

Spring 2024: CS5720

NEURAL NETWORK AND DEEP LEARNING

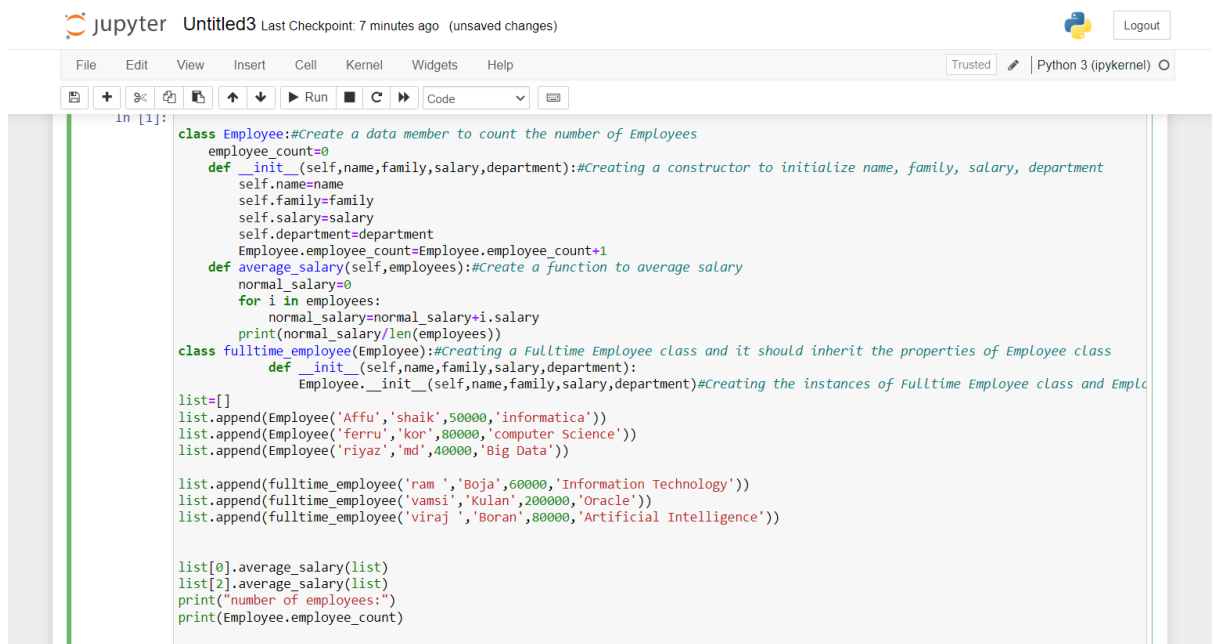
ICP-3 CRN: 22317 Name: Afroz Mohammad [700758012]

Git Hub Link: <https://github.com/Afrozmohammad19/Assignment3>

Video

Link: <https://drive.google.com/file/d/1c8ghJHR9rol4IEMgM0ZKOt6ngNqRYIAD/view?usp=sharing>

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.



```
in [1]:
class Employee:#Create a data member to count the number of Employees
    employee_count=0
    def __init__(self,name,family,salary,department):#Creating a constructor to initialize name, family, salary, department
        self.name=name
        self.family=family
        self.salary=salary
        self.department=department
        Employee.employee_count=Employee.employee_count+1
    def average_salary(self,employees):#Create a function to average salary
        normal_salary=0
        for i in employees:
            normal_salary=normal_salary+i.salary
        print(normal_salary/len(employees))
class fulltime_employee(Employee):#Creating a Fulltime Employee class and it should inherit the properties of Employee class
    def __init__(self,name,family,salary,department):
        Employee.__init__(self,name,family,salary,department)#Creating the instances of Fulltime Employee class and Employee class

list=[]
list.append(Employee('Affu','shaik',50000,'informatics'))
list.append(Employee('ferru','kor',80000,'computer Science'))
list.append(Employee('riyaz','md',40000,'Big Data'))

list.append(fulltime_employee('ram ','Boja',60000,'Information Technology'))
list.append(fulltime_employee('vamsi','Kulan',200000,'Oracle'))
list.append(fulltime_employee('viraj ','Boran',80000,'Artificial Intelligence'))

list[0].average_salary(list)
list[2].average_salary(list)
print("number of employees:")
print(Employee.employee_count)
```

Output:

```
85000.0
85000.0
number of employees:
6
```

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
# Create random vector of size 20 with floats between 1 and 20
vec = np.random.uniform(6, 4, 20)
# Reshape the vector to 4 by 5
mat = vec.reshape(4, 5)
# Replacing the max in each row by 0
mat[np.arange(4), mat.argmax(axis=1)] = 0
# Print the output
print(mat)
```

Output:

```
[[4.06387688 0.          4.73714352 5.09483815 5.10260552]
 [0.          4.54910646 5.48952827 4.10878087 4.10095593]
 [4.94031129 4.29577979 4.15623534 0.          5.22557876]
 [0.          5.68966272 5.65071936 5.10694858 5.27398924]]
```