

LAB 8 - OOP

Part 1 : Employee, Instructor and Administrative_personnel

We used to manage the employee at IUT. We consider two kinds which are instructors and administrative personnel.

Employee contains 3 attributes **Lname**, **Fname**, **DDN**.

Instructor inherits from Employee and contains 3 attributes : Department and the code of two courses. Administrative personnel inherits from Employee and contains one attribute : function (director,

Create the 3 classes **Employee**, **Instructor**, and **Administrative_personnel** knowing that Employee is an abstract class.

- Write for each class constructors, properties and a method Display() which display the attributes.
- Define the attribute **NbEmployee** which is an **attribute of the class Employee** . Each time an Employee is created, NbEmployee is incremented.
- Override the **ToString()** method for each class.

Part 2 : Class listEmployee

listEmployee inherits from the class **ArrayList** :

class listEmployee :ArrayList{ ... }

So that it inherits from all methods and properties of ArrayList:

- **L.Add(Object o)** : add an object **o** to the listEmployee **L**.
- **L.Count** is a property which returns the number of element in the list.
- **L[i]** returns the element number **i** in the list.

Create the following methods:

- **AddEmployee(Employee E)** allows to add an Employee
- **DisplayEmployee()** Display all employees in the list
- **DisplayInstructor()** Display all Instructors in the list.

Remark :

L :typeListEmployee ,

To check if an object is an instructor you have to test each element of L and test (L[i] is

Enseignant)

Part 3 : Test

1. Create two objects I1 and I2 of the class Instructors.
2. Create two objects P1 et P2 of the class Instructors **Administrative_personnel**
3. Create an object L of the class ListEmployee (**listEmployee**) and add I1,I2,P1,P2.
4. Display all employees from L.
5. Display all Instructors from L.