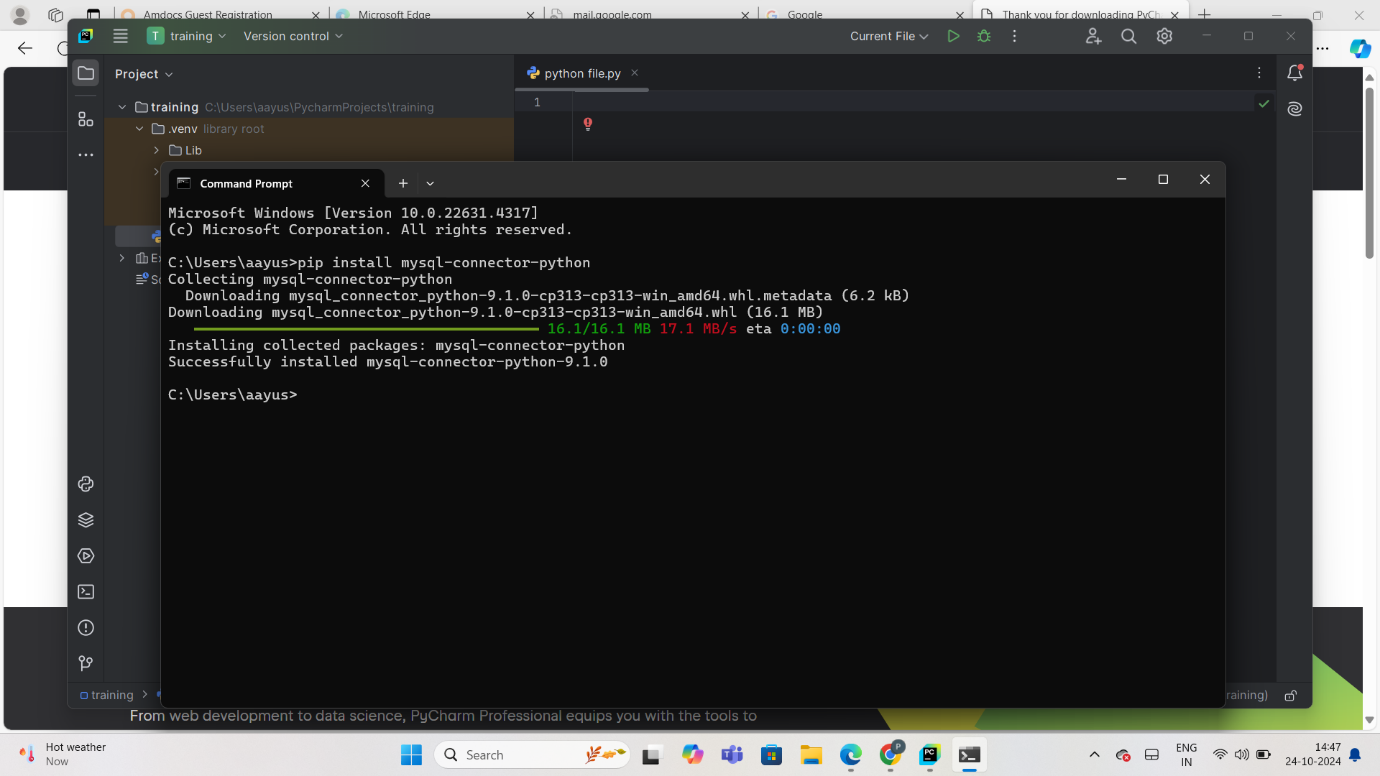
**Name: Aayushi Pandit**

**Department: SmartOps Fusion**

**EMPLOYEE MANAGEMENT SYSTEM**

**Command Prompt Code and Output:**

****

**MySQL Workbench Code:**

CREATE DATABASE EmpManage;

USE EmpManage;

CREATE TABLE Employees (

EmployeeID INT PRIMARY KEY,

FirstName VARCHAR(50) NOT NULL,

LastName VARCHAR(50) NOT NULL,

Department VARemployeesusersCHAR(50),

Position VARCHAR(50),

Salary DECIMAL(10, 2),

HireDate DATE

);

INSERT INTO Employees (EmployeeID, FirstName, LastName, Department, Position, Salary, HireDate)

VALUES

(1, 'John', 'Doe', 'IT', 'Software Engineer', 60000, '2022-01-15'),

(2, 'Jane', 'Smith', 'HR', 'HR Manager', 75000, '2020-03-01'),

(3, 'Bob', 'Johnson', 'Finance', 'Accountant', 55000, '2019-05-10'),

(4, 'Alice', 'Williams', 'IT', 'Data Analyst', 62000, '2021-07-20'),

(5, 'Michael', 'Brown', 'Marketing', 'Marketing Executive', 50000, '2018-11-05'),

(6, 'Emily', 'Davis', 'IT', 'Project Manager', 85000, '2017-09-17'),

(7, 'David', 'Wilson', 'Finance', 'Financial Analyst', 58000, '2021-04-12'),

(8, 'Sarah', 'Miller', 'Sales', 'Sales Manager', 72000, '2016-12-09'),

(9, 'Robert', 'Moore', 'Sales', 'Sales Associate', 46000, '2023-02-28'),

(10, 'Jessica', 'Taylor', 'HR', 'HR Associate', 48000, '2022-10-22');

CREATE TABLE users (

UserID INT AUTO\_INCREMENT PRIMARY KEY,

Username VARCHAR(50) UNIQUE NOT NULL,

UserPassword VARCHAR(255) NOT NULL

);

INSERT INTO users (Username, UserPassword)

VALUES ('root', 'root');

