Payroll Management System

-Shrutika Mahajan

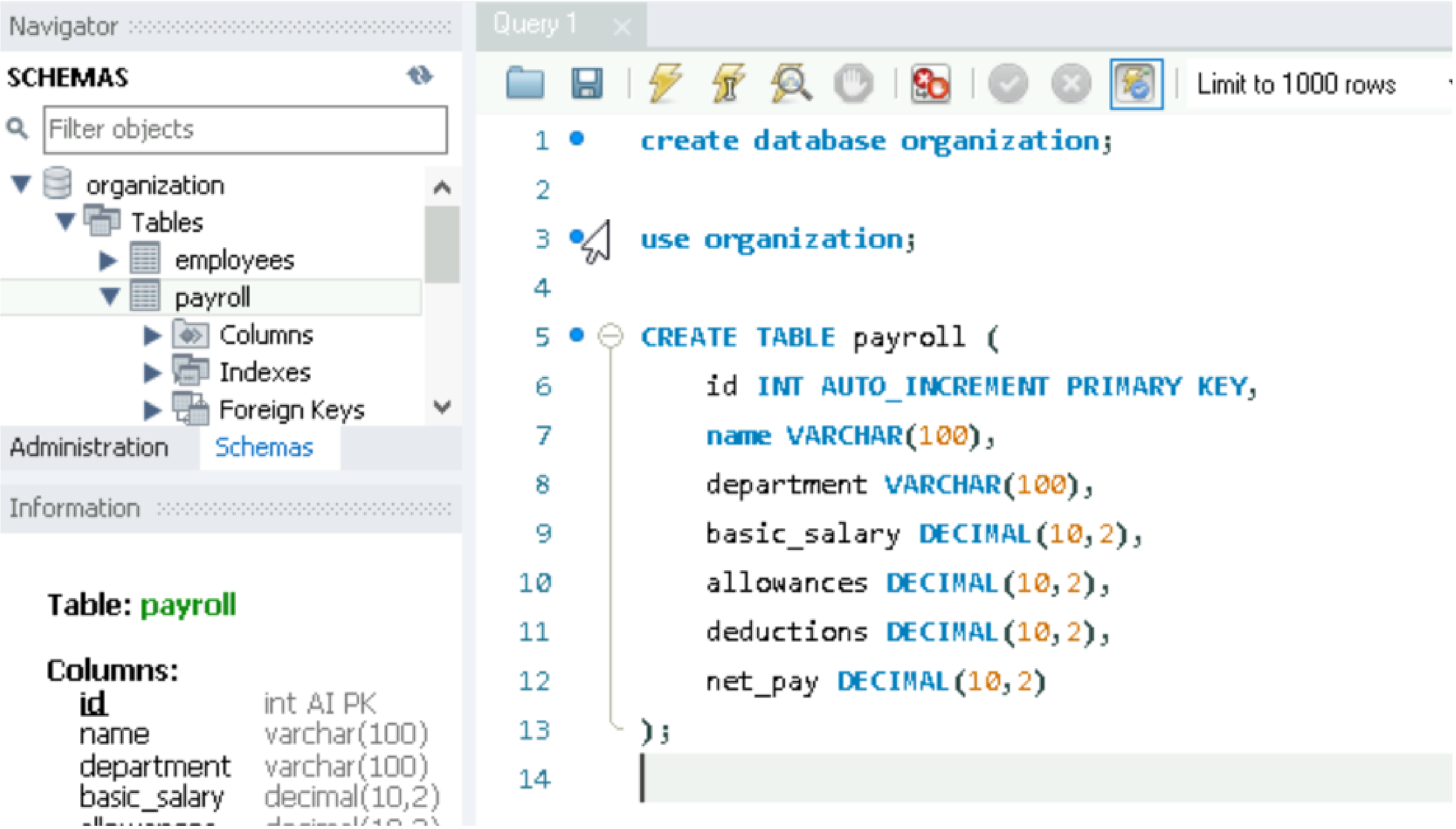
Overview:

Managing payroll manually can be time-consuming and error-prone. This project aims to simplify payroll operations using a Python-based command-line application connected to a MySQLdatabase

Goal:

To automate salary calculations, streamline record management, and provide flexible reporting features.

Database for storing payroll records:



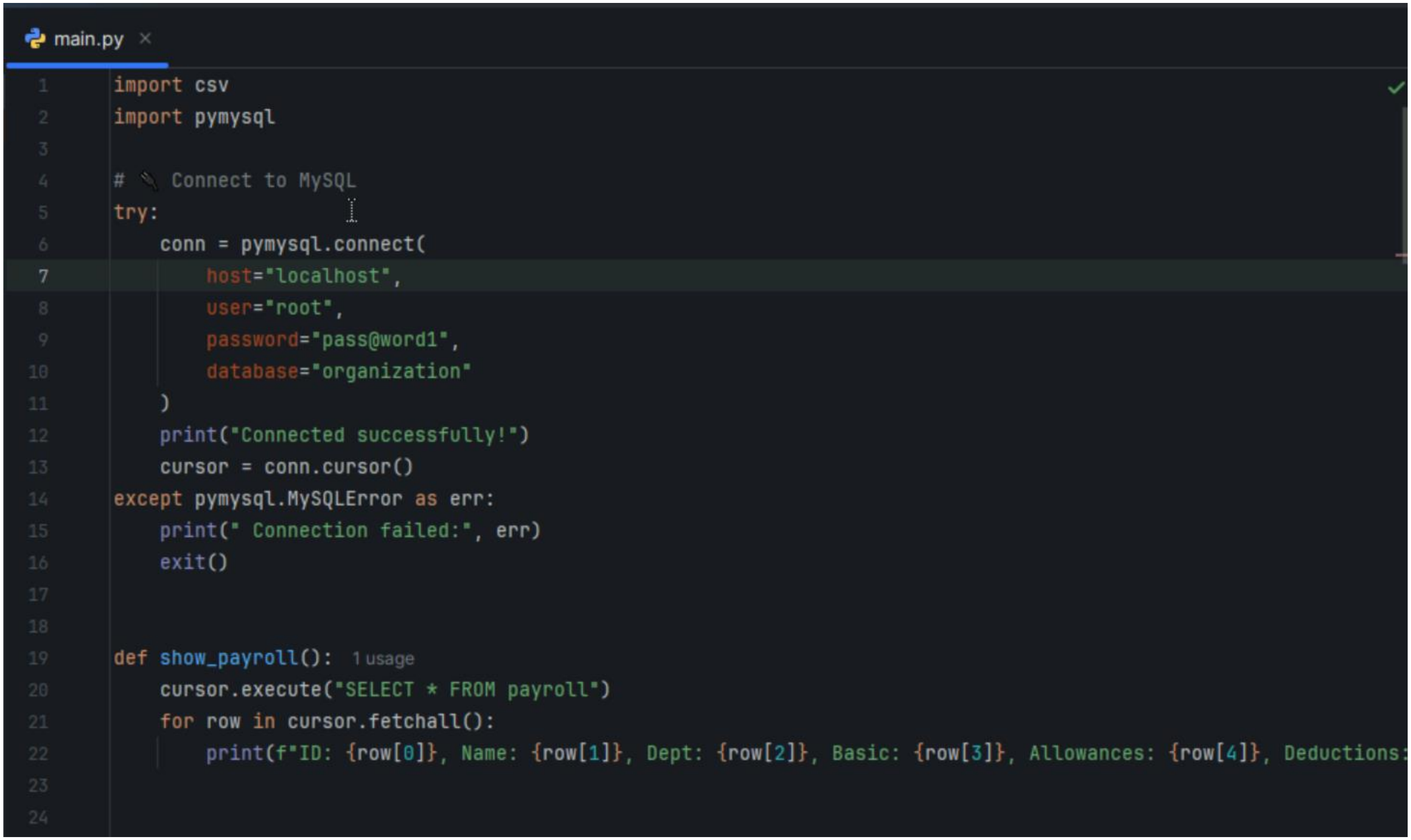
```sql
create database organization;

use organization;

CREATE TABLE payroll (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100),
    department VARCHAR(100),
    basic_salary DECIMAL(10,2),
    allowances DECIMAL(10,2),
    deductions DECIMAL(10,2),
    net_pay DECIMAL(10,2)
);
```

