```
//Class "Employee":

using System;
namespace LAB_8
{
public class Employee
{
private string Lname,Fname;
private DateTime dob;
private static int NbEmployee=0;
public static int NB
{
get{return NbEmployee;}
}
public string L
{
get{return Lname;}
set{Lname=value;}
}
public string F
{
get{return Fname;}

set{Fname=value;}
}
public DateTime D
{
get{return dob;}
set{dob=value;}
}
public Employee(string Lname,string Fname
                ,DateTime dob)
{
this.Lname=Lname;
```

```
this.Fname=Fname;
this.dob=dob;
NbEmployee++;
        }
public Employee()//Default Constructor
{
}
public string Display()
{
return "Fname:"+F+"\nLname:"+L+"\nDate of birth"+D;
}
public override string ToString()
{
return "Fname:"+F+"\nLname:"+L+"\nDate of birth:"
        +D.DayOfYear;
}
}
}



//Class "Instructor":
using System;
namespace LAB_8
{
public class Instructor:Employee
{
private string department,code1,code2;
public Instructor()
{
}
public new string D
{
get{return department;}
```

```
set{department=value;}
}
public string C1
{
get{return code1;}
set{code1=value;}
}
public string C2
{
get{return code2;}
set{code2=value;}
}

public Instructor(string department,string code1,
          string code2,string Lname,string Fname,
          DateTime dob):base(Lname,Fname,dob)
{
this.department=department;
this.code1=code1;
this.code2=code2;
}
public new string Display()
{
return "Fname:"+F+"\nLname:"+L+"\nDepartment:"
        +department+"\nCode1:"+code1+"\nCode2"
        +code2+"\nDate of birth"+D;
}
public override string ToString()
{
return string.Format("Inst. Department={0},Code1={1}
                     ,Code2={2}",department,code1,code2)
                     +base.ToString();
}
```

```
}
}
```

```
//Class "Adminstrator_person":
using System;
namespace LAB_8
{
public class Adminsrator_person:Employee
{
private string function;
public Adminsrator_person(string function,
        string Lname,string Fname,
        DateTime dob):base(Lname,Fname,dob)
{
this.function=function;
}
public new string Display()
{
return "Fname:"+F+"\nLname:"+L+"\nDate of birth:"
+D+"\nFunction:"+function;
}
public override string ToString()
{
return string.Format("Adminsrator Function={0}"
                    ,function)+base.ToString();
}
public Adminsrator_person()
{
}
}
}
```

```
//Class "listEmployee":
using System;
using System.Collections;
namespace LAB_8
{
public class listEmployee:ArrayList
{
public void AddEmployee(Employee E)
{
Add(E);
}
public void DisplayEmployee()
{
for(int i=0;i<Count;i++)
Console.WriteLine(this[i]);
}
public void DisplayInstructor()
{
foreach(Employee I in this)
if(I is Instructor)
Console.WriteLine(I.ToString());
}
}
}
```

```
//Main:
using System;
namespace LAB_8
{
class Program
{
public static void Main(string[] args)
```

```
{
Instructor L1,L2;
Adminsrator_person P1,P2;
L1=new Instructor("CCNE","192bh","mj217","Rammal"
                  ,"Ali",new DateTime(1982,8,3));
L2=new Instructor();
P1=new Adminsrator_person("Director","Alaa","Nour",
                          new DateTime(2016,8,3));
P2=new Adminsrator_person();
listEmployee L=new listEmployee();
L.Add(L1);
L.Add(L2);
L.Add(P1);
L.Add(P2);
L.DisplayEmployee();
L.DisplayInstructor();

Console.Write("Press any key to continue . . . ");
Console.ReadKey(true);
}
}
}
```