```python
import mysql.connector
import getpass
import os

# Helper function to clear the screen
def clear_screen():
    os.system('cls' if os.name == 'nt' else 'clear')

# Helper function to center text
def center_text(text, width=60):
    return text.center(width)

# Helper function to create a styled header
def print_header(title):
    print()
    print("=" * 60)
    print(center_text(title.upper()))
    print("=" * 60)

# Helper function for a simple menu border
def print_border():
    print("-" * 60)


# Database connection
def connect_db():
     return mysql.connector.connect(
        host="localhost",
        user="root",
        password="root",
        database="employee_management"
    )


# Class for Admin
class Admin:
    def __init__(self):
        self.username = "admin"
        self.password = "admin123"  # Hardcoded
```

```
    def login(self):
        clear_screen()
        print_header("Admin Login")
        username = input(center_text("Enter Username: "))
        password = getpass.getpass(center_text("Enter Password:
"))

        if username == self.username and password ==
self.password:
            print(center_text("Login successful!"))
            self.admin_menu()
        else:
            print(center_text("Invalid credentials. Please try
again."))

    def admin_menu(self):
        while True:
            clear_screen()
            print_header("Admin Menu")
            print(center_text("1. Insert Employee"))
            print(center_text("2. Update Employee"))
            print(center_text("3. Delete Employee"))
            print(center_text("4. View Employees"))
            print(center_text("5. Logout"))
            print_border()

            choice = input(center_text("Enter your choice (1-
5): "))

            if choice == '1':
                self.insert_employee()
            elif choice == '2':
                self.update_employee()
            elif choice == '3':
                self.delete_employee()
            elif choice == '4':
                self.view_employees()
            elif choice == '5':
                break
            else:
```

```
                print(center_text("Invalid choice, please try
again."))

    def insert_employee(self):
        conn = connect_db()
        cursor = conn.cursor()

        clear_screen()
        print_header("Insert New Employee")
        name = input(center_text("Enter Name: "))
        age = input(center_text("Enter Age: "))
        department = input(center_text("Enter Department: "))
        username = input(center_text("Enter Username: "))
        password = getpass.getpass(center_text("Enter Password:
"))

        try:
            query = "INSERT INTO employees (name, age,
department, username, password) VALUES (%s, %s, %s, %s, %s)"
            cursor.execute(query, (name, age, department,
username, password))
            conn.commit()
            print(center_text("Employee added successfully!"))
        except mysql.connector.Error as err:
            print(center_text(f"Error: {err}"))
        finally:
            cursor.close()
            conn.close()
        # Wait for user input before clearing the screen
        input(center_text("\nPress Enter to return to the
menu..."))

    def update_employee(self):
        conn = connect_db()
        cursor = conn.cursor()

        emp_id = input(center_text("\nEnter Employee ID to
update: "))
```

```
        cursor.execute("SELECT * FROM employees WHERE emp_id =
%s", (emp_id,))
        employee = cursor.fetchone()

        if employee:
            print(("Leave field blank to keep current value."))
            name = input((f"Update Name (current:
{employee[1]}): ")) or employee[1]
            age = input((f"Update Age (current: {employee[2]}):
")) or employee[2]
            department = input((f"Update Department (current:
{employee[3]}): ")) or employee[3]

            query = "UPDATE employees SET name = %s, age = %s,
department = %s WHERE emp_id = %s"
            cursor.execute(query, (name, age, department,
emp_id))
            conn.commit()
            print(center_text("Employee updated
successfully!"))
        else:
            print(center_text("Employee not found."))
        # Wait for user input before clearing the screen
        input(center_text("\nPress Enter to return to the
menu..."))
        cursor.close()
        conn.close()

    def delete_employee(self):
        conn = connect_db()
        cursor = conn.cursor()

        emp_id = input(center_text("\nEnter Employee ID to
delete: "))

        cursor.execute("DELETE FROM employees WHERE emp_id =
%s", (emp_id,))
        conn.commit()

        if cursor.rowcount > 0:
```

```
            print(center_text("Employee deleted
successfully."))
        else:
            print(center_text("Employee not found."))

        # Wait for user input before clearing the screen
        input(center_text("\nPress Enter to return to the
menu..."))
        cursor.close()
        conn.close()

    def view_employees(self):
        conn = connect_db()
        cursor = conn.cursor()

        clear_screen()
        print_header("View Employees")
        print(center_text("1. View All Employees"))
        print(center_text("2. View Employee by ID"))
        print_border()

        choice = input(center_text("Enter your choice (1-2):
"))

        if choice == '1':
            cursor.execute("SELECT * FROM employees")
            employees = cursor.fetchall()
            if employees:
                for emp in employees:
                    print(center_text(f"ID: {emp[0]}, Name:
{emp[1]}, Age: {emp[2]}, Department: {emp[3]}"))
            else:
                print(center_text("No employees to display."))
        elif choice == '2':
            emp_id = input(center_text("Enter Employee ID: "))
            cursor.execute("SELECT * FROM employees WHERE
emp_id = %s", (emp_id,))
            employee = cursor.fetchone()
            if employee:
```

```
                print(center_text(f"ID: {employee[0]}, Name:
{employee[1]}, Age: {employee[2]}, Department: {employee[3]}"))
            else:
                print(center_text("Employee not found."))
        else:
            print(center_text("Invalid choice."))

        cursor.close()
        conn.close()

        # Wait for user input before clearing the screen
        input(center_text("\nPress Enter to return to the
menu..."))


# Class for Employee
class Employee:
    def register(self):
        conn = connect_db()
        cursor = conn.cursor()

        clear_screen()
        print_header("Employee Registration")
        name = input(("Enter Name: "))
        age = input(("Enter Age: "))
        department = input(("Enter Department: "))
        username = input(("Enter Username: "))
        password = getpass.getpass(("Enter Password: "))

        try:
            query = "INSERT INTO employees (name, age,
department, username, password) VALUES (%s, %s, %s, %s, %s)"
            cursor.execute(query, (name, age, department,
username, password))
            conn.commit()
            emp_id = cursor.lastrowid  # Get the inserted
employee ID
            print(center_text("Employee registered
successfully!"))
            self.employee_menu(emp_id)
```