Table: payroll

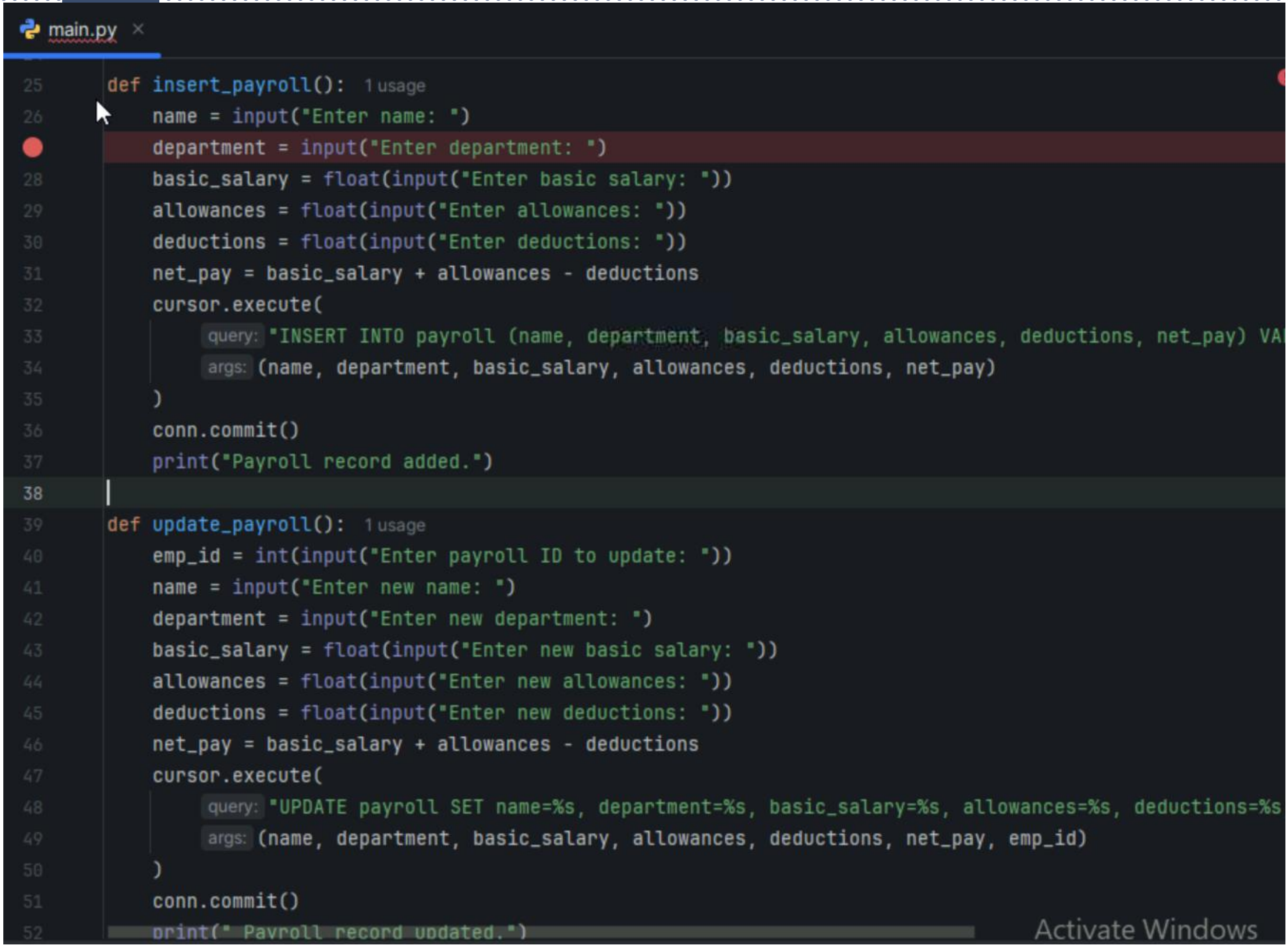
Columns:

id	int AI PK
name	varchar(100)
department	varchar(100)
basic_salary	decimal(10,2)

Code:

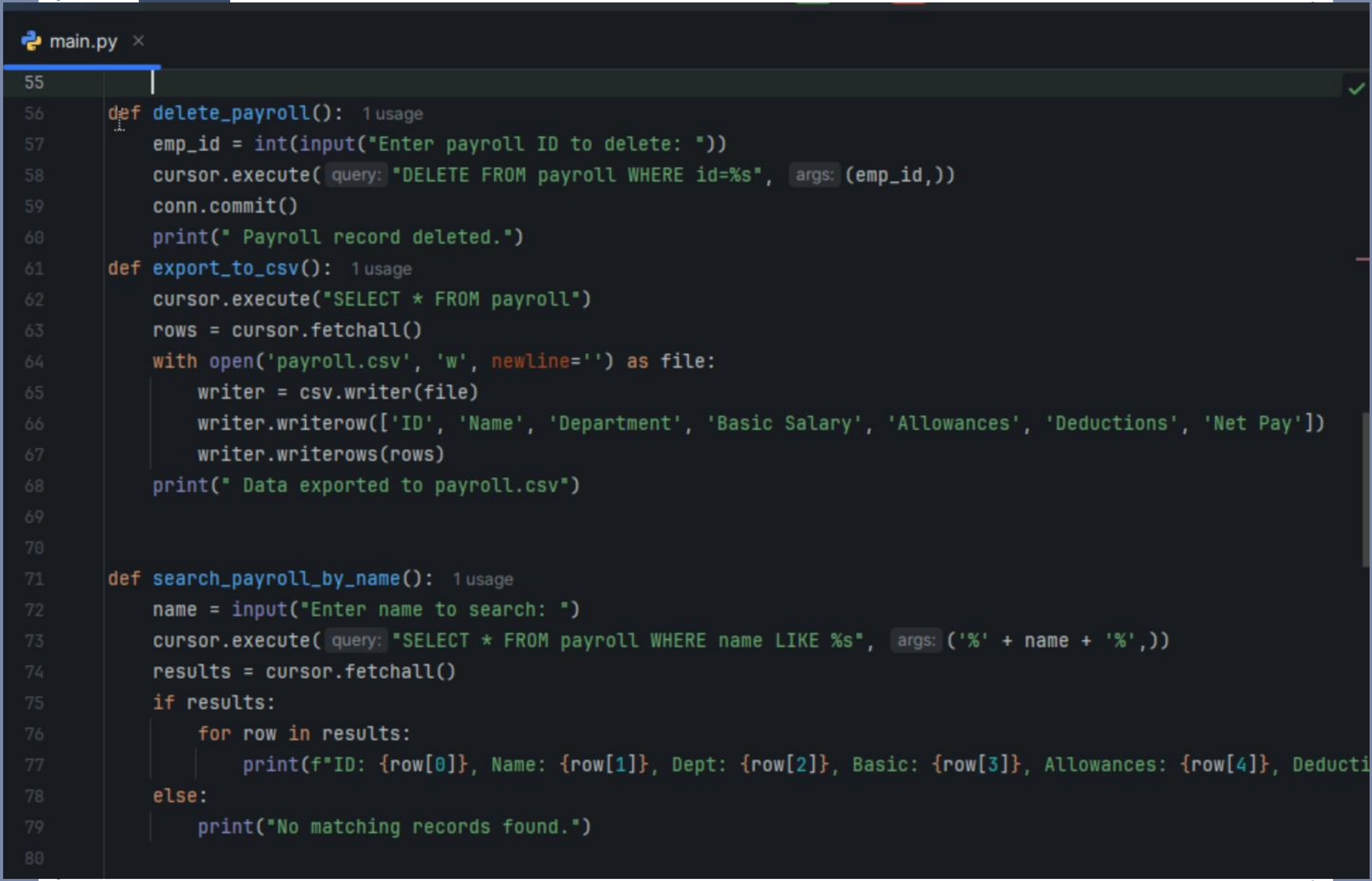


```python
import csv
import pymysql

# 🔌 Connect to MySQL
try:
    conn = pymysql.connect(
        host="localhost",
        user="root",
        password="pass@word1",
        database="organization"
    )
    print("Connected successfully!")
    cursor = conn.cursor()
except pymysql.MySQLError as err:
    print(" Connection failed:", err)
    exit()


def show_payroll():
    cursor.execute("SELECT * FROM payroll")
    for row in cursor.fetchall():
        print(f"ID: {row[0]}, Name: {row[1]}, Dept: {row[2]}, Basic: {row[3]}, Allowances: {row[4]}, Deductions:
```

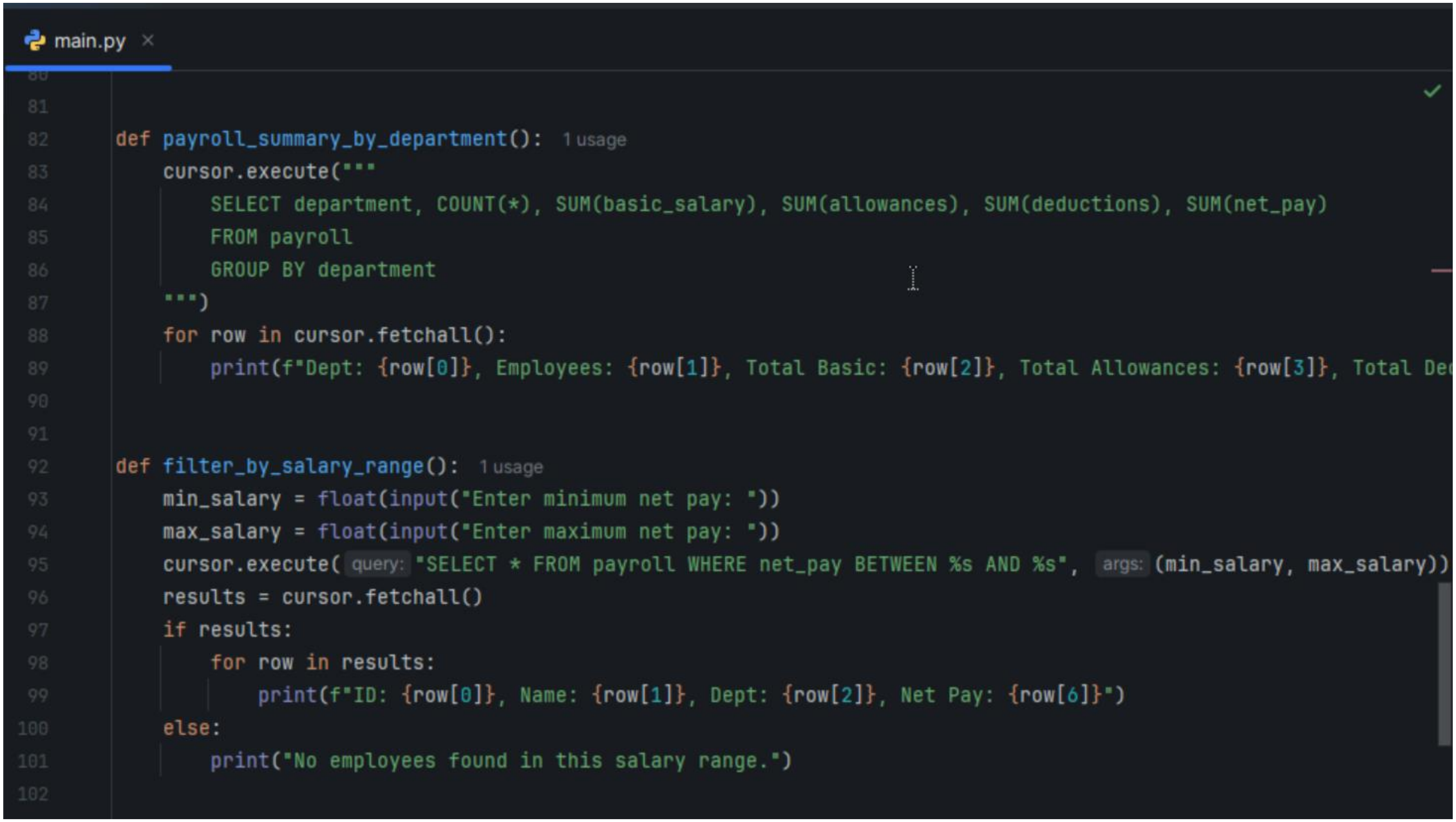


```
def insert_payroll():  1 usage
    name = input("Enter name: ")
    department = input("Enter department: ")
    basic_salary = float(input("Enter basic salary: "))
    allowances = float(input("Enter allowances: "))
    deductions = float(input("Enter deductions: "))
    net_pay = basic_salary + allowances - deductions
    cursor.execute(
        query: "INSERT INTO payroll (name, department, basic_salary, allowances, deductions, net_pay) VA
        args: (name, department, basic_salary, allowances, deductions, net_pay)
    )
    conn.commit()
    print("Payroll record added.")

def update_payroll():  1 usage
    emp_id = int(input("Enter payroll ID to update: "))
    name = input("Enter new name: ")
    department = input("Enter new department: ")
    basic_salary = float(input("Enter new basic salary: "))
    allowances = float(input("Enter new allowances: "))
    deductions = float(input("Enter new deductions: "))
    net_pay = basic_salary + allowances - deductions
    cursor.execute(
        query: "UPDATE payroll SET name=%s, department=%s, basic_salary=%s, allowances=%s, deductions=%s
        args: (name, department, basic_salary, allowances, deductions, net_pay, emp_id)
    )
    conn.commit()
    print(" Payroll record updated.")
```