ALTER TABLE users

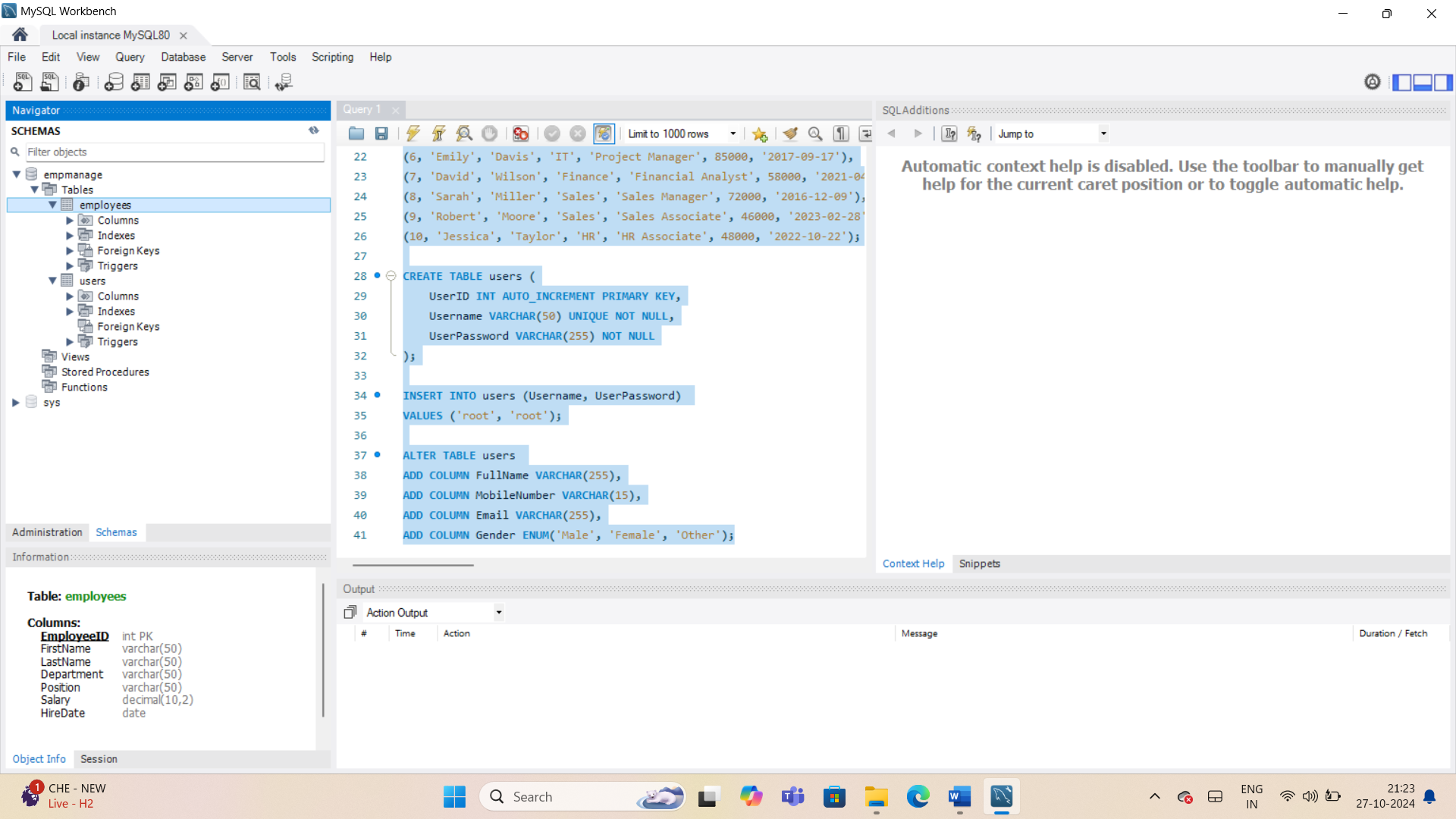
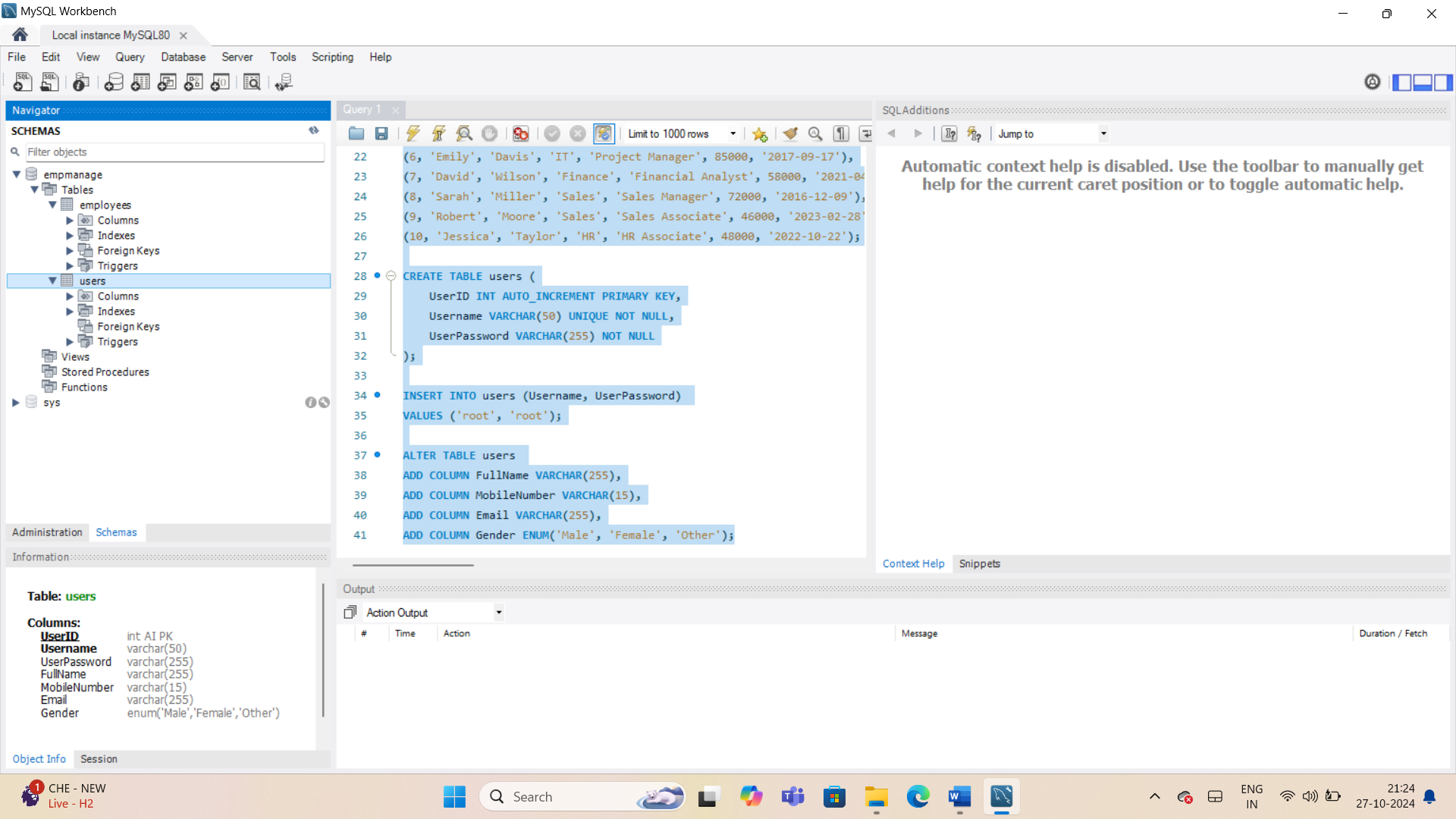
ADD COLUMN FullName VARCHAR(255),

ADD COLUMN MobileNumber VARCHAR(15),

ADD COLUMN Email VARCHAR(255),

ADD COLUMN Gender ENUM('Male', 'Female', 'Other');

