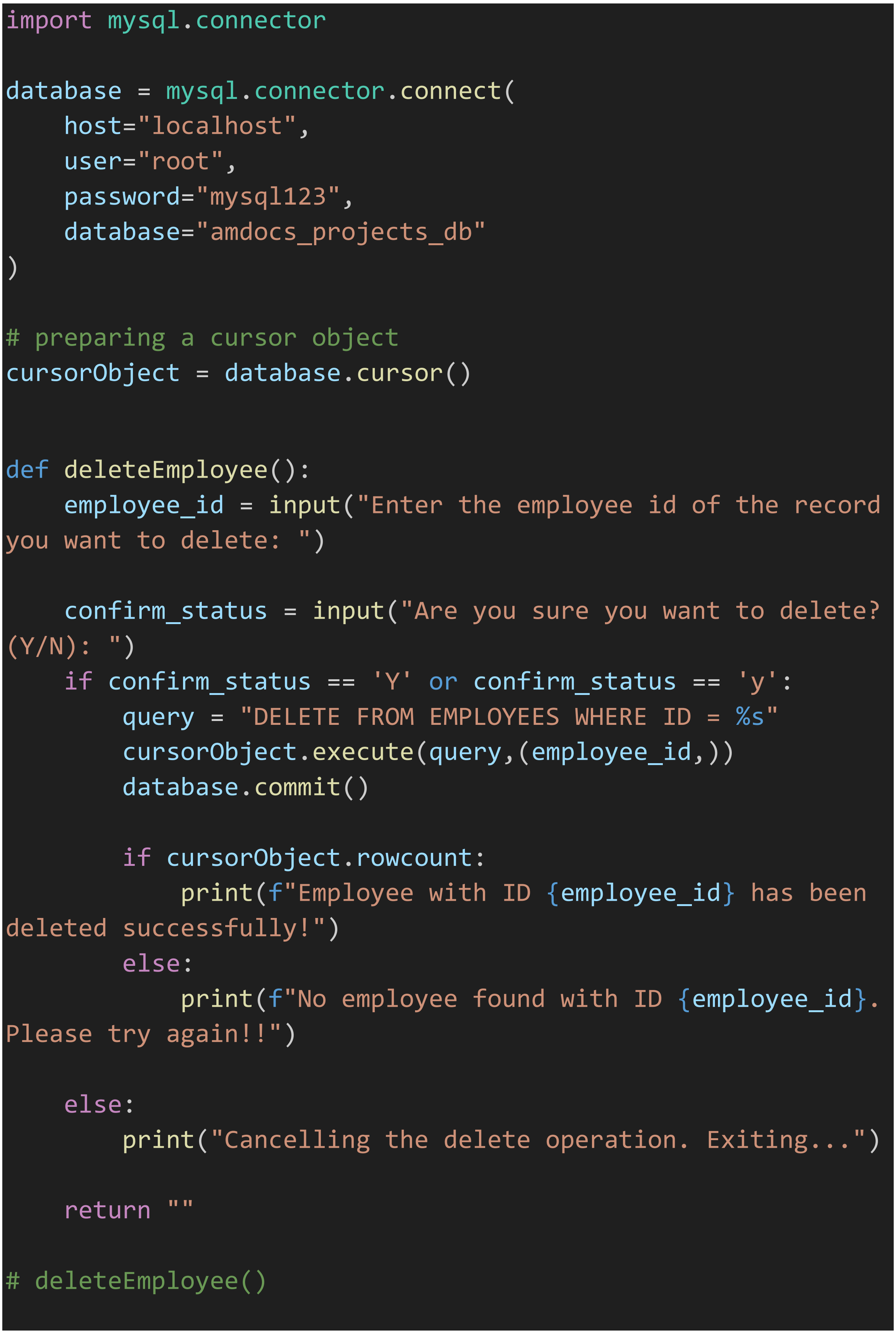
```python
import mysql.connector

database = mysql.connector.connect(
    host="localhost",
    user="root",
    password="mysql123",
    database="amdocs_projects_db"
)

# preparing a cursor object
cursorObject = database.cursor()

def updateEmployee():
    employee_id = input("Enter the employee id of the record you want to update: ")

    #  get new employee data from user input
    fname = input("Enter the new first name (leave blank if no change:) ")
    lname = input("Enter the new last name (leave blank if no change:) ")
    dob = input("Enter the new date of birth (YYYY-MM-DD, leave blank if no change): ")
    address = input("Enter the new address (leave blank if no change): ")
    contact = input("Enter the new contact number (leave blank if no change): ")
    password = input("Enter the new password (leave blank if no change): ")

    # dynamic sql query based on user input
    set_clause = []
    parameters = []

    if fname:
        set_clause.append("FIRST_NAME = %s")
        parameters.append(fname)
    if lname:
        set_clause.append("LAST_NAME = %s")
        parameters.append(lname)
    if dob:
        set_clause.append("DATE_OF_BIRTH = %s")
        parameters.append(dob)
    if address:
        set_clause.append("ADDRESS = %s")
        parameters.append(address)
    if contact:
        set_clause.append("CONTACT = %s")
        parameters.append(contact)
    if password:
        set_clause.append("PASSWORD = %s")
        parameters.append(password)

    # ensuring there's atleast one column to update
    if not set_clause:
        print("No updates provided. Exiting...")
        return ""

    set_clause_str = ", ".join(set_clause)
    query = f"UPDATE EMPLOYEES SET {set_clause_str} WHERE ID
= %s"
    parameters.append(employee_id)

    # execute the query and commit changes
    cursorObject.execute(query,tuple(parameters))
    database.commit()

    if cursorObject.rowcount:
        print(f"Employee with ID {employee_id} has been
updated successfully!")
    else:
        print(f"No employee found with ID {employee_id} or
no changes were made")

    return ""

# updateEmployee()
```