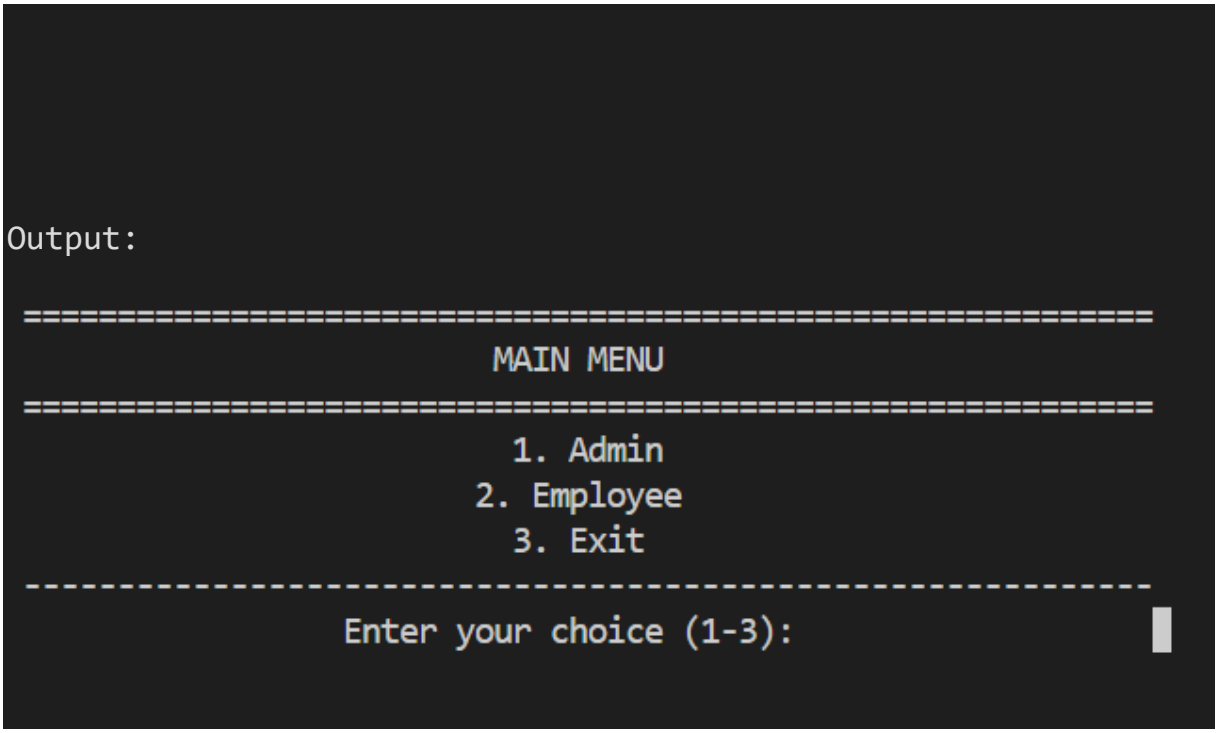
```
        except mysql.connector.Error as err:
            print(center_text(f"Error: {err}"))
        finally:
            cursor.close()
            conn.close()

    def login(self):
        conn = connect_db()
        cursor = conn.cursor()

        clear_screen()
        print_header("Employee Login")
        username = input(center_text("Enter Username: "))
        password = getpass.getpass(center_text("Enter Password:
"))

        cursor.execute("SELECT emp_id FROM employees WHERE
username = %s AND password = %s", (username, password))
        employee = cursor.fetchone()

        if employee:
            print(center_text("Login successful!"))
            self.employee_menu(employee[0])
        else:
            print(center_text("Invalid credentials. Please try
again."))

        cursor.close()
        conn.close()

    def employee_menu(self, emp_id):
        while True:
            clear_screen()
            print_header("Employee Menu")
            print(center_text("1. Update Details"))
            print(center_text("2. View Details"))
            print(center_text("3. Logout"))
            print_border()
```