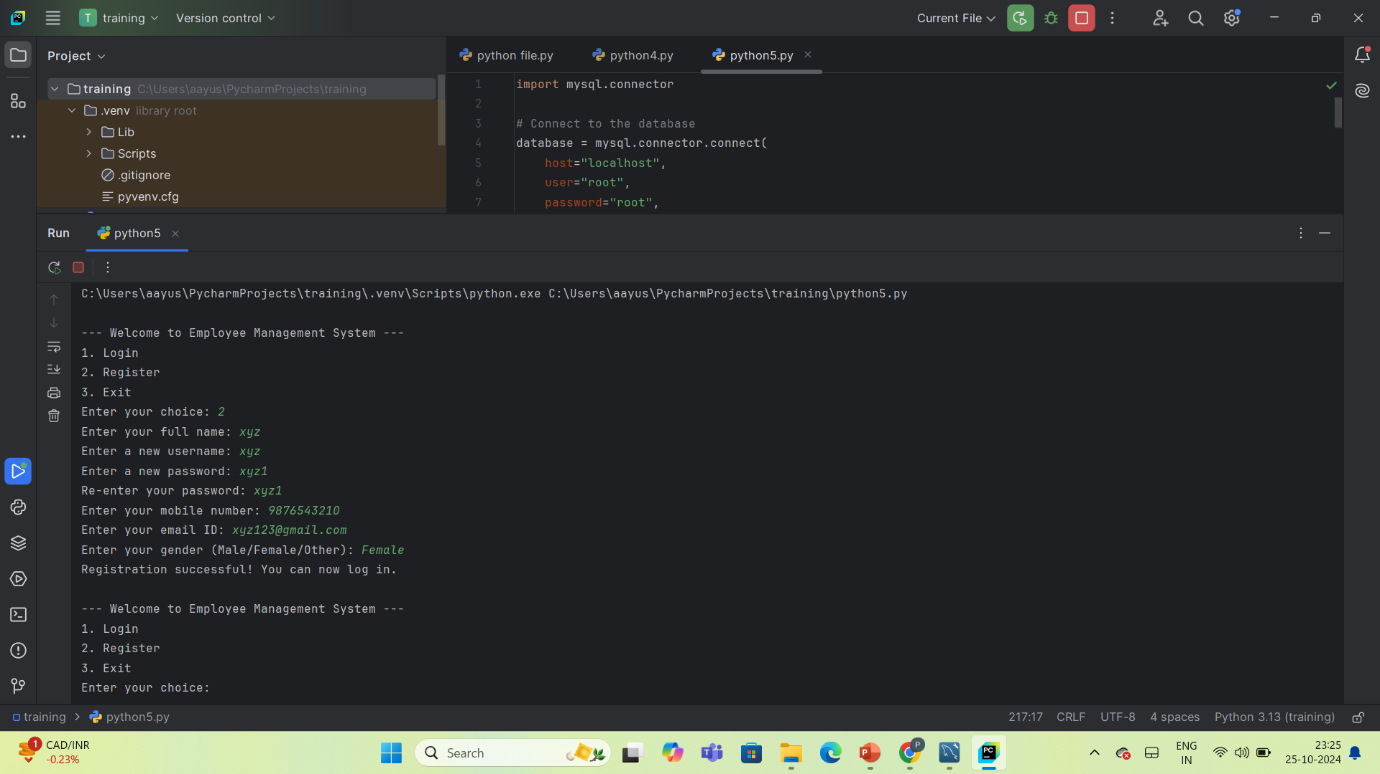
**MYSQL Workbench Output:**

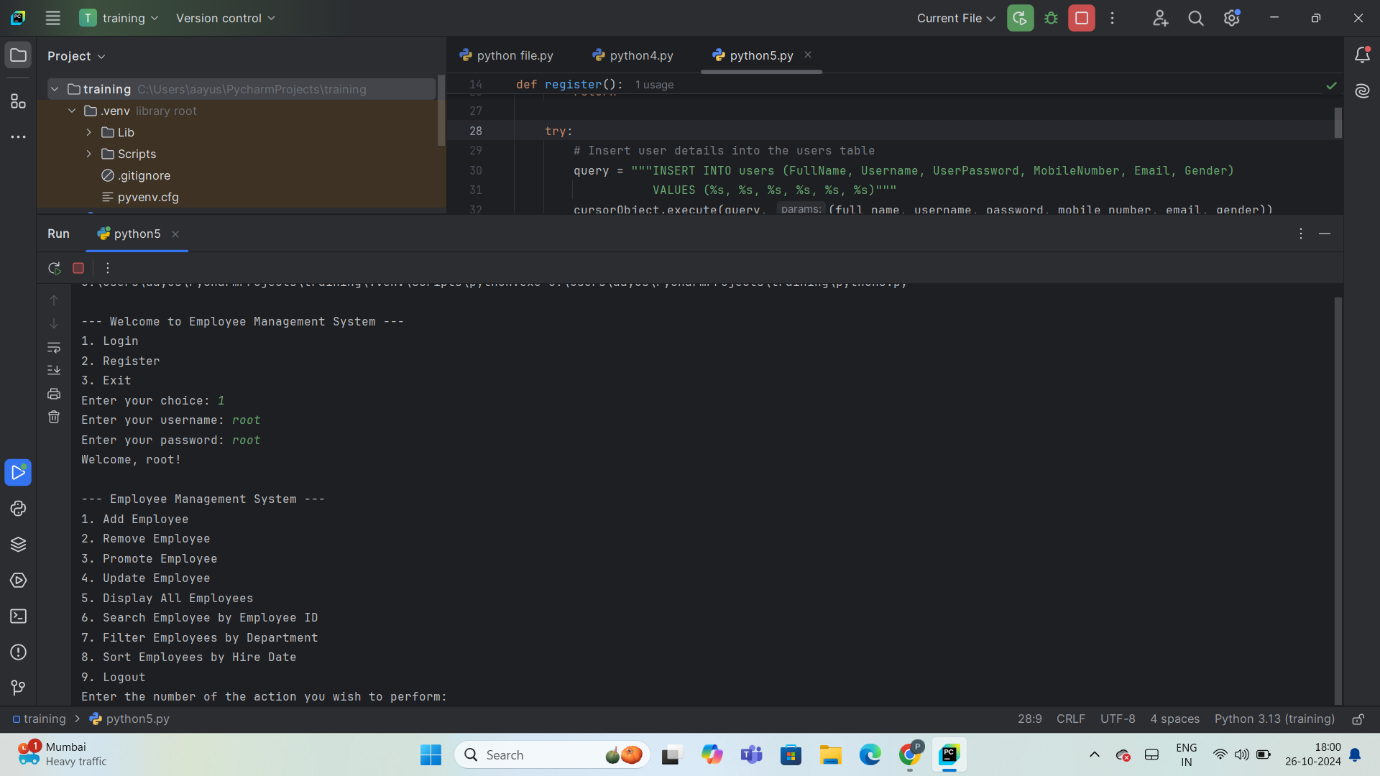
 

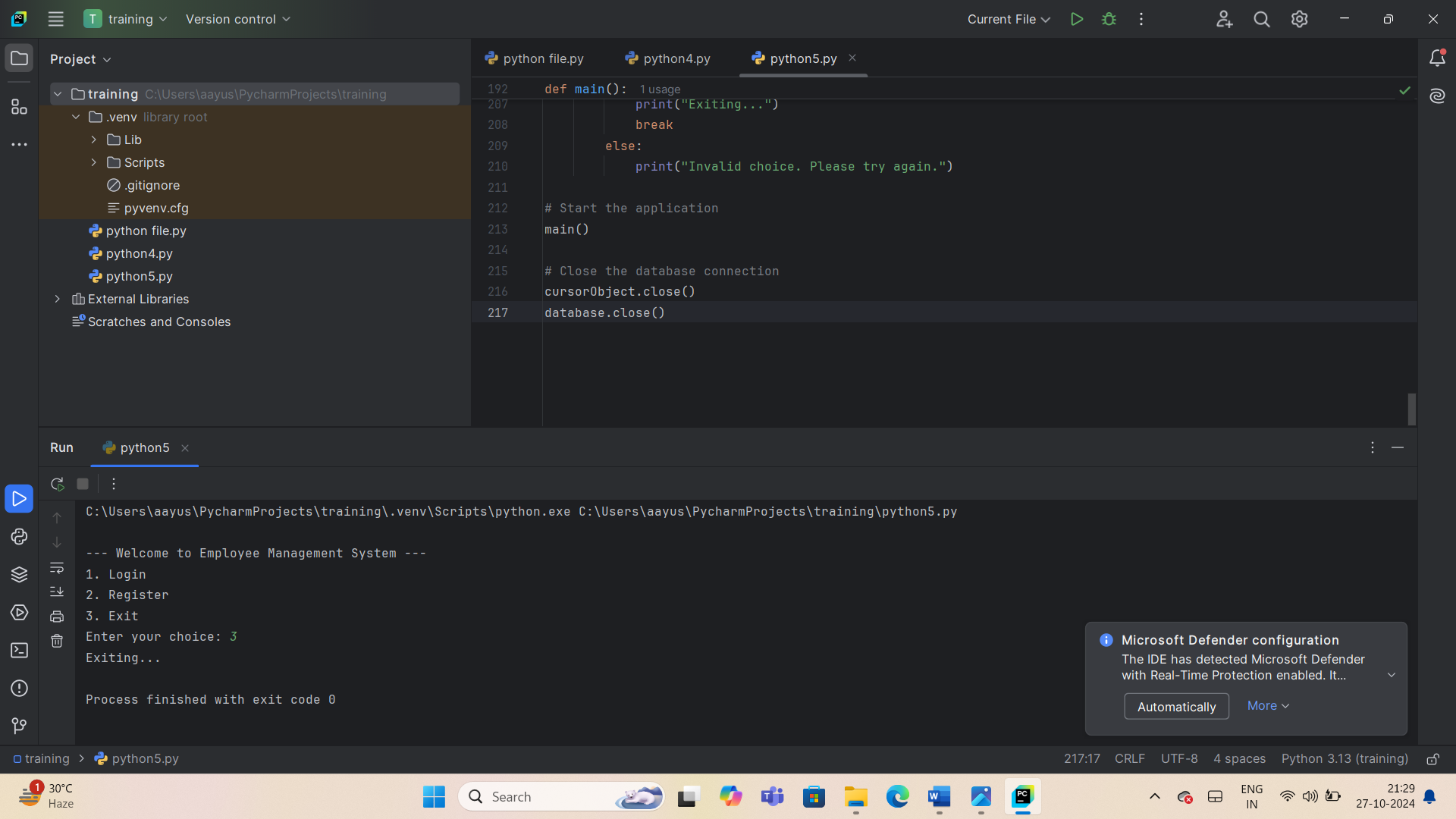
**PyCharm Code:**

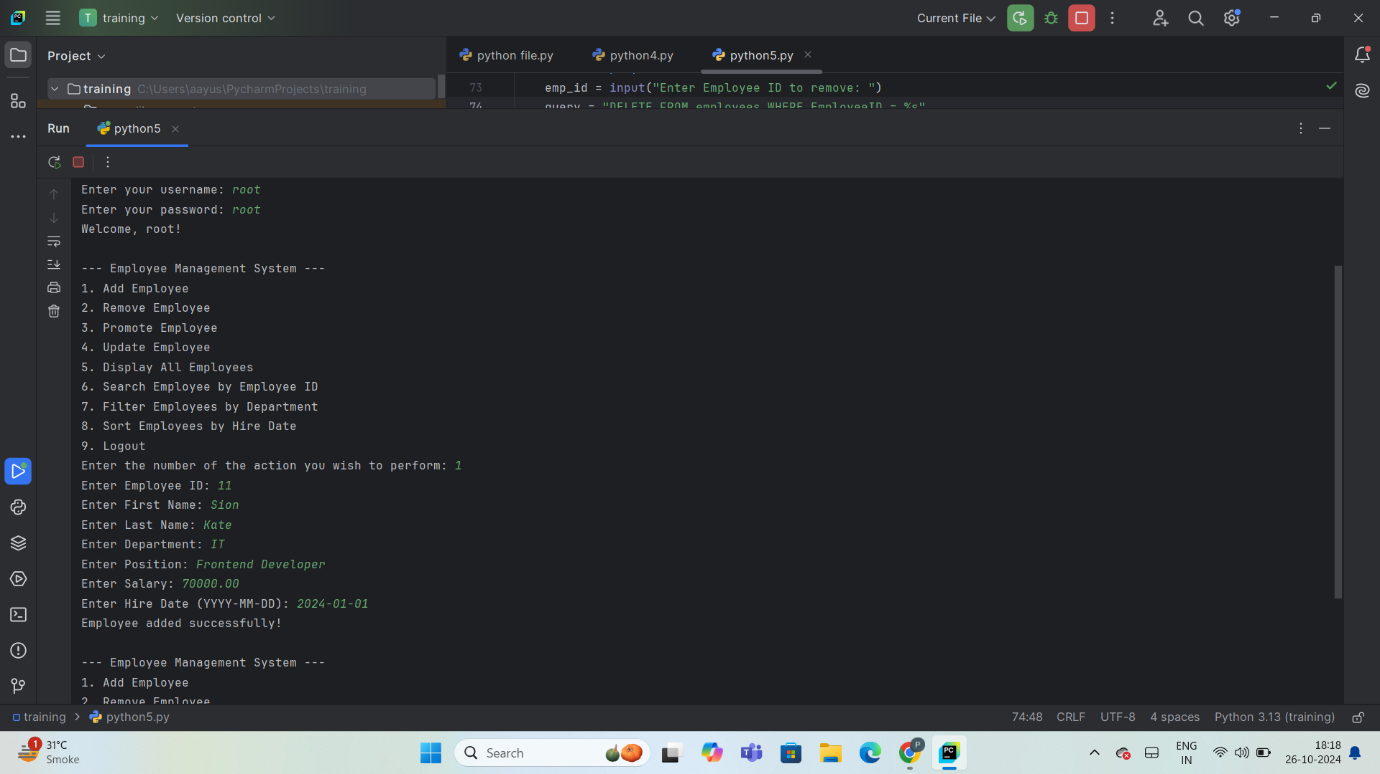
import mysql.connector  
  
# Connect to the database  
database = mysql.connector.connect(  
 host="localhost",  
 user="root",  
 password="root",  
 database="EmpManage"  
)  
  
cursorObject = database.cursor()  
  
# Function to register a new user  
def register():  
 full\_name = input("Enter your full name: ")  
 username = input("Enter a new username: ")  
 password = input("Enter a new password: ")  
 re\_password = input("Re-enter your password: ")  
 mobile\_number = input("Enter your mobile number: ")  
 email = input("Enter your email ID: ")  
 gender = input("Enter your gender (Male/Female/Other): ")  
  
 # Check if passwords match  
 if password != re\_password:  
 print("Passwords do not match. Please try again.")  
 return  
  
 try:  
 # Insert user details into the users table  
 query = """INSERT INTO users (FullName, Username, UserPassword, MobileNumber, Email, Gender)   
 VALUES (%s, %s, %s, %s, %s, %s)"""  
 cursorObject.execute(query, (full\_name, username, password, mobile\_number, email, gender))  
 database.commit()  
 print("Registration successful! You can now log in.")  
 except mysql.connector.Error as err:  
 print(f"Error: {err}")  
 print("Registration failed. Try a different username.")  
  
