Day 22 Assignment

EMPLOYEE MANAGEMENT APPLICATION

By

VARUN SAI KUMAR CHEGONI

NB Healthcare and Technology

Date: 22 Feb 2022

**Topics**

**Project: Employee Management Application.**

|  |
| --- |
| Project Employee Management Application |
| Code : |
| Data Access Layer: |
| using System;  using System.Collections.Generic;  using System.IO;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace DataAccessLayer  {  public static class EmpDAL  {  public static string FilePath = "D:\\NB\_Training\\Training\_Assignments\\DotNET\_Assignments\\Day22(22 Feb)\\Employees.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 id)  {  var allEmployees = File.ReadAllLines(FilePath);  bool isFound = false;  List<String> empFound = new List<string>();    foreach(String employee in allEmployees)  {  var empDetails = employee.Split(',');  if (Convert.ToInt32(empDetails[0]) == id)  {  isFound = true;  empFound.Add(employee);  break;  }  }  return empFound;  }  public static List<String> GetEmployeeByName(string name)  {  var allEmployees = File.ReadAllLines(FilePath);  bool isFound = false;  List<String> empFound = new List<string>();  foreach (String employee in allEmployees)  {  var empDetails = employee.Split(',');  if (empDetails[1].Contains(name))  {  empFound.Add(employee);  }  }  return empFound;  }  public static string[] GetAllEmployee()  {  var allEmployees = File.ReadAllLines(FilePath);  return allEmployees;  }  }  } |
| Business Layer Library: |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  using DataAccessLayer;  namespace BusinessLogicLibrary  {  public class EmployeeBLL  {  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 id)  {  var result = EmpDAL.GetEmployeeById(id);  return result;  }  public static List<String> GetEmployeeByName(string name)  {  var result = EmpDAL.GetEmployeeByName(name);  return result;  }  public static string[] GetAllEmployee()  {  var result = EmpDAL.GetAllEmployee();  return result;  }  }  } |

|  |
| --- |
| Empolyee Client Application: |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  using BusinessLogicLibrary;  namespace EmpClientApp  {  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  \* Author : Varun Sai Kumar Chegoni.  \* Purpose : simple division program and handle three exceptions discussed in the class., also add super exception at the last.  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  internal class Program  {  static void Main(string[] args)  {  int choice1;  string choice2;  do  {  Console.WriteLine("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");  Console.WriteLine("Employee Management Application By Varun");  Console.WriteLine("========================================");  Console.WriteLine("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");  Console.WriteLine("1. Add Employee: ");  Console.WriteLine("2. Search Employee by ID: ");  Console.WriteLine("3. Search Employee by Name: ");  Console.WriteLine("4. Display All Employee: ");  Console.WriteLine("Enter your Choice: ");  choice1 = Convert.ToInt32(Console.ReadLine());  switch (choice1)  {  case 1:  AddEmployee();  break;  case 2:  SearchEmployeeById();  break;  case 3:  SearchEmployeeByName();  break;  case 4:  DisplayAllEmployees();  break;  default:  Console.WriteLine("Invalid Option");  break;  }  Console.WriteLine("Do you Wish to Continue (y/n): ");  choice2 = Console.ReadLine();  }  while (choice2.Equals("y"));  }  public static void AddEmployee()  {  int id, salary, age;  string name;  Console.WriteLine("Enter Employee ID: ");  id = Convert.ToInt32(Console.ReadLine());  Console.WriteLine("Enter Employee Name: ");  name = Console.ReadLine();  Console.WriteLine("Enter Employee Salary: ");  salary = Convert.ToInt32(Console.ReadLine());  Console.WriteLine("Enter Employee Age: ");  age = Convert.ToInt32(Console.ReadLine());  // Calling BLL Method  var result = EmployeeBLL.AddEmployee(id, name, salary, age);  if(result)  Console.WriteLine("Employee Details has been Saved Successfully");  else  Console.WriteLine("Some Error Occured");  }  public static void SearchEmployeeById()  {  int id;  Console.WriteLine("Enter Employee ID: ");  id = Convert.ToInt32(Console.ReadLine());  var result = EmployeeBLL.GetEmployeeById(id);  if(result.Count == 0)  Console.WriteLine("No Records Found");  else  {  result.ForEach(x => Console.WriteLine(x));  }  }  public static void SearchEmployeeByName()  {  string name;  Console.WriteLine("Enter Employee Name: ");  name = Console.ReadLine();  var result = EmployeeBLL.GetEmployeeByName(name);  if (result.Count == 0)  Console.WriteLine("No Records Found");  else  {  result.ForEach(x => Console.WriteLine(x));  }  }  public static void DisplayAllEmployees()  {  var result = EmployeeBLL.GetAllEmployee();  result.ToList().ForEach(x => Console.WriteLine(x));  }  }  } |

|  |
| --- |
| Output : |
|  |