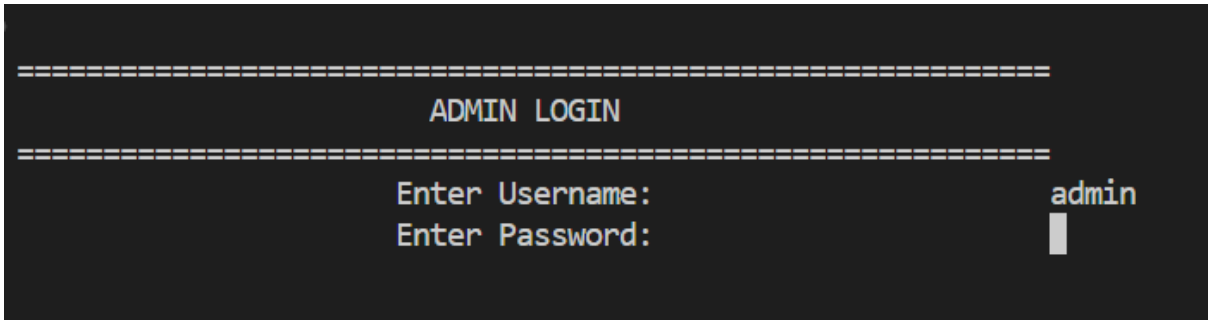
```
            choice = input(center_text("Enter your choice (1-
3): "))

            if choice == '1':
                self.update_employee_details(emp_id)
            elif choice == '2':
                self.view_employee_details(emp_id)
            elif choice == '3':
                break
            else:
                print(center_text("Invalid choice, please try
again."))

    def update_employee_details(self, emp_id):
        conn = connect_db()
        cursor = conn.cursor()

        cursor.execute("SELECT * FROM employees WHERE emp_id =
%s", (emp_id,))
        employee = cursor.fetchone()

        if employee:
            print(center_text("\nUpdate Your Details:"))
            name = input(center_text(f"Update Name (current:
{employee[1]}): ")) or employee[1]
            age = input(center_text(f"Update Age (current:
{employee[2]}): ")) or employee[2]
            department = input(center_text(f"Update Department
(current: {employee[3]}): ")) or employee[3]

            query = "UPDATE employees SET name = %s, age = %s,
department = %s WHERE emp_id = %s"
            cursor.execute(query, (name, age, department,
emp_id))
            conn.commit()
            print(center_text("Details updated successfully!"))
        else:
            print(center_text("Employee not found."))

         # Wait for user input before clearing the screen
```