# Function to log in an existing user  
def login():  
 username = input("Enter your username: ")  
 password = input("Enter your password: ")  
  
 query = "SELECT \* FROM users WHERE Username = %s AND UserPassword = %s"  
 cursorObject.execute(query, (username, password))  
 user = cursorObject.fetchone()  
  
 if user:  
 print(f"Welcome, {username}!")  
 return True  
 else:  
 print("Invalid credentials. Please try again.")  
 return False  
  
# Employee Management Functions  
def add\_employee():  
 emp\_id = input("Enter Employee ID: ")  
 first\_name = input("Enter First Name: ")  
 last\_name = input("Enter Last Name: ")  
 department = input("Enter Department: ")  
 position = input("Enter Position: ")  
 salary = float(input("Enter Salary: "))  
 hire\_date = input("Enter Hire Date (YYYY-MM-DD): ")  
  
 query = """INSERT INTO employees (EmployeeID, FirstName, LastName, Department, Position, Salary, HireDate)   
 VALUES (%s, %s, %s, %s, %s, %s, %s)"""  
 values = (emp\_id, first\_name, last\_name, department, position, salary, hire\_date)  
 cursorObject.execute(query, values)  
 database.commit()  
 print("Employee added successfully!")  
  
def remove\_employee():  
 emp\_id = input("Enter Employee ID to remove: ")  
 query = "DELETE FROM employees WHERE EmployeeID = %s"  
 cursorObject.execute(query, (emp\_id,))  
 database.commit()  
 print("Employee removed successfully!")  
  
def promote\_employee():  
 emp\_id = input("Enter Employee ID to promote: ")  
 new\_position = input("Enter new Position: ")  
 new\_salary = float(input("Enter new Salary: "))  
 query = "UPDATE employees SET Position = %s, Salary = %s WHERE EmployeeID = %s"  
 cursorObject.execute(query, (new\_position, new\_salary, emp\_id))  
 database.commit()  
 print("Employee promoted successfully!")  
  
def update\_employee():  
 emp\_id = input("Enter Employee ID to update: ")  
 first\_name = input("Enter new First Name: ")  
 last\_name = input("Enter new Last Name: ")  
 department = input("Enter new Department: ")  
 position = input("Enter new Position: ")  
  
 # Fetch the current salary from the database  
 query = "SELECT Salary FROM employees WHERE EmployeeID = %s"  
 cursorObject.execute(query, (emp\_id,))  
 current\_salary = cursorObject.fetchone()  
  
 if current\_salary:  
 current\_salary\_float = float(current\_salary[0]) # Convert Decimal to float  
 print(f"Current Salary: {current\_salary\_float}")  
 percentage\_increment = float(input("Enter the percentage increment for Salary: "))  
 new\_salary = current\_salary\_float + (current\_salary\_float \* percentage\_increment / 100)  
 print(f"New Salary after {percentage\_increment}% increment: {new\_salary}")  
  
 # Proceed with the update  
 query = """UPDATE employees SET FirstName = %s, LastName = %s, Department = %s,   
 Position = %s, Salary = %s WHERE EmployeeID = %s"""  
 cursorObject.execute(query, (first\_name, last\_name, department, position, new\_salary, emp\_id))  
 database.commit()  
 print("Employee updated successfully!")  
 else:  
 print("Employee not found!")  
  
def display\_all\_employees():  
 query = "SELECT \* FROM employees"  
 cursorObject.execute(query)  
 employees = cursorObject.fetchall()  
 total\_employees = len(employees)  
  
 print("\n--- Employee List ---")  
 for employee in employees:  
 print(employee)  
  
 print(f"\nTotal Number of Employees: {total\_employees}")  
  
def search\_employee\_by\_id():  
 emp\_id = input("Enter Employee ID to search: ")  
 query = "SELECT \* FROM employees WHERE EmployeeID = %s"  
 cursorObject.execute(query, (emp\_id,))  
 employee = cursorObject.fetchone()  
 if employee:  
 print(employee)  
 else:  
 print("Employee not found!")  
  
def filter\_employees\_by\_department():  
 department = input("Enter department to filter by: ")  
 query = "SELECT \* FROM employees WHERE Department = %s"  
 cursorObject.execute(query, (department,))  
 employees = cursorObject.fetchall()  
 for employee in employees:  
 print(employee)  
  
def sort\_employees\_by\_joining\_date():  
 query = "SELECT \* FROM employees ORDER BY HireDate"  
 cursorObject.execute(query)  
 employees = cursorObject.fetchall()  
 for employee in employees:  
 print(employee)  
  
# Employee Management System Menu  
def employee\_management\_system():  
 while True:  
 print("\n--- Employee Management System ---")  
 print("1. Add Employee")  
 print("2. Remove Employee")  
 print("3. Promote Employee")  
 print("4. Update Employee")  
 print("5. Display All Employees")  
 print("6. Search Employee by Employee ID")  
 print("7. Filter Employees by Department")  
 print("8. Sort Employees by Hire Date")  
 print("9. Logout")  
  
 choice = input("Enter the number of the action you wish to perform: ")  
  
 if choice == '1':  
 add\_employee()  
 elif choice == '2':  
 remove\_employee()  
 elif choice == '3':  
 promote\_employee()  
 elif choice == '4':  
 update\_employee()  
 elif choice == '5':  
 display\_all\_employees()  
 elif choice == '6':  
 search\_employee\_by\_id()  
 elif choice == '7':  
 filter\_employees\_by\_department()  
 elif choice == '8':  
 sort\_employees\_by\_joining\_date()  
 elif choice == '9':  
 print("Logging out...")  
 break  
 else:  
 print("Invalid choice. Please try again.")  
  
# Main Function: Login or Register  
def main():  
 while True:  
 print("\n--- Welcome to Employee Management System ---")  
 print("1. Login")  
 print("2. Register")  
 print("3. Exit")  
  
 choice = input("Enter your choice: ")  
  
 if choice == '1':  
 if login():  
 employee\_management\_system()  
 elif choice == '2':  
 register()  
 elif choice == '3':  
 print("Exiting...")  
 break  
 else:  
 print("Invalid choice. Please try again.")  
  