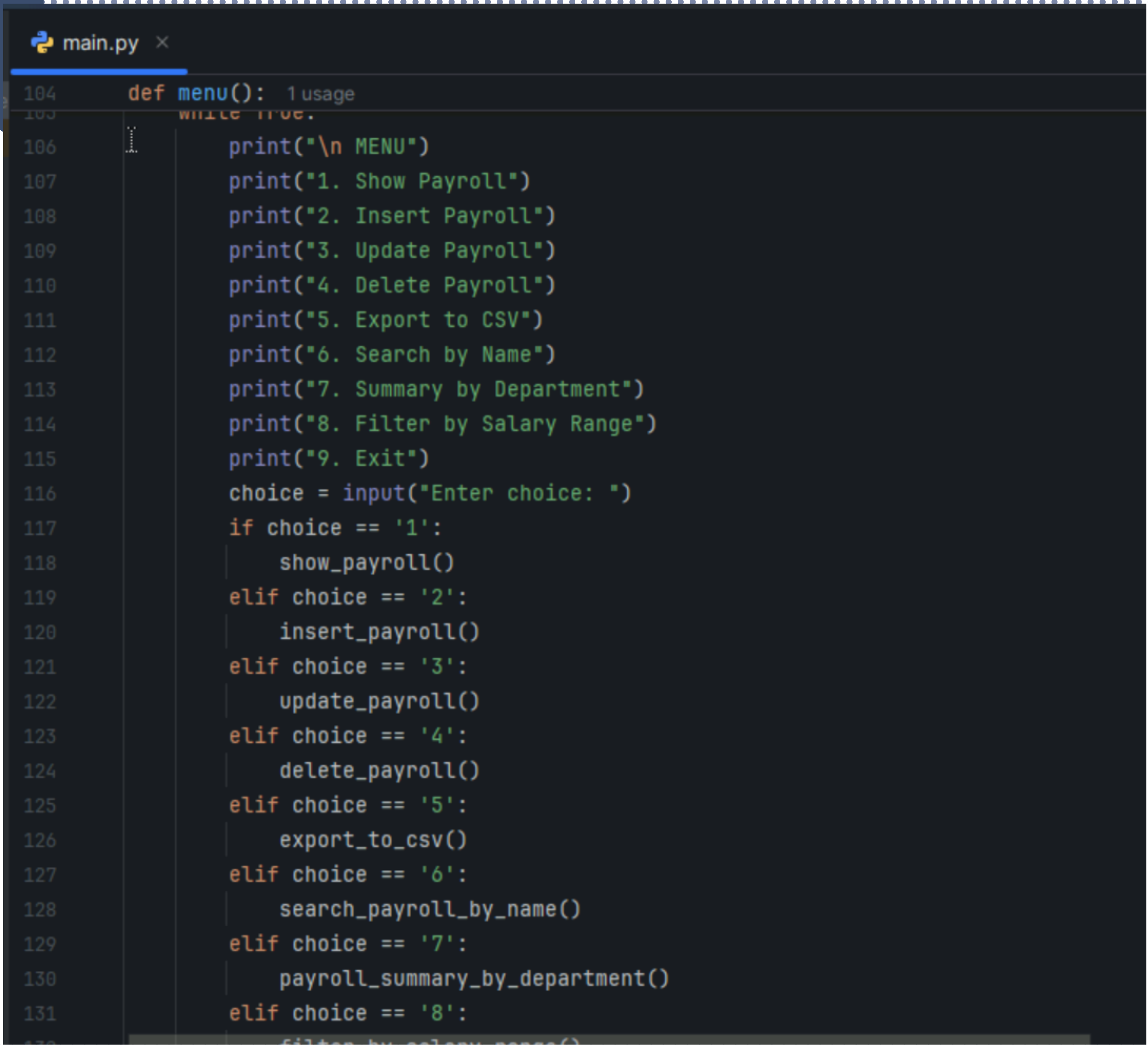
Activate Windows



```

def delete_payroll():  1 usage
    emp_id = int(input("Enter payroll ID to delete: "))
    cursor.execute( query: "DELETE FROM payroll WHERE id=%s",  args: (emp_id,))
    conn.commit()
    print(" Payroll record deleted.")
def export_to_csv():  1 usage
    cursor.execute("SELECT * FROM payroll")
    rows = cursor.fetchall()
    with open('payroll.csv', 'w', newline='') as file:
        writer = csv.writer(file)
        writer.writerow(['ID', 'Name', 'Department', 'Basic Salary', 'Allowances', 'Deductions', 'Net Pay'])
        writer.writerows(rows)
    print(" Data exported to payroll.csv")


def search_payroll_by_name():  1 usage
    name = input("Enter name to search: ")
    cursor.execute( query: "SELECT * FROM payroll WHERE name LIKE %s",  args: ('%' + name + '%',))
    results = cursor.fetchall()
    if results:
        for row in results:
            print(f"ID: {row[0]}, Name: {row[1]}, Dept: {row[2]}, Basic: {row[3]}, Allowances: {row[4]}, Deducti
    else:
        print("No matching records found.")

```

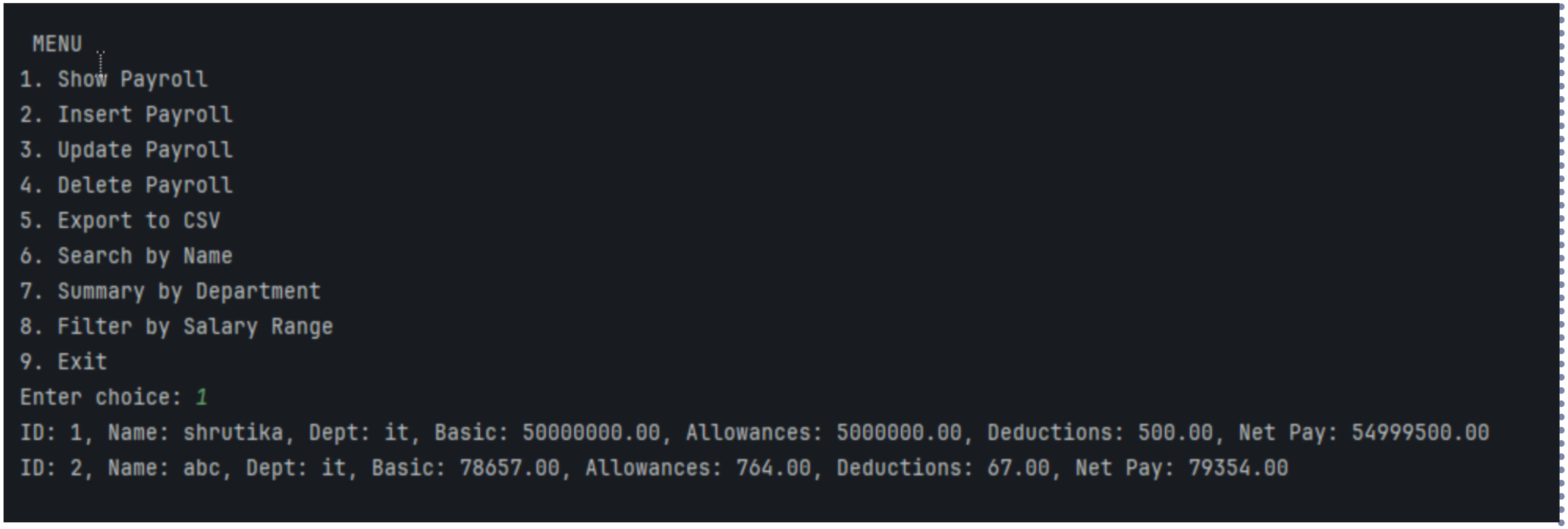


```
def payroll_summary_by_department():  1 usage
    cursor.execute("""
        SELECT department, COUNT(*), SUM(basic_salary), SUM(allowances), SUM(deductions), SUM(net_pay)
        FROM payroll
        GROUP BY department
    """)
    for row in cursor.fetchall():
        print(f"Dept: {row[0]}, Employees: {row[1]}, Total Basic: {row[2]}, Total Allowances: {row[3]}, Total Dec


def filter_by_salary_range():  1 usage
    min_salary = float(input("Enter minimum net pay: "))
    max_salary = float(input("Enter maximum net pay: "))
    cursor.execute( query: "SELECT * FROM payroll WHERE net_pay BETWEEN %s AND %s",  args: (min_salary, max_salary))
    results = cursor.fetchall()
    if results:
        for row in results:
            print(f"ID: {row[0]}, Name: {row[1]}, Dept: {row[2]}, Net Pay: {row[6]}")
    else:
        print("No employees found in this salary range.")
```