Output:



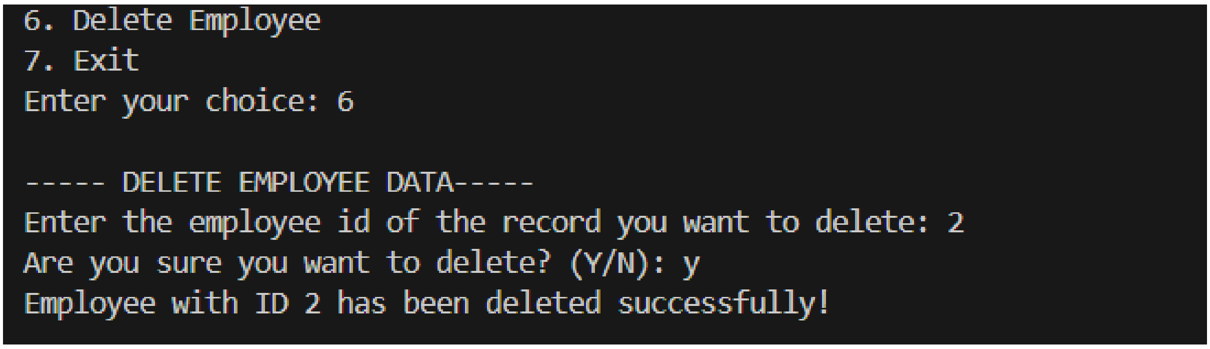
```
Enter your choice: 5

----- UPDATE EMPLOYEE DETAILS -----
Enter the employee id of the record you want to update: 2
Enter the new first name (leave blank if no change:)
Enter the new last name (leave blank if no change:)
Enter the new date of birth (YYYY-MM-DD, leave blank if no change):
Enter the new address (leave blank if no change): 11/12 Vikaspuri, New Delhi, Delhi
Enter the new contact number (leave blank if no change):
Enter the new password (leave blank if no change):
Employee with ID 2 has been updated successfully!
```

9. Delete employee



```
import mysql.connector

database = mysql.connector.connect(
    host="localhost",
    user="root",
    password="mysql123",
    database="amdocs_projects_db"
)

# preparing a cursor object
cursorObject = database.cursor()


def deleteEmployee():
    employee_id = input("Enter the employee id of the record you want to delete: ")

    confirm_status = input("Are you sure you want to delete? (Y/N): ")
    if confirm_status == 'Y' or confirm_status == 'y':
        query = "DELETE FROM EMPLOYEES WHERE ID = %s"
        cursorObject.execute(query,(employee_id,))
        database.commit()

        if cursorObject.rowcount:
            print(f"Employee with ID {employee_id} has been deleted successfully!")
        else:
            print(f"No employee found with ID {employee_id}. Please try again!!")

    else:
        print("Cancelling the delete operation. Exiting...")

    return ""

# deleteEmployee()
```

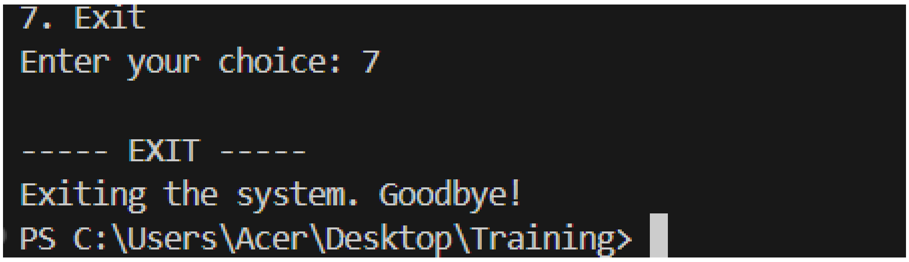
Output:



```
6. Delete Employee
7. Exit
Enter your choice: 6

----- DELETE EMPLOYEE DATA-----
Enter the employee id of the record you want to delete: 2
Are you sure you want to delete? (Y/N): y
Employee with ID 2 has been deleted successfully!
```

10. Exiting the system

Output:



```
7. Exit
Enter your choice: 7

----- EXIT -----
Exiting the system. Goodbye!
PS C:\Users\Acer\Desktop\Training>
```