|  |
| --- |
| **Day 22 Assignment**  **(22-02-2022)**  **By**  **Ram Charan** |

|  |
| --- |
| **Q) creating project for Employee management application** |
| **Code:** |
| **DAL Code:**  using System;  using System.Collections.Generic;  using System.IO;  using System.Linq;  using System.Text;  using System.Threading.Tasks;    namespace DataAccessLibrary  {    //Author: Ram charan  //Purpose: Creating app for Employee management  public static class EmployeeDAL  {  public static string filePath = "C:\\Users\\admin\\source\\repos\\C#Project\\Employee.txt";  /// <summary>  /// This method is to AddEmployee details  /// </summary>  /// <param name="eid"></param>  /// <param name="ename"></param>  /// <param name="esalary"></param>  /// <param name="eage"></param>  /// <returns>true or false</returns>  public static bool AddEmployee(int eid,string ename,int esalary,int eage)  {  //Write Code to append data  try  {  String textContent=String.Concat(eid,",",ename,",",esalary,",",eage);  File.AppendAllText(filePath, textContent+Environment.NewLine);  return true;  }  catch (Exception ex)  {  return false;  }    }  /// <summary>  /// This method is used to search employees by id  /// </summary>  /// <param name="id"></param>  /// <returns></returns>  public static List<string> GetEmpById(int id)  {  var allEmployees=File.ReadAllLines(filePath);  bool isFound=false;  List<string> employeeFound = new List<string>();  foreach(string employee in allEmployees)  {  var employeeDetails=employee.Split(',');  if(Convert.ToInt32(employeeDetails[0])==id)  {  isFound = true;  employeeFound.Add(employee);  break;  }  }  return employeeFound;  }  /// <summary>  /// This method is used to search by name  /// </summary>  /// <param name="id"></param>  /// <returns></returns>  public static List<string> GetEmpByName(string name)  {  var allEmployees = File.ReadAllLines(filePath);  List<string> employeeFound = new List<string>();  foreach (string employee in allEmployees)  {  var employeeDetails = employee.Split(',');  if ((employeeDetails[1]).Contains(name))  {  employeeFound.Add(employee);  break;  }  }  return employeeFound;  }  /// <summary>  /// This method is used to Display all Employees  /// </summary>  /// <returns></returns>  public static string[] DisplayAllEmployees()  {  var allEmployees = File.ReadAllLines(filePath);  return allEmployees;  }  }  }  **BLL Code:**  using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  using DataAccessLibrary;    namespace BusinessLogicLibrary  {  public static class EmployeeBLL  {  public static bool AddEmployee(int eid,string ename,int esalary,int age)  {  //To do      //all Success then call DAL  var result=EmployeeDAL.AddEmployee(eid,ename,esalary,age);  return result;  }  public static List<string> GetEmpById(int id)  {  var result=EmployeeDAL.GetEmpById(id);  return result;  }  public static List<string> GetEmpByName(string name)  {  var result= EmployeeDAL.GetEmpByName(name);  return result;  }  public static string[] DisplayAllEmployees()  {  var result=EmployeeDAL.DisplayAllEmployees();  return result;  }      }  }  **ClientApp:**  using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  using BusinessLogicLibrary;    namespace MyClientApp  {  public class Program  {  public static void AddEmployee()  {  //user input  int id, salary, age;  string name;  Console.WriteLine("Enter Id:");  id = Convert.ToInt32(Console.ReadLine());  Console.WriteLine("Enter Name");  name = Console.ReadLine();  Console.WriteLine("Enter Salary");  salary = Convert.ToInt32(Console.ReadLine());  Console.WriteLine("Enter age");  age = Convert.ToInt32(Console.ReadLine());  //call BLL  var result = EmployeeBLL.AddEmployee(id, name, salary, age);  if(result)  Console.WriteLine("Employee Details Saved");  else  Console.WriteLine("Error Occured");  }  public static void GetEmpById()  {  //user input  int id;  Console.WriteLine("Enter id:");  id=Convert.ToInt32(Console.ReadLine());  //call BLL  var result=EmployeeBLL.GetEmpById(id);  if(result.Count==0)  Console.WriteLine("No Records Found");  else  result.ForEach(d=>Console.WriteLine(d));  }  public static void GetEmpByName()  {  //user input  string name;  Console.WriteLine("Enter name");  name=Console.ReadLine();  //call BLL  var result=EmployeeBLL.GetEmpByName(name);  if(result!=null )  result.ForEach(d => Console.WriteLine(d));  else  Console.WriteLine("No Data Found");  }  public static void DisplayAllEmployees()  {  var result=EmployeeBLL.DisplayAllEmployees();  result.ToList().ForEach(d => Console.WriteLine(d));  }  static void Main(string[] args)  {  int ch;  string choice;  do  {  Console.WriteLine("Employee Management");  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("\nEnter your Choice:");  ch= Convert.ToInt32(Console.ReadLine());  switch(ch)  {  case 1:AddEmployee();  break;  case 2:GetEmpById();  break;  case 3:GetEmpByName();  break;  case 4:DisplayAllEmployees();  break;  default:  Console.WriteLine("Invalid input");  break;    }  Console.WriteLine("Do you want to continue (y/n)");  choice = Console.ReadLine();  }while(choice.Equals("y"));  }  }  } |
| **Output:** |
|  |

|  |
| --- |
| END OF THE DAY |