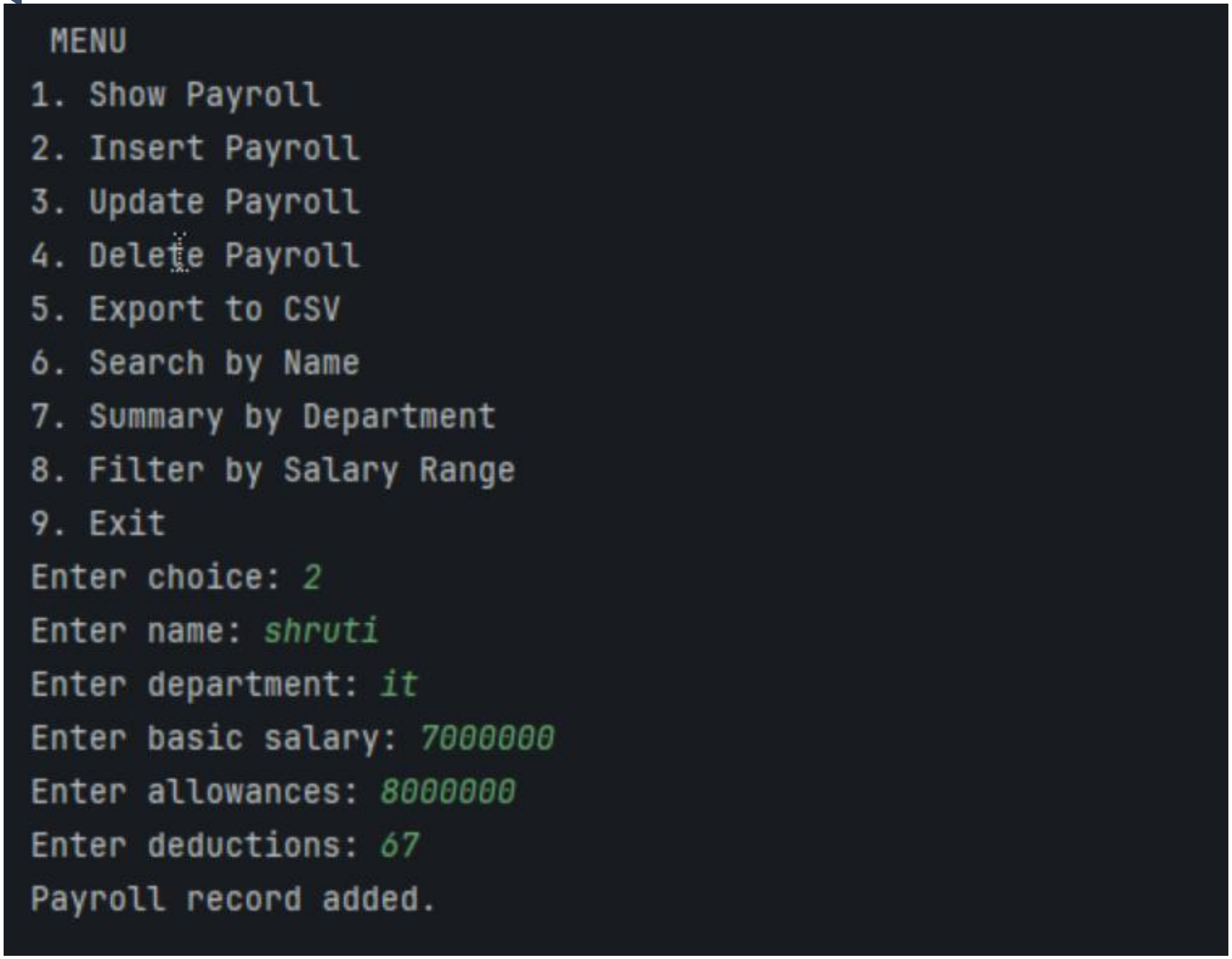


```python
def menu():
    while True:
        print("\n MENU")
        print("1. Show Payroll")
        print("2. Insert Payroll")
        print("3. Update Payroll")
        print("4. Delete Payroll")
        print("5. Export to CSV")
        print("6. Search by Name")
        print("7. Summary by Department")
        print("8. Filter by Salary Range")
        print("9. Exit")
        choice = input("Enter choice: ")
        if choice == '1':
            show_payroll()
        elif choice == '2':
            insert_payroll()
        elif choice == '3':
            update_payroll()
        elif choice == '4':
            delete_payroll()
        elif choice == '5':
            export_to_csv()
        elif choice == '6':
            search_payroll_by_name()
        elif choice == '7':
            payroll_summary_by_department()
        elif choice == '8':
            filter_by_salary_range()
```

