DAY 22 ASSIGNMENTS PRESENTED

BY

POTUKANUMA JEEVITHA

22-02-2022

|  |
| --- |
| 1. **Employee Management Application**   **Add employee**  **Search employee [using id, name]**  **Display all employee**  **Employee id, employee name, employee salary, employee age using layers. (UI, BLL, DAL)**  **Employees.txt ( to be save in file)** |
| **Data Access Layer:** |
| using System;  using System.Collections.Generic;  using System.IO;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  //Author: JEEVITHA  //Purpose: To create Employee Management Application using DAL  //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  namespace DataAcessLibrary  {  public static class EmployeeDAL  {  public static string filePath = "F:\\EmployeeData\\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> employeeFound = new List<string>();  foreach (string employee in allEmployees)  {  var empDetails = employee.Split(',');  if (Convert.ToInt32(empDetails[0]) == Id)  {  isFound = true;  employeeFound.Add(employee);  break;  }  }  return employeeFound;  }  public static List<string> GetEmployeeByName(string Name)  {  var allEmployees = File.ReadAllLines(filePath);  bool isFound = false;  List<string> employeeFound = new List<string>();  foreach (string employee in allEmployees)  {  var empDetails = employee.Split(',');  if (empDetails[1].Contains(Name))  {  employeeFound.Add(employee);  }  }  return employeeFound;  }  public static string [] GetAllEmployees()  {  var allEmployees=File.ReadAllLines(filePath);  return allEmployees;  }  }  } |
| **Business Logic Layer:** |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  using DataAcessLibrary;  //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  //Author: JEEVITHA  //Purpose: To Create Employee Management Application by using BLL  //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  namespace BusinessLogicLibrary  {  public class EmployeeBLL  {  public static bool AddEmployee(int empId, string empName, int empSalary, int empAge)  {  // if all validations are succeccful then call DAL  var result = EmployeeDAL.AddEmployee(empId, empName, empSalary, empAge);  return result;  }  public static List<string> GetEmployeeById(int Id)  {  var result= EmployeeDAL.GetEmployeeById(Id);  return result;  }  public static List<string> GetEmployeeByName(string Name)  {  var result = EmployeeDAL.GetEmployeeByName(Name);  return result;  }  public static string[] GetAllEmployees()  {  var result = EmployeeDAL.GetAllEmployees();  return result;  }  }  } |
| **UI Layer (ClientApp):** |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  using BusinessLogicLibrary;  //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  //Author: JEEVITHA  //Purpose: To create Employee Management Application by using ClientApp  //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  namespace JeevithaClientApp  {  internal class Program  {  static void Main(string[] args)  {  int ch;  string choice;  do  {  Console.WriteLine("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");  Console.WriteLine("Employee Management Application");  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 Employees:");  Console.WriteLine(" Enter Your Choice:");  ch =Convert.ToInt32(Console.ReadLine());  switch (ch)  {  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 want to Continue (y/n):");  choice = Console.ReadLine();  } while (choice.Equals("y"));  }  // Add employees details  public static void AddEmployee()  {  int id, salary, age;  string name;  Console.WriteLine("Enter id:");  id=Convert.ToInt32(Console.ReadLine());  Console.WriteLine("Enter salary:");  salary = Convert.ToInt32(Console.ReadLine());  Console.WriteLine("Enter age:");  age = Convert.ToInt32(Console.ReadLine());  Console.WriteLine("Enter name:");  name = Console.ReadLine();  // call BLL Method  var result = EmployeeBLL.AddEmployee(id, name, salary, age);  if (result)  Console.WriteLine("Employee Details Saved Successfully");  else  Console.WriteLine("some error occured");  }  // search employees by using ID  public static void SearchEmployeeById()  {  int id;  Console.WriteLine("Enter id:");  id = Convert.ToInt32(Console.ReadLine());  var result = EmployeeBLL.GetEmployeeById(id);  if (result.Count == 0)  Console.WriteLine("No records exists with this id");  else  {  result.ForEach(p => Console.WriteLine(p));  }  }  // search employee details by using name  public static void SearchEmployeeByName()  {  string name;  Console.WriteLine("Enter name:");  name=Console.ReadLine();  var result=EmployeeBLL.GetEmployeeByName(name);  if (result.Count == 0)  Console.WriteLine("No records exists with this name");  else  {  result.ForEach(p => Console.WriteLine(p));  }  }  // display all employees details  public static void DisplayAllEmployees()  {  var result = EmployeeBLL.GetAllEmployees();  result.ToList().ForEach(p => Console.WriteLine(p));  }  }  } |
| **Output:** |
| **SOLUTION EXPLORER:** |
| **ADD EMPLOYEE DETAILS:** |
| **SEARCH EMPLOYEE DETAILS USING ID:** |
| **SEARCH EMPLOYEE DETAILS USING NAME:** |
| **DISPLAY ALL EMPLOYEES DETAILS:** |