Student ID: 700752115

Student Name: Rumana Shaik

Neural Networks & Deep Learning - ICP-3

Github link: https://github.com/ShaikRumana301/Neural-Network-DL-ICP-3.git

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 [36]:
```
class Employee:
    no_of_emp=0
    def __init__(self,name,family,salary,department): #constructor to initialise Name,Family,Salary and Department.
        self.name =name
        self.family =family
        self.salary=salary
        self.department=department
        Employee.no_of_emp+=1 #counting the number of employees.
    def avg_salary(self,emp):  #function for calculating the average salary of all employees.
        sal=sum(emp.salary for emp in emp)
        print("The average salary of the all the Employees and Full time Employees are: ",sal/len(emp))

class Fulltime_Employee(Employee):
    pass #using pass to inherit the Employee class

n=int(input("Please Enter number of Employees: "))
emp=[]
for i in range(0,n):  #using for loop to take the input dynamically.
    na=input("Enter Employee name: ")
    fam=int(input("Enter number of family members: "))
    sal=int(input("Enter Salary: "))
    dept=input("Enter the Employee department: ")
    obj=Employee(na,fam,sal,dept)
    emp.append(obj)

num_fte=int(input("Please Enter number of Full time employees: "))
for i in range(0,num_fte):
    na=input("Enter the Full time Employee name: ")
    fam=int(input("Enter number of family members: "))
    sal=int(input("Enter the Salary: "))
    dept=input("Enter the Full Time Employee Department: ")
    fte=Fulltime_Employee(na,fam,sal,dept)
    emp.append(fte)

print("The total Number of Employees and Full time Employess are: ",Employee.no_of_emp) #using datamember to print the employee c
result=Employee(na,fam,sal,dept)
result.avg_salary(emp) #function call to print the Employees avg salary
```

Output:

```
Please Enter number of Employees: 2
Enter Employee name: Rumana Shaik
Enter number of family members: 4
Enter Salary: 8790
Enter the Employee department: IT
Enter Employee name: Vani
Enter number of family members: 4
Enter Salary: 7890
Enter the Employee department: Cyber
Please Enter number of Full time employees: 1
Enter the Full time Employee name: Rahul
Enter number of family members: 3
Enter the Salary: 6780
Enter the Full Time Employee Department: Finance
The total Number of Employees and Full time Employess are:  3
The average salary of the all the Employees and Full time Employees are:  7820.0
```

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)

In [25]:
```
import numpy as np #import numpy
x = np.arange(1,21,dtype=float) #given vector of size 1-20
print("Vector :",x)
x=x.reshape(4,5) #use the reshape() method to reshape the array into 4*5
print("Then reshape the array to 4 by 5:",x)
def replace(x):
    b=x
    b[:,np.argmax(x, axis=1)] = 0 #replace the max in each row by 0 with axis =1
    return b
result= replace(x)
print("The Resultant array where replace the max in each row by 0 (axis=1) is:",result)
```

```
Vector : [ 1.  2.  3.  4.  5.  6.  7.  8.  9. 10. 11. 12. 13. 14. 15. 16. 17. 18.
 19. 20.]
Then reshape the array to 4 by 5: [[ 1.  2.  3.  4.  5.]
 [ 6.  7.  8.  9. 10.]
 [11. 12. 13. 14. 15.]
 [16. 17. 18. 19. 20.]]
The Resultant array where replace the max in each row by 0 (axis=1) is: [[ 1.  2.  3.  4.  0.]
 [ 6.  7.  8.  9.  0.]
 [11. 12. 13. 14.  0.]
 [16. 17. 18. 19.  0.]]
```