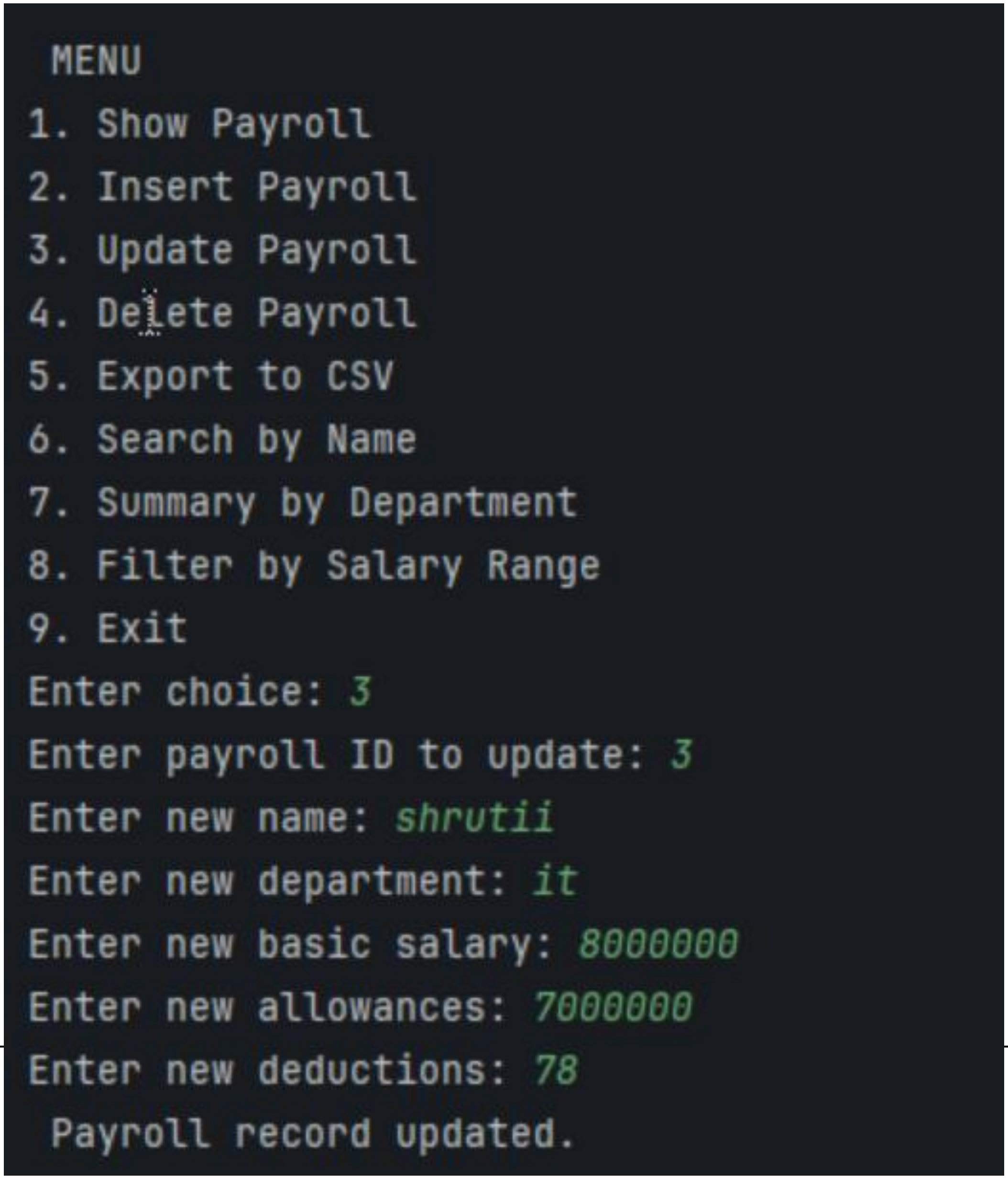
Output:



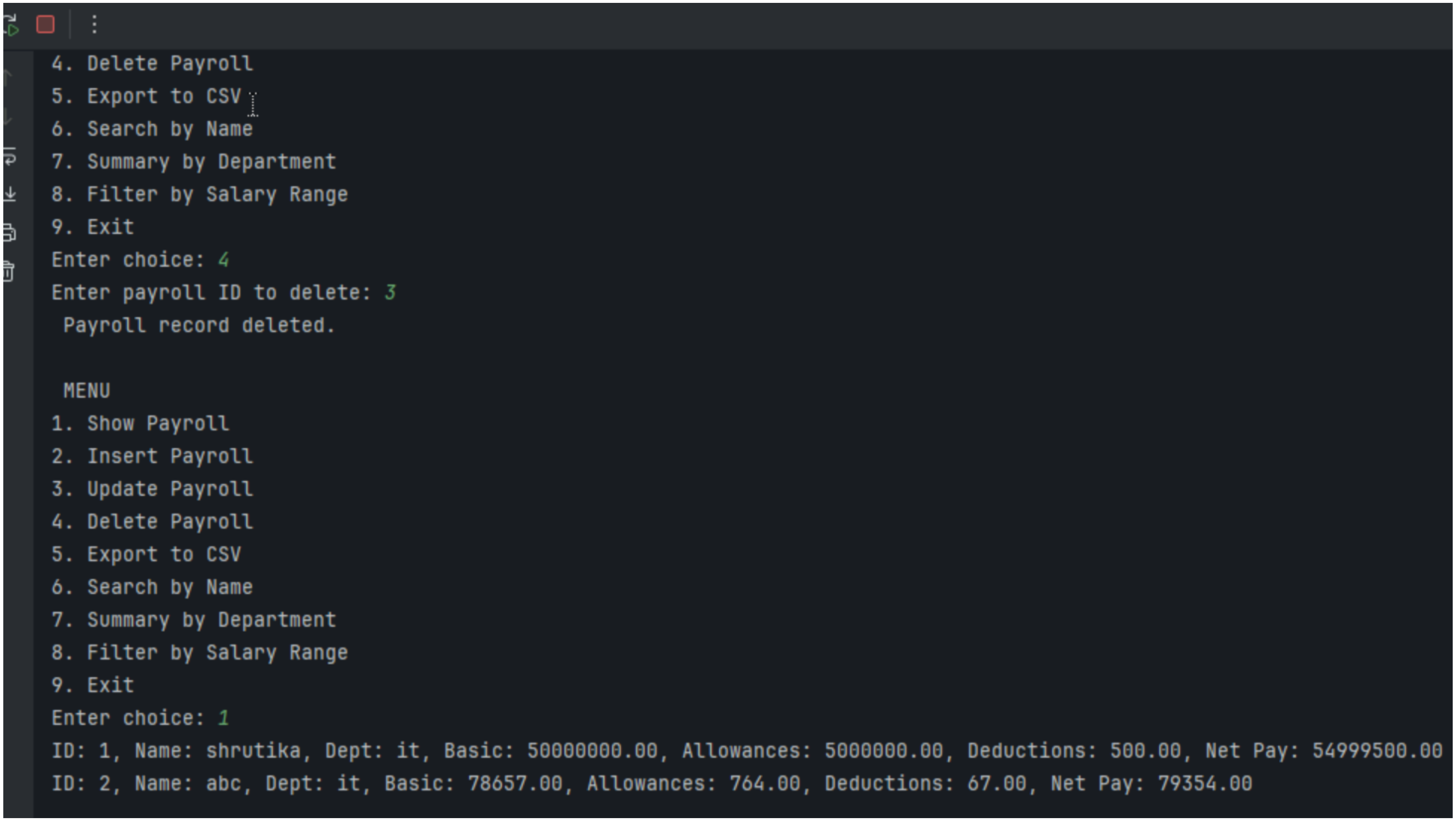
```
 MENU
1. Show Payroll
2. Insert Payroll
3. Update Payroll
4. Delete Payroll
5. Export to CSV
6. Search by Name
7. Summary by Department
8. Filter by Salary Range
9. Exit
Enter choice: 1
ID: 1, Name: shrutika, Dept: it, Basic: 50000000.00, Allowances: 5000000.00, Deductions: 500.00, Net Pay: 54999500.00
ID: 2, Name: abc, Dept: it, Basic: 78657.00, Allowances: 764.00, Deductions: 67.00, Net Pay: 79354.00
```