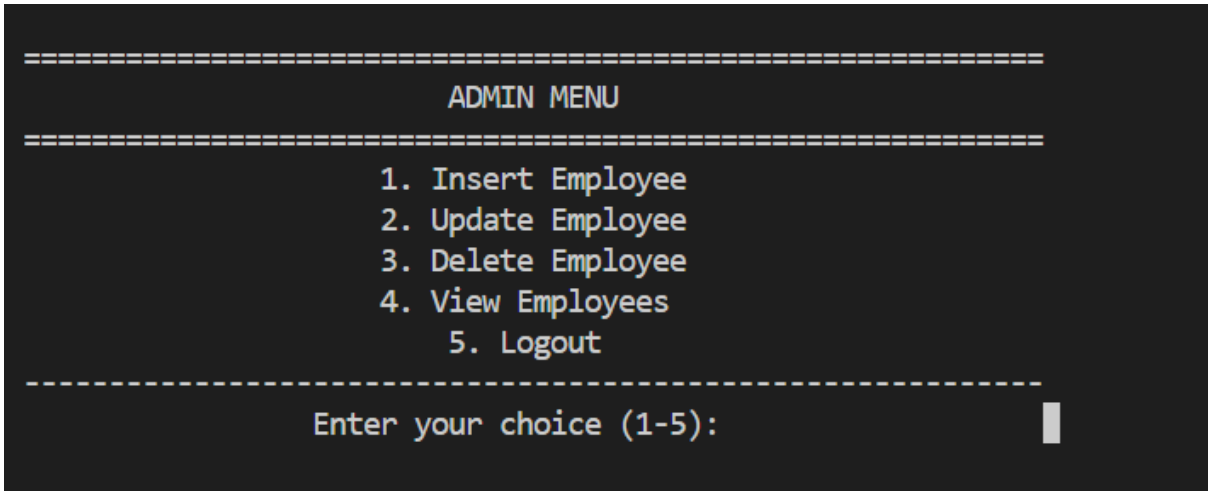
```
        input(center_text("\nPress Enter to return to the
menu..."))
        cursor.close()
        conn.close()

    def view_employee_details(self, emp_id):
        conn = connect_db()
        cursor = conn.cursor()

        cursor.execute("SELECT * FROM employees WHERE emp_id =
%s", (emp_id,))
        employee = cursor.fetchone()

        if employee:
            print(center_text(f"\nYour Details:\nID:
{employee[0]}, Name: {employee[1]}, Age: {employee[2]},
Department: {employee[3]}"))
        else:
            print(center_text("Employee not found."))

        # Wait for user input before clearing the screen
        input(center_text("\nPress Enter to return to the
menu..."))

        cursor.close()
        conn.close()


# Main class to handle the system
class EmployeeManagementSystem:
    def __init__(self):
        self.admin = Admin()
        self.employee = Employee()

    def main_menu(self):
        while True:
            clear_screen()
            print_header("Main Menu")
            print(center_text("1. Admin"))
            print(center_text("2. Employee"))
```

```
            print(center_text("3. Exit"))
            print_border()

            choice = input(center_text("Enter your choice (1-
3): "))

            if choice == '1':
                self.admin.login()
            elif choice == '2':
                self.employee_portal()
            elif choice == '3':
                print(center_text("Exiting the system."))
                break
            else:
                print(center_text("Invalid choice, please try
again."))

    def employee_portal(self):
        clear_screen()
        print_header("Employee Portal")
        print(center_text("1. Register"))
        print(center_text("2. Login"))
        print_border()

        choice = input(center_text("Enter your choice (1-2):
"))

        if choice == '1':
            self.employee.register()
        elif choice == '2':
            self.employee.login()
        else:
            print(center_text("Invalid choice, please try
again."))


# Main entry point
if __name__ == "__main__":
    system = EmployeeManagementSystem()
    system.main_menu()
```