# Start the application  
main()  
  
# Close the database connection  
cursorObject.close()  
database.close()

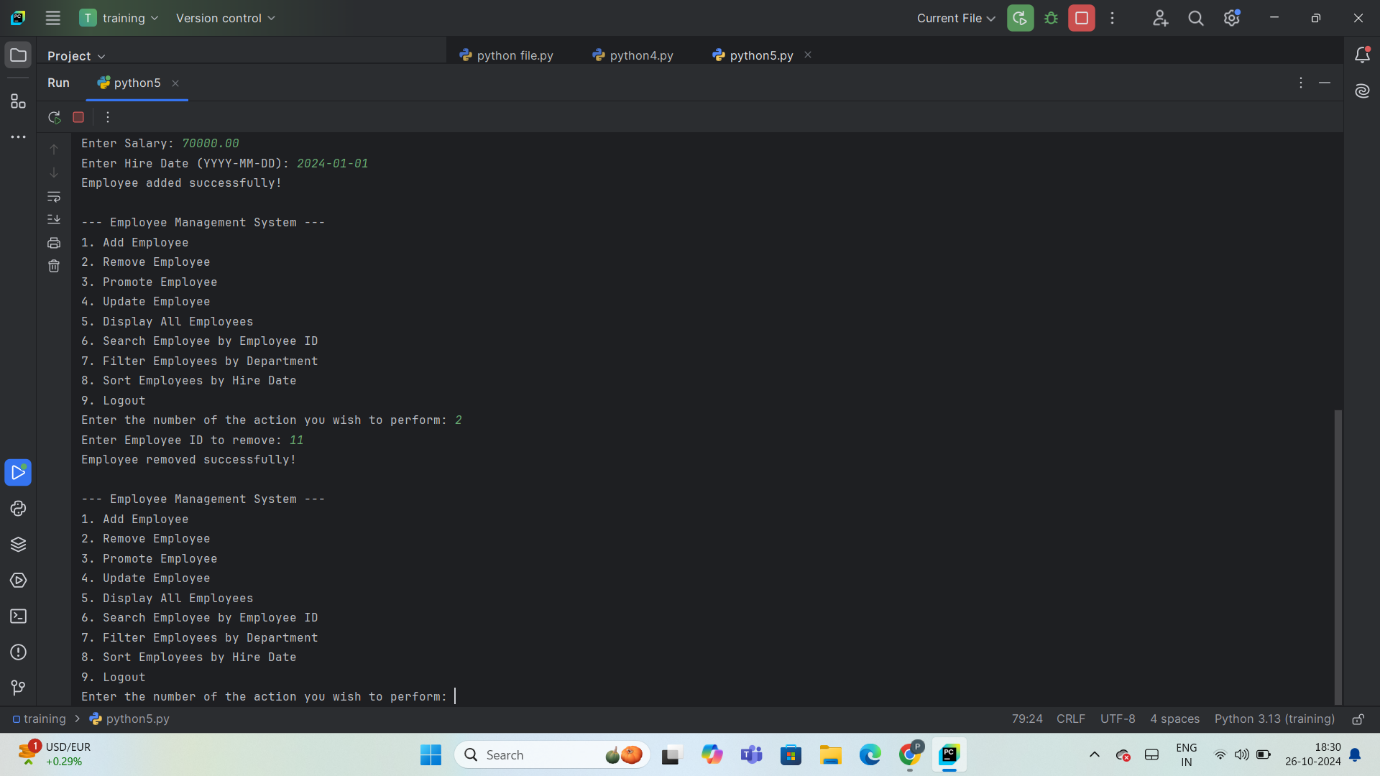
