

Neural Networks & Deep Learning Assignment 02

Name: Sai Rohith Meela

700#: 700758505

GitHub Link: https://github.com/Rohithmeela15/NNDL-ICPL01_700758505

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.

```
self.department = department
Employee.num_employees += 1

@classmethod
def average_salary(cls, employees):
    total_salary = sum(emp.salary for emp in employees)
    return total_salary / len(employees)

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

employee1 = Employee("Meela", "Family 1", 55000, "manager")
employee2 = Employee("Sai", "Family 2", 80000, "IT")
employee3 = Employee("Rohith", "Family3", 150000, "Team lead")
employee4 = Employee("Msr", "Family 4", 160000, "Team lead")

fulltime_employee1 = FulltimeEmployee("Nick", "Family 5", 90000, "hospital", "Healthcare")
fulltime_employee2 = FulltimeEmployee("Ram", "Family 6", 90000, "Finance", "manager")
```

```
employees = [employee1, employee2, employee3, employee4, fulltime_employee1, fulltime_employee2]
avg_salary = Employee.average_salary(employees)

print(f"Total number of employees: {Employee.num_employees}")
print(f"Average salary: {avg_salary}")
```


```
Total number of employees: 6
Average salary: 104166.66666666667
```

2. Numpy

Using NumPy create random vector of size 20 having only float in the range 1-20.
Then reshape the array to 4 by 5
Then replace the max in each row by 0 (axis=1)
(you can NOT implement it via for loop)

```
import numpy as np

random_vector = np.random.uniform(1, 20, size=20)
reshaped_array = random_vector.reshape(4, 5)
max_indices = np.argmax(reshaped_array, axis=1)
col_indices = np.arange(5)
reshaped_array[np.arange(4)[:, np.newaxis], max_indices] = 0
print(reshaped_array)
```



[0.	0.	1.90639322	0.	12.1606614]
[0.	0.	17.61307578	0.	2.47672815]
[0.	0.	2.04965655	0.	13.10123096]
[0.	0.	14.91379835	0.	1.89994471]