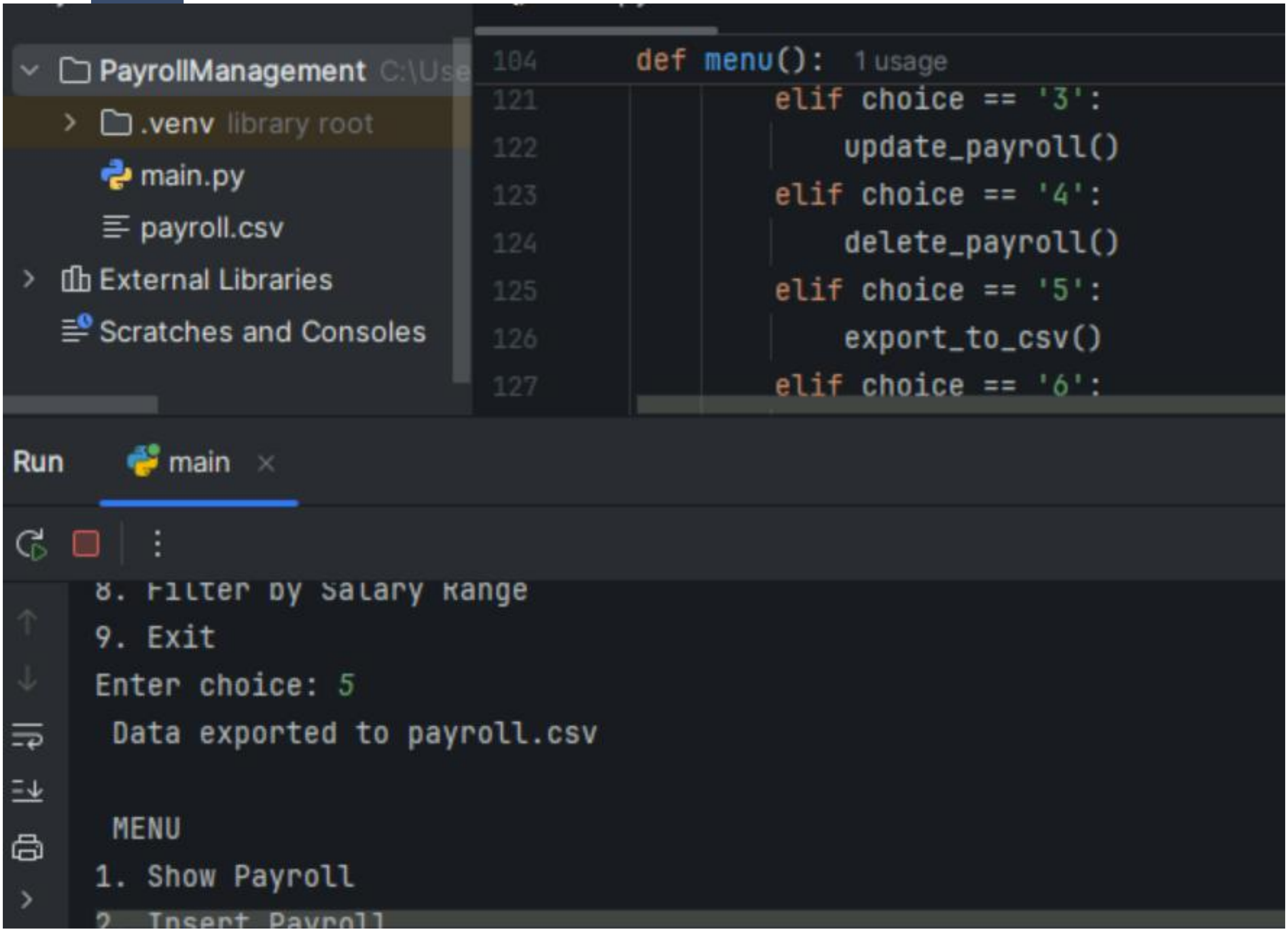
```
 MENU
1. Show Payroll
2. Insert Payroll
3. Update Payroll
4. Delete Payroll
5. Export to CSV
6. Search by Name
7. Summary by Department
8. Filter by Salary Range
9. Exit
Enter choice: 2
Enter name: shruti
Enter department: it
Enter basic salary: 7000000
Enter allowances: 8000000
Enter deductions: 67
Payroll record added.
```



```
 MENU
1. Show Payroll
2. Insert Payroll
3. Update Payroll
4. Delete Payroll
5. Export to CSV
6. Search by Name
7. Summary by Department
8. Filter by Salary Range
9. Exit
Enter choice: 3
Enter payroll ID to update: 3
Enter new name: shrutii
Enter new department: it
Enter new basic salary: 8000000
Enter new allowances: 7000000
Enter new deductions: 78
 Payroll record updated.
```

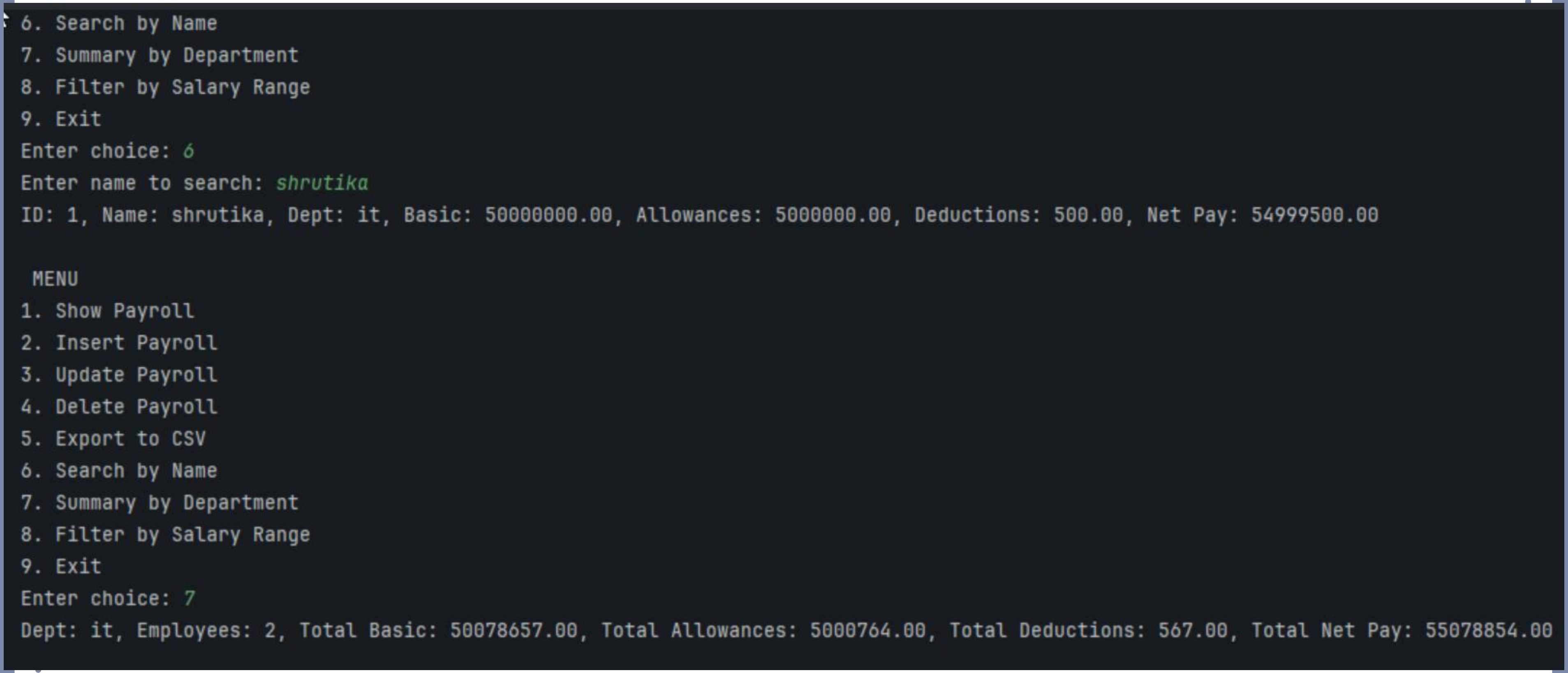


```
4. Delete Payroll
5. Export to CSV
6. Search by Name
7. Summary by Department
8. Filter by Salary Range
9. Exit
Enter choice: 4
Enter payroll ID to delete: 3
 Payroll record deleted.

 MENU
1. Show Payroll
2. Insert Payroll
3. Update Payroll
4. Delete Payroll
5. Export to CSV
6. Search by Name
7. Summary by Department
8. Filter by Salary Range
9. Exit
Enter choice: 1
ID: 1, Name: shrutika, Dept: it, Basic: 50000000.00, Allowances: 5000000.00, Deductions: 500.00, Net Pay: 54999500.00
ID: 2, Name: abc, Dept: it, Basic: 78657.00, Allowances: 764.00, Deductions: 67.00, Net Pay: 79354.00
```