**PyCharm Output:**

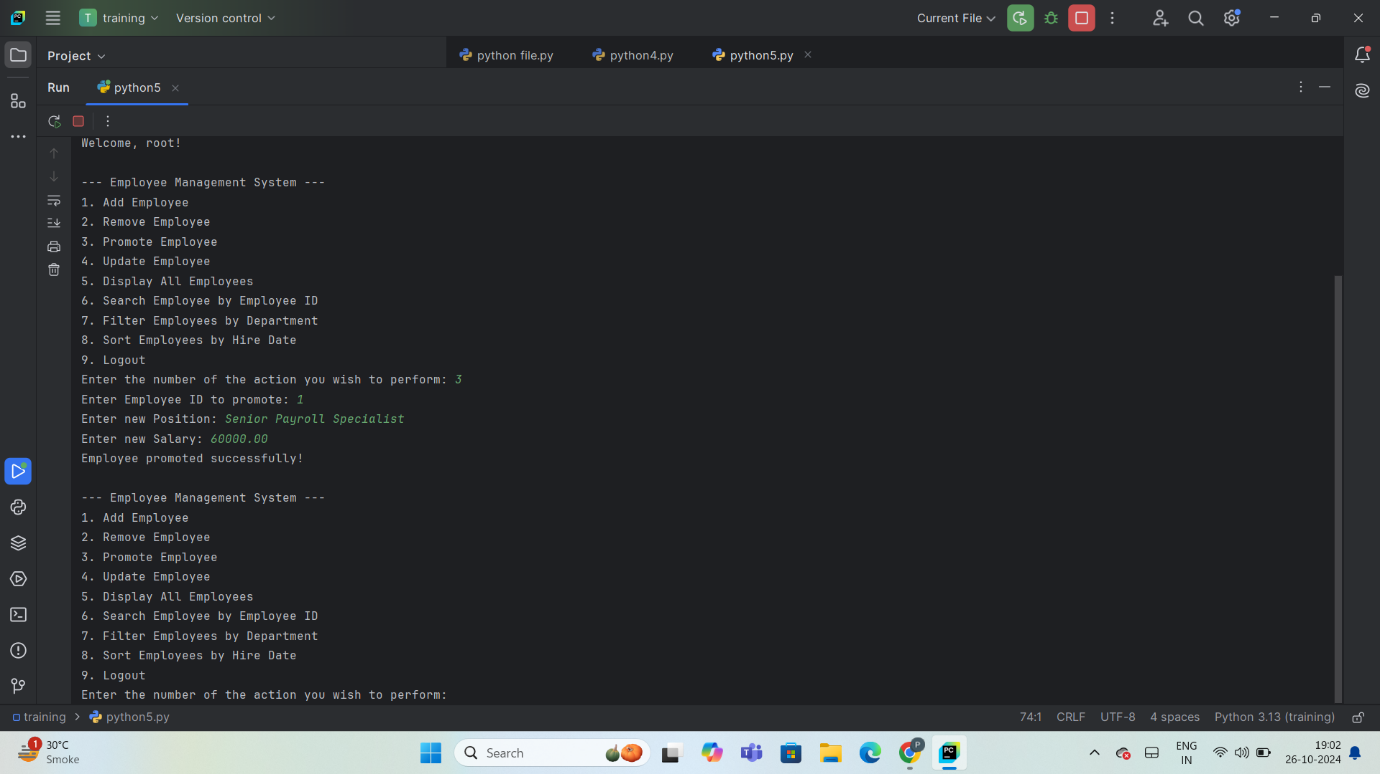
****

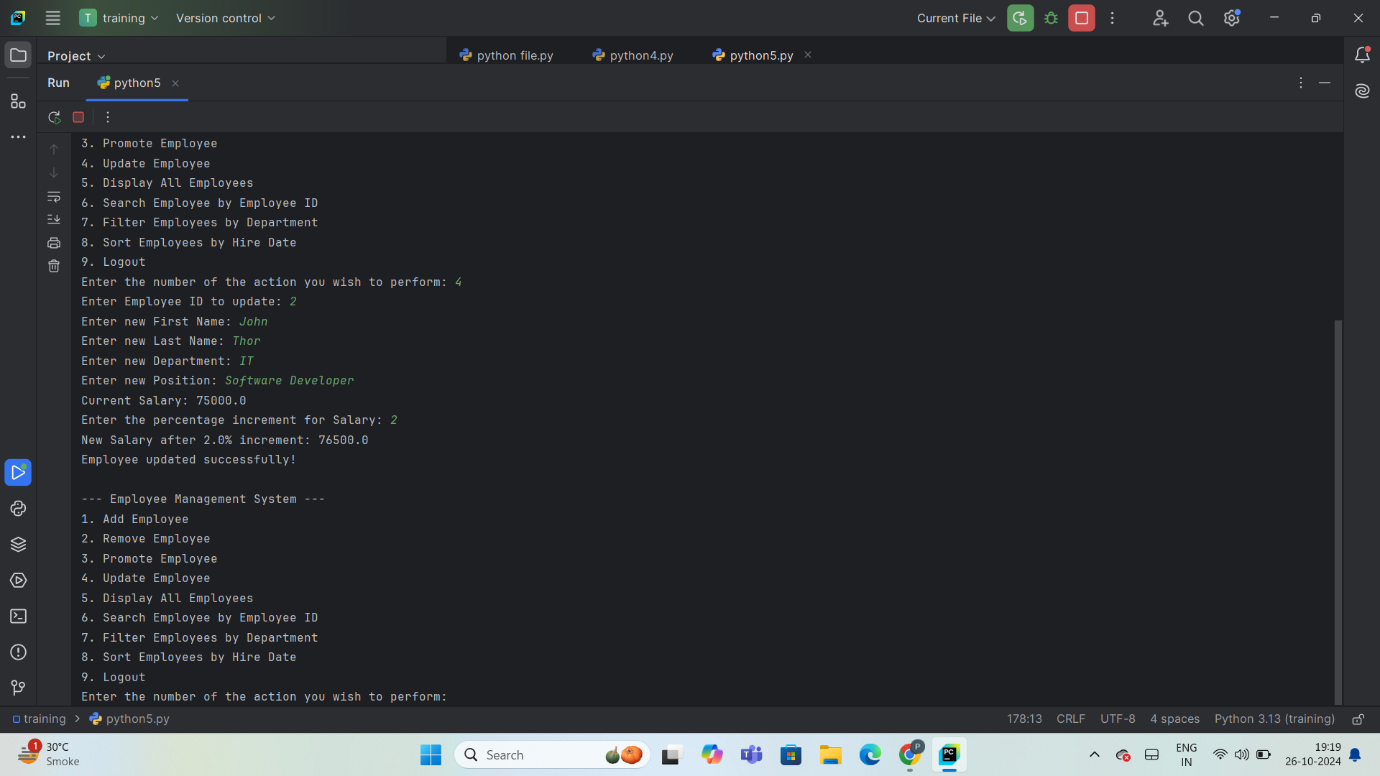
****

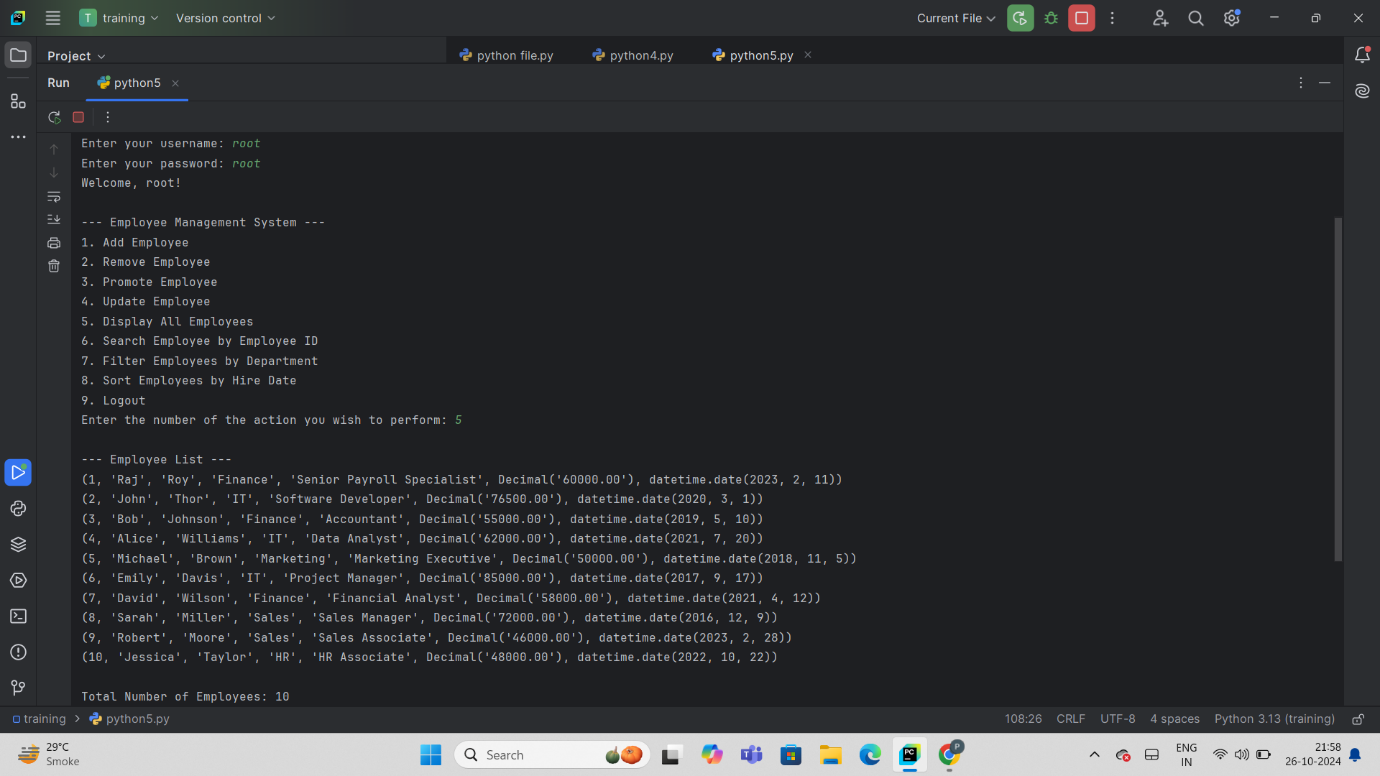