Output:


MAIN MENU
1. Admin
2. Employee
3. Exit
Enter your choice (1-3):


ADMIN LOGIN
Enter Username: admin
Enter Password:


ADMIN MENU
1. Insert Employee
2. Update Employee
3. Delete Employee
4. View Employees
5. Logout
Enter your choice (1-5):

```
============================================================
                    INSERT NEW EMPLOYEE
============================================================
                       Enter Name:                          Steve Rogers
                       Enter Age:                           26
                    Enter Department:                       Sales
                     Enter Username:                        captain
                     Enter Password:
                Employee added successfully!
```

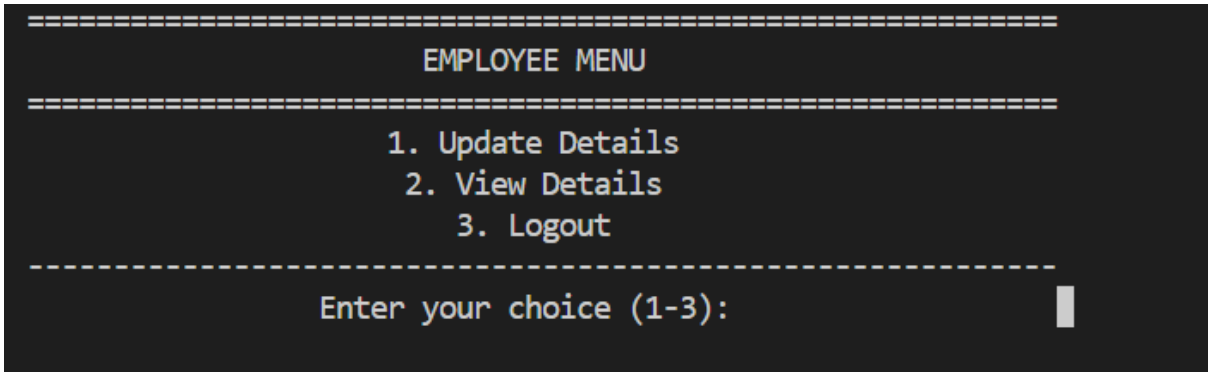
```
                  Enter your choice (1-5):                  2

Enter Employee ID to update:                5
          Leave field blank to keep current value.
           Update Name (current: Steve Rogers):
                 Update Age (current: 26):                  27
            Update Department (current: Sales):
               Employee updated successfully!

Press Enter to return to the menu...
```

