DAY 2

DATE:05-08-25
NAME: NANCY M

Program.cs:

```
using Day2Daily_Task.Repo;
namespace Day2Daily_Task;

class Program
{
   static void Main(string[] args)
   {
      Employeedetails emp_1 = new Employeedetails(1, "john", "manager",2);
      Employeedetails emp_2 = new Employeedetails(2, "jcy", "ui", 1);
      Employeedetails emp_3 = new Employeedetails(3, "jack", "dev", 4);
      Employeedetails emp_4 = new Employeedetails(4, "ncy", "ba", 1);

      IEmployeeRepo<Employeedetails> repo = new
EmployeeRepo<Employeedetails>();
      repo.Add(emp_1);
      repo.Add(emp_2);
      repo.Add(emp_3);
      repo.Add(emp_4);
      foreach (Employeedetails emp in repo.GetAllEmp())
      {
         Console.WriteLine(emp.GetDetails());
      }

   }
}
```

Employee.cs:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
```

```
using System.Threading.Tasks;

namespace Day2Daily_Task
{
    abstract class Employee
    {
        public int Id { get; set; }
        public string Name { get; set; }
        public string Designation { get; set; }
        public int Experience { get; set; }
        public Employee()
        {
            Console.WriteLine("Constructor is called: ");
        }

        public Employee(int id, string name, string designation, int experience)
        {
            Id = id;
            Name = name;
            Designation = designation;
            Experience = experience;
        }

        public abstract string GetDetails();

        ~Employee()
        {
            Console.WriteLine("Destroyed: " + Name);
        }
    }
}
```

EmployeeDetails.cs:

```
using System;
using System.Collections.Generic;
using System.Diagnostics;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
```

```
using System.Xml.Linq;

namespace Day2Daily_Task
{
    internal class Employeedetails : Employee
    {
        public Employeedetails(int Id, string Name, string Designation, int Experience) :
base(Id, Name, Designation, Experience)
        {

        }
        public override string GetDetails()
        {
            string details = string.Format("Id:  {0}, Name: {1}, Designation {2}, Experience:
{3}", Id, Name, Designation, Experience);
            return details;
        }
    }
}
```

IEmployeeRepo.cs:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Day2Daily_Task.Repo
{
    internal interface IEmployeeRepo<T>
    {
        List<T> GetAllEmp();
        void Add(T Employee);

    }
}
```

EmployeeRepo.cs:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Day2Daily_Task.Repo
{
    internal class EmployeeRepo<T> : IEmployeeRepo<T>
    {
        List<T> listOfEmp = new List<T>();
        public void Add(T Employee)
        {
            listOfEmp.Add(Employee);
        }
        public List<T> GetAllEmp()
        {
            return listOfEmp;
        }
    }
}
```

Output:

```
Id:  1, Name: john, Designation: manager, Experience: 2
Id:  2, Name: jcy, Designation: ui, Experience: 1
Id:  3, Name: jack, Designation: dev, Experience: 4
Id:  4, Name: ncy, Designation: ba, Experience: 1
```