**Task # 01:**

**Create Employee CRUD app in windows form application c#**

public class Employees

{

private static string myConn = @"Data Source=DESKTOP-7BNIILF\SQLEXPRESS;Initial Catalog=final;Integrated Security=True";

public int EmpId { get; set; }

public string EmpName { get; set; }

public string Job { get; set; }

public string Date { get; set; }

public string Address { get; set; }

public string ContactNo { get; set; }

private const string SelectQuery = "Select \* from Employees";

private const string InsertQuery = "Insert Into Employees(Name, Job\_Status, HireDate,Address,Contact) Values (@EmpName, @Job, @Date, @Address,@ContactNo)";

private const string UpdateQuery = "Update Employees Set Name=@EmpName,Job\_Status=@Job,HireDate=@Date,Address=@Address,Contact=@ContactNo where EmployeeID =@EmpId";

private const string DeleteQuery = "Delete from Employees where EmployeeID=@EmpId";

private const string Id = "Select \* from Employees where EmployeeID=@EmpId";

public DataTable GetEmployees()

{

var datatable = new DataTable();

using (SqlConnection con = new SqlConnection(myConn))

{

con.Open();

using (SqlCommand com = new SqlCommand(SelectQuery, con))

{

using (SqlDataAdapter adapter = new SqlDataAdapter(com))

{

adapter.Fill(datatable);

}

}

}

return datatable;

}

public DataTable GetEmployeeswithid(string empid)

{

var datatable = new DataTable();

using (SqlConnection con = new SqlConnection(myConn))

{

string Id = "Select \* from Employees where EmployeeID='"+empid+"'";

con.Open();

using (SqlCommand com = new SