ICP2

Tejaswini Tatikonda – 700762235

Github link: https://github.com/tejaswini22350/Assignment-2.git
Video link:
https://drive.google.com/file/d/197GipWnhP7KMdZwJjRX8Zw6naDc1L6Hl/view?usp=share_link

In class programming:

1. Create a class Employee and then do the following
• Create a data member to count the number of Employees
• Create a constructor to initialize name, family, salary, department
• Create a function to average salary
• Create a Fulltime Employee class and it should inherit the properties of Employee class
• Create the instances of Fulltime Employee class and Employee class and call their
member functions.

```python
class Employee:
  employee_count = 0

  def __init__(self, name, family, salary, department):
      self.name = name
      self.family = family
      self.salary = salary
      self.department = department

      Employee.employee_count += 1

  def average_salary(self, *salaries):
      total_salary = sum(salaries)
      return total_salary / len(salaries) if len(salaries) > 0 else 0

class FulltimeEmployee(Employee):
  def __init__(self, name, family, salary, department, working_hours):
      super().__init__(name, family, salary, department)

      self.working_hours = working_hours

employee1 = Employee("Tejaswini", "Family1", 75000, "SE1")
employee2 = Employee("Syalini", "Family2", 50000, "ASE")

fulltime_employee = FulltimeEmployee("Chinnu", "Family3", 25000, "IT", 38)

average_salary_all = employee1.average_salary(employee1.salary, employee2.salary)
average_salary_fulltime = fulltime_employee.average_salary(fulltime_employee.salary)

# Print results
print("Total Employees:", Employee.employee_count)
print("Average Salary of all Employees:", average_salary_all)
print("Average Salary of Fulltime Employee:", average_salary_fulltime)
```

```
Total Employees: 3
Average Salary of all Employees: 62500.0
```

2. NumPy

Using NumPy create random vector of size 20 having only float in the range 1-20.

Then reshape the 2array to 4 by 5

Then replace the max in each row by 0 (axis=1)

(you can NOT implement it via for loop)

```python
import numpy as np

random_vector = np.random.uniform(1, 20, 20)
reshaped_array = random_vector.reshape(4, 5)
reshaped_array[np.arange(len(reshaped_array)), reshaped_array.argmax(axis=1)] = 0

print("Original Random Vector:")
print(random_vector)
print("\nReshaped Array (4 by 5):")
print(reshaped_array)
```

```
Original Random Vector:
[ 0.          5.81170753  5.7689511   6.23592308  9.19023956 10.72492816
  4.35178333 11.10326294  0.          8.31644019 17.98468246  6.68562868
  5.51177351  0.         19.81094382 17.76133543  6.10413814  2.16216652
  0.         11.35365254]

Reshaped Array (4 by 5):
[[ 0.          5.81170753  5.7689511   6.23592308  9.19023956]
 [10.72492816  4.35178333 11.10326294  0.          8.31644019]
 [17.98468246  6.68562868  5.51177351  0.         19.81094382]
 [17.76133543  6.10413814  2.16216652  0.         11.35365254]]
```