ICP2

BODIGE SAINATH

700756620

Git : https://github.com/BodigeSainath/NNDL/tree/icp2

Video : https://screenrec.com/share/JLA1Do9yNt

Q1

[4]:
```python
class Employee:
    employee_count = 0
    total_salary = 0

    def __init__(self, name, family, salary, department):
        self.name = name
        self.family = family
        self.salary = salary
        self.department = department
        Employee.employee_count += 1
        Employee.total_salary += salary

    @classmethod
    def average_salary(cls):
        if cls.employee_count > 0:
            return cls.total_salary / cls.employee_count
        return 0

    def display_employee(self):
        return f"Employee: {self.name}, Family: {self.family} Department: {self.department}, Salary: {self.salary}"

class FulltimeEmployee(Employee):
    ft_employee_count = 0
    ft_total_salary = 0

    def __init__(self, name, family, salary, department):
        super().__init__(name, family, salary, department)
        FulltimeEmployee.ft_employee_count += 1
        FulltimeEmployee.ft_total_salary += salary
```

```python
    @classmethod
    def average_salary(cls):
        if cls.ft_employee_count > 0:
            return cls.ft_total_salary / cls.ft_employee_count
        return 0

    def display_employee(self):
        return f"Fulltime Employee: {self.name}, Family: {self.family} Department: {self.department}, Salary: {self.salary}"

# Creating instances
emp1 = Employee("Sainath", "Bodige", 100000, "IT Development")
emp2 = Employee("Suman", "D", 95000, "Marketing")
ft_emp1 = FulltimeEmployee("Rohit", "Reddy", 70000, "Human Resource")
ft_emp2 = FulltimeEmployee("Priya", "K", 80000, "Finance")

# Displaying information
print(emp1.display_employee())
print(emp2.display_employee())
print(ft_emp1.display_employee())
print(ft_emp2.display_employee())

# Displaying average salaries and counts
print(f"Average Salary of Employees: {Employee.average_salary()}")
print(f"Number of Employees: {Employee.employee_count}")
print(f"Average Salary of Fulltime Employees: {FulltimeEmployee.average_salary()}")
print(f"Number of Fulltime Employees: {FulltimeEmployee.ft_employee_count}")
```

Output

```
Employee: Sainath, Family: Bodige Department: IT Development, Salary: 100000
Employee: Suman, Family: D Department: Marketing, Salary: 95000
Fulltime Employee: Rohit, Family: Reddy Department: Human Resource, Salary: 70000
Fulltime Employee: Priya, Family: K Department: Finance, Salary: 80000
Average Salary of Employees: 86250.0
Number of Employees: 4
Average Salary of Fulltime Employees: 75000.0
Number of Fulltime Employees: 2
```

Q2

```python
import numpy as np

# Create random vector of size 20 with floats in the range 1-20
random_vector = np.random.uniform(1, 20, 20)

# Reshape the array to 4 by 5
reshaped_array = random_vector.reshape(4, 5)

# Replace the max in each row by 0
reshaped_array[reshaped_array == reshaped_array.max(axis=1, keepdims=True)] = 0

reshaped_array
```

Output

```
array([[11.3040334 ,  7.10693206,  9.32762676,  0.        ,  8.03959706],
       [ 0.        ,  2.03449646, 15.53546164,  1.05751896, 12.41856938],
       [ 8.35471883,  8.71903467, 12.29108148,  0.        , 10.41056685],
       [ 9.06762737,  7.9426185 ,  3.02267439, 13.46843113,  0.        ]])
```