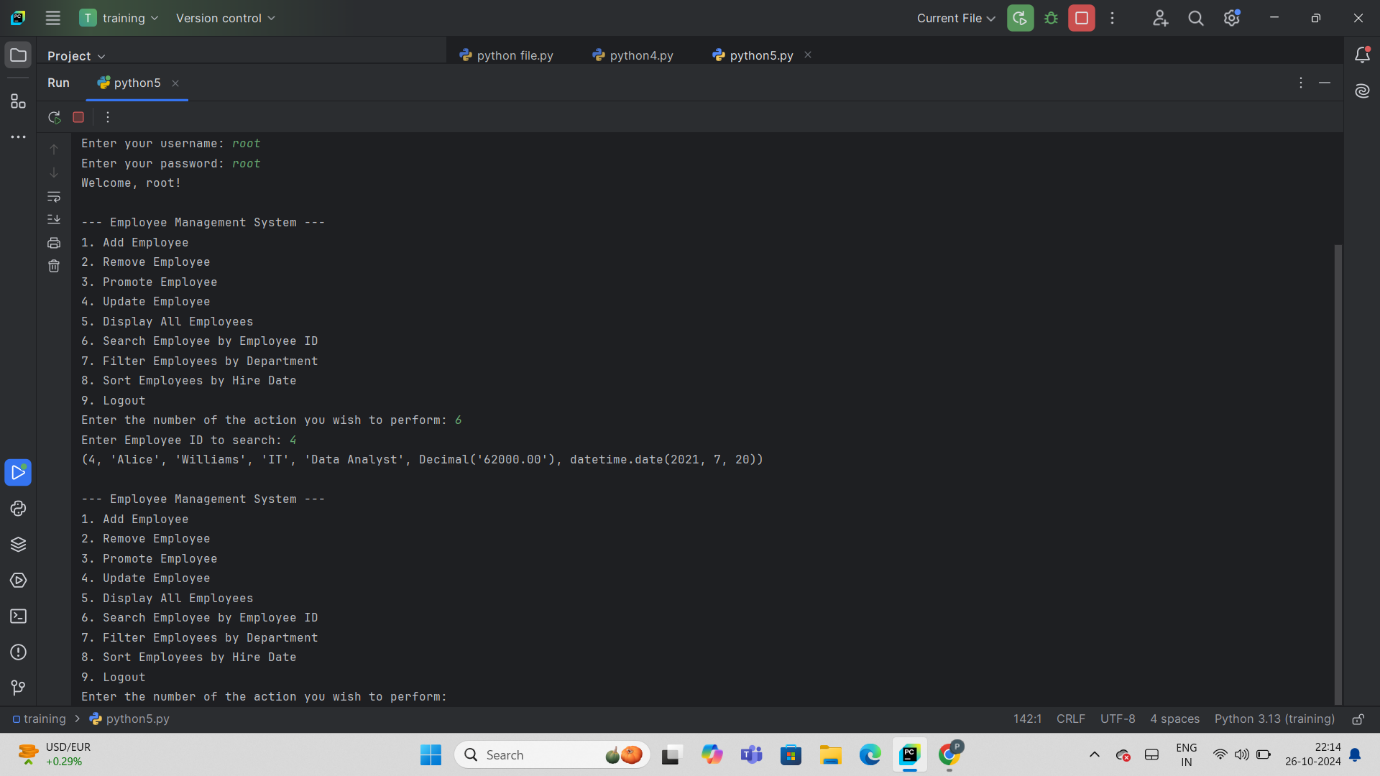
****

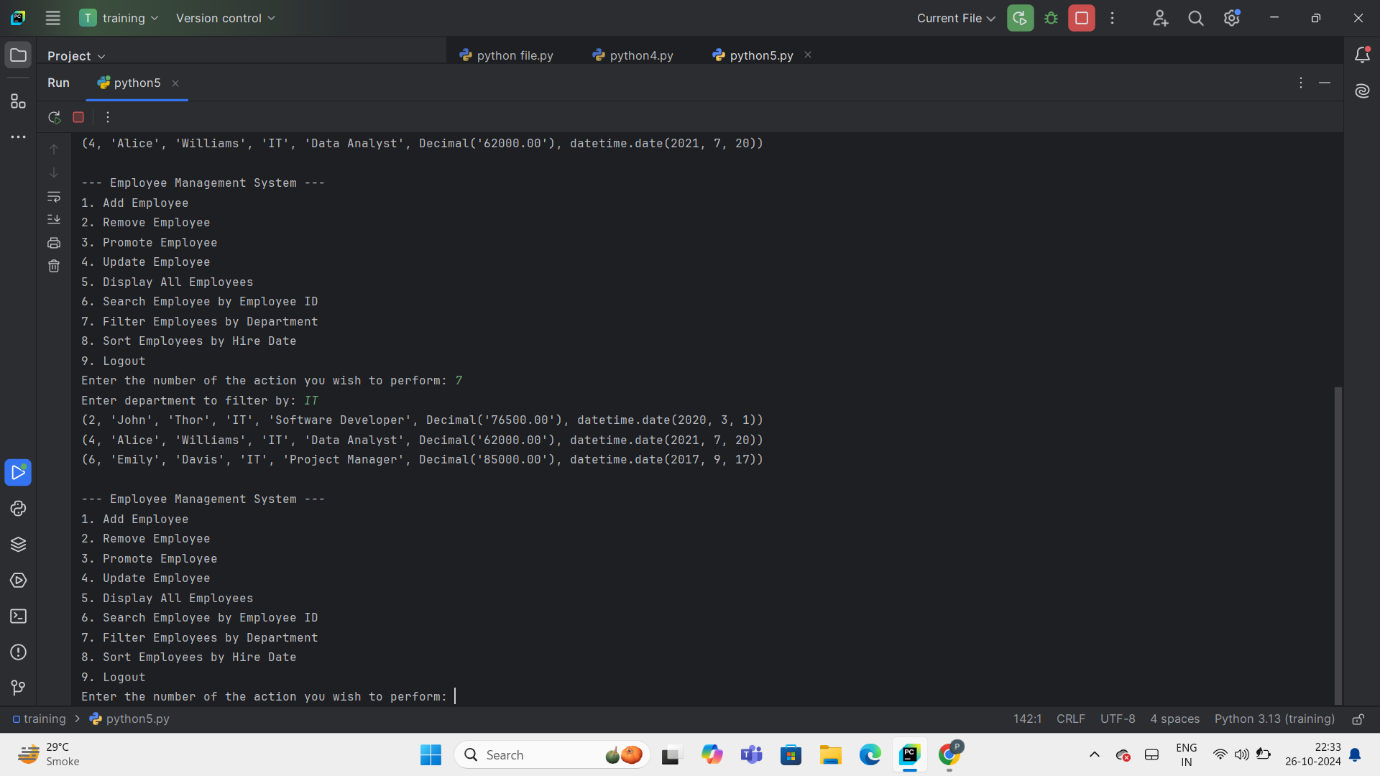
****

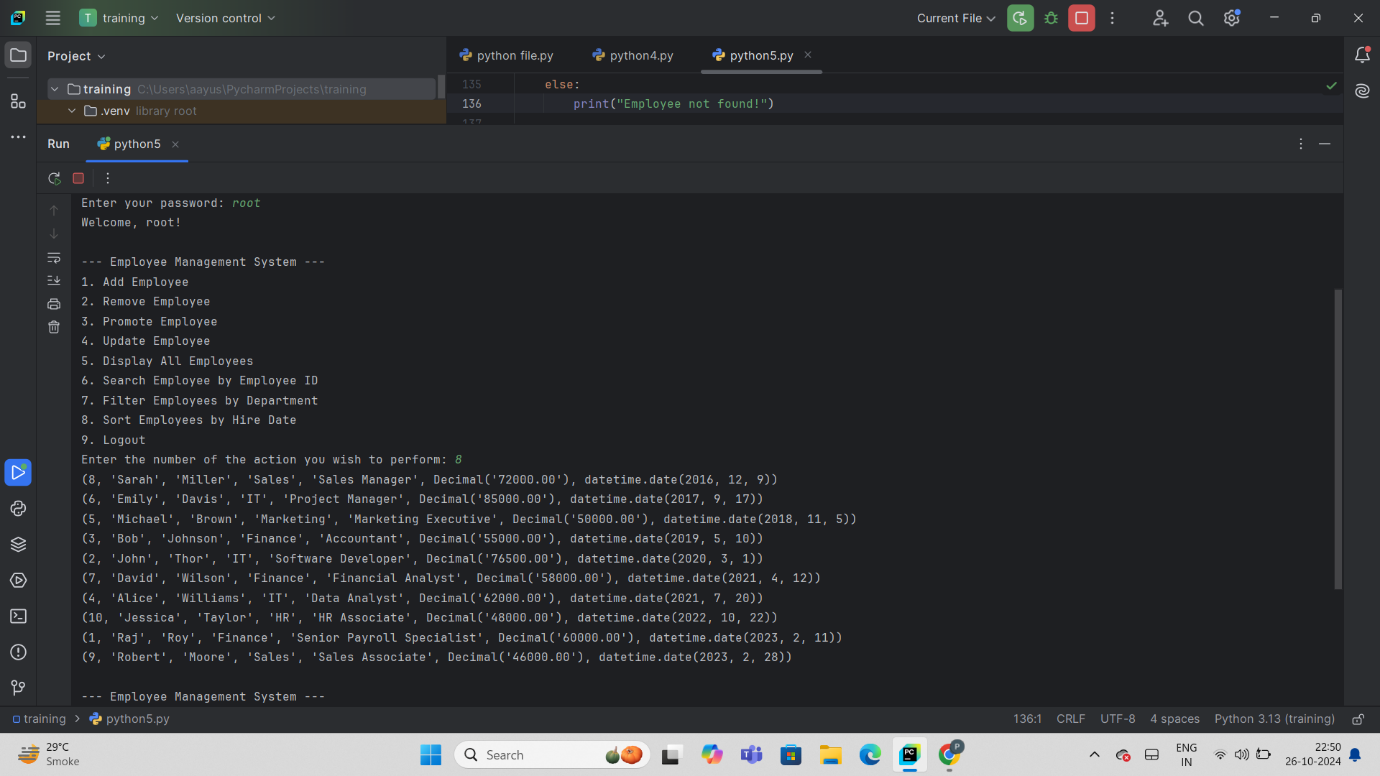
****

****

****

****

****

****

