

Final Project

By M Mary Margarette

Employee Management Application

METHODS TO BE USED:

- Add Employee
- Search Employee
- Display All Employees

VARIABLES TO BE USED:

- Employee Name (Should more than 4 char).
- Employee ID (Shouldn't be negative).
- Employee Salary (Min = 10,0000).
- Employee Age (age>=18, age <=58).

--

```

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

namespace DataAccessLibrary
{
    4 references
    public static class EmpDAL
    {
        public static string filepath = "E:\\NH Assignments\\Final Project\\EmployeeData.txt";
        1 reference
        public static bool AddEmployee(int empid, string empname, int empsalary, int empage)...

        1 reference
        public static List<String> GetEmployeeById(int empid)...

        1 reference
        public static List<String> GetEmployeeByName(string empname)...

        1 reference
        public static string[] GetAllEmployee()...

    }
}

```

Data Access Layer

```

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

namespace DataAccessLibrary
{
    public static class EmpDAL
    {
        public static string filepath = "E:\\NH Assignments\\Final Project\\EmployeeData.txt";
        public static bool AddEmployee(int empid, string empname, int empsalary, int empage)
        {
            try
            {
                String textcontent = String.Concat(empid, ",", empname, ",", empsalary, ",", empage);
                File.AppendAllText(filepath, textcontent + Environment.NewLine);
                return true;
            }
        }
    }
}

```

```

    }
    catch (Exception ex)
    {
        return false;
    }
}

public static List<String> GetEmployeeById(int empid)
{
    var allEmployees = File.ReadAllLines(filepath);
    bool isFound = false;
    List<String> EmployeeFound = new List<String>();
    foreach (string employee in allEmployees)
    {
        var empDetails = employee.Split(',');
        if(Convert.ToInt32(empDetails[0]) == empid)
        {
            isFound = true;
            EmployeeFound.Add(employee);
            break;
        }
    }
    return EmployeeFound;
}

public static List<String> GetEmployeeByName(string empname)
{
    var allEmployees = File.ReadAllLines(filepath);
    List<String> EmployeeFound = new List<String>();
    foreach (string employee in allEmployees)
    {
        var empDetails = employee.Split(',');
        if (empDetails[1] .Contains(empname))
        {
            EmployeeFound.Add(employee);
        }
    }
    return EmployeeFound;
}

public static string[] GetAllEmployee()

```

```

    {
        var result = File.ReadAllLines(filepath);
        return result;
    }

}

```

Business Logic Layer

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.IO;
using DataAccessLibrary;

namespace BusinessLogicLibrary
{
    public static class EmpBLL
    {
        private static string filepath;

        public static bool AddEmployee(int empid, string empname, int empsalary, int empage)
        {
            var result = EmpDAL.AddEmployee(empid, empname, empsalary, empage);
            return result;
        }

        public static List<String> GetEmployeeById(int empid)
        {
            var result = EmpDAL.GetEmployeeById(empid);
            return result;
        }

        public static List<String> GetEmployeeByName(string empname)
        {
            var result = EmpDAL.GetEmployeeByName(empname);
            return result;
        }
    }
}

```

```

    }

    public static string[] GetAllEmployee()
    {
        var result = EmpDAL.GetAllEmployee();
        return result;
    }
}

```

UI / Presentation Layer

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using BusinessLogicLibrary;
using System.IO;

namespace ClientApp
{
    internal class Program
    {
        static void Main(string[] args)
        {
            int ch;
            string choice;
            do
            {
                Console.WriteLine("-----");
                Console.WriteLine("Employee Managment Application");
                Console.WriteLine("-----");
                Console.WriteLine("1.Add Employee");
                Console.WriteLine("2.Get Employee By ID");
                Console.WriteLine("3.Get Employee By Name");
                Console.WriteLine("4.Get All Employess");
                Console.WriteLine("Enter Choice :");
                ch = Convert.ToInt32(Console.ReadLine());
                switch (ch)
                {

```

```

        case 1:
            AddEmployee();
            break;
        case 2:
            GetEmployeeById();
            break;
        case 3:
            GetEmployeeByName();
            break;
        case 4:
            GetAllEmployees();
            break;
        default:
            Console.WriteLine("Invalid Option");
            break;
    }
    Console.WriteLine("Do You want to continue Y/N : ");
    choice = Console.ReadLine();
}while (choice.Equals("Y"));

}

public static void AddEmployee()
{
    int empid, empage, empsalary;
    string empname;
    Console.WriteLine("Enter ID :");
    empid = Convert.ToInt32(Console.ReadLine());
    Console.WriteLine("Enter salary :");
    empsalary = Convert.ToInt32(Console.ReadLine());
    Console.WriteLine("Enter age :");
    empage = Convert.ToInt32(Console.ReadLine());
    Console.WriteLine("Enter Name :");
    empname =Console.ReadLine();

    var result = EmpBLL.AddEmployee(empid, empname, empsalary,empage);

    if(result)
        Console.WriteLine("Employee Details Saved Successfully");
    else
        Console.WriteLine("Some Error Occured");
}

public static void GetEmployeeById()
{
    int empid;
    Console.WriteLine("Enter ID :");
    empid = Convert.ToInt32(Console.ReadLine());

```

```
var result = EmpBLL.GetEmployeeById(empid);
if (result.Count == 0)
    Console.WriteLine("No Data");
else
    result.ForEach(d => Console.WriteLine(d));
}
public static void GetEmployeeByName()
{
    string empname;
    Console.WriteLine("Enter Name :");
    empname = Console.ReadLine();

    var result = EmpBLL.GetEmployeeByName(empname);
    if (result.Count == 0)
        Console.WriteLine("No Data");
    else
        result.ForEach(d => Console.WriteLine(d));
}
public static void GetAllEmployees()
{
    var result = EmpBLL.GetAllEmployee();
    result.ToList().ForEach(d=> Console.WriteLine(d));
}
}
}
```


--

E:\NH Assignments\Final Project\FinalProject\ClientApp\bin\Debug\ClientApp.exe

```
-----  
Employee Managment Application  
-----  
1.Add Employee  
2.Get Employee By ID  
3.Get Employee By Name  
4.Get All Employess  
Enter Choice :  
1  
Enter ID :  
34  
Enter salary :  
15000  
Enter age :  
23  
Enter Name :  
Mercy  
Employee Details Saved Successfully  
Do You want to continue Y/N :  
Y
```

E:\NH Assignments\Final Project\FinalProject\ClientApp\bin\Debug\ClientApp.exe

```
-----  
Employee Managment Application  
-----  
1.Add Employee  
2.Get Employee By ID  
3.Get Employee By Name  
4.Get All Employess  
Enter Choice :  
2  
Enter ID :  
27  
27,Rajendra,12000,28  
Do You want to continue Y/N :  
Y
```


 E:\NH Assignments\Final Project\FinalProject\ClientApp\bin\Debug\ClientApp.exe

Employee Managment Application

- 1.Add Employee
- 2.Get Employee By ID
- 3.Get Employee By Name
- 4.Get All Employess

Enter Choice :

4

26,Margaret,15000,25

27,Rajendra,12000,28

28,Mary,19000,23

29,Teja,17000,29

30,Rakesh,15000,55

31,Charan,19000,47

35,Prakash,16000,45

34,Mercy,15000,23

Do You want to continue Y/N :

Y