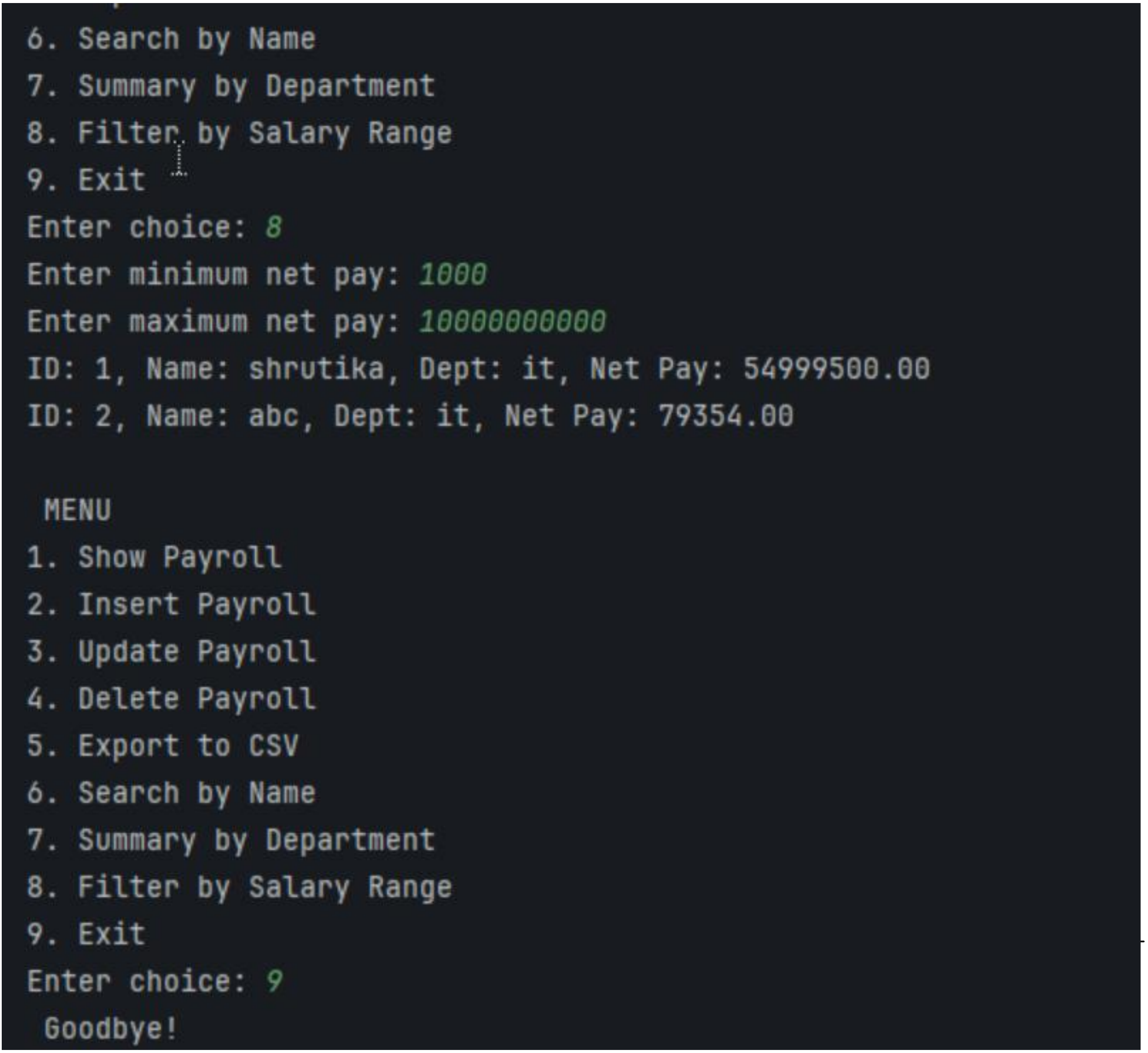


```python
def menu():
        elif choice == '3':
            update_payroll()
        elif choice == '4':
            delete_payroll()
        elif choice == '5':
            export_to_csv()
        elif choice == '6':
```

```
8. Filter by Salary Range
9. Exit
Enter choice: 5
 Data exported to payroll.csv

 MENU
1. Show Payroll
2. Insert Payroll
```



```
6. Search by Name
7. Summary by Department
8. Filter by Salary Range
9. Exit
Enter choice: 6
Enter name to search: shrutika
ID: 1, Name: shrutika, Dept: it, Basic: 50000000.00, Allowances: 5000000.00, Deductions: 500.00, Net Pay: 54999500.00

 MENU
1. Show Payroll
2. Insert Payroll
3. Update Payroll
4. Delete Payroll
5. Export to CSV
6. Search by Name
7. Summary by Department
8. Filter by Salary Range
9. Exit
Enter choice: 7
Dept: it, Employees: 2, Total Basic: 50078657.00, Total Allowances: 5000764.00, Total Deductions: 567.00, Total Net Pay: 55078854.00
```



```
6. Search by Name
7. Summary by Department
8. Filter by Salary Range
9. Exit
Enter choice: 8
Enter minimum net pay: 1000
Enter maximum net pay: 10000000000
ID: 1, Name: shrutika, Dept: it, Net Pay: 54999500.00
ID: 2, Name: abc, Dept: it, Net Pay: 79354.00

 MENU
1. Show Payroll
2. Insert Payroll
3. Update Payroll
4. Delete Payroll
5. Export to CSV
6. Search by Name
7. Summary by Department
8. Filter by Salary Range
9. Exit
Enter choice: 9
 Goodbye!
```