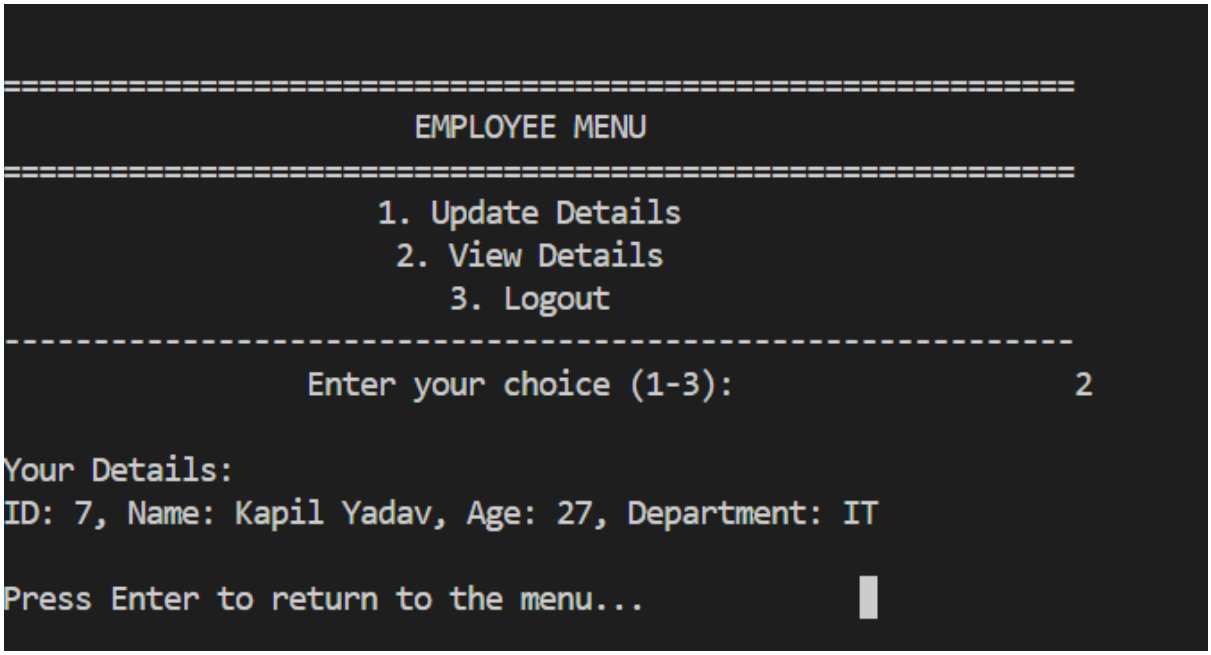
```
============================================================
                       VIEW EMPLOYEES
============================================================
                   1. View All Employees
                   2. View Employee by ID
------------------------------------------------------------
                  Enter your choice (1-2):
```

```
============================================================
                      VIEW EMPLOYEES
============================================================
                   1. View All Employees
                   2. View Employee by ID
------------------------------------------------------------
                 Enter your choice (1-2):                  1
ID: 1, Name: Rishabh Raj, Age: 23, Department: Software Engineer
   ID: 2, Name: Harish Kumar, Age: 25, Department: Sales
        ID: 3, Name: rahulk, Age: 25, Department: IT
 ID: 4, Name: Subrojit Roy, Age: 25, Department: Marketing
   ID: 5, Name: Steve Rogers, Age: 27, Department: Sales
```

```
============================================================
                      EMPLOYEE PORTAL
============================================================
                        1. Register
                          2. Login
------------------------------------------------------------
                 Enter your choice (1-2):
```

```
============================================================
                   EMPLOYEE REGISTRATION
============================================================
Enter Name: Kapil Yadav
Enter Age: 27
Enter Department: IT
Enter Username: kapyad
Enter Password:
```


EMPLOYEE MENU
1. Update Details
2. View Details
3. Logout
Enter your choice (1-3):


EMPLOYEE MENU
1. Update Details
2. View Details
3. Logout
Enter your choice (1-3): 2
Your Details:
ID: 7, Name: Kapil Yadav, Age: 27, Department: IT
Press Enter to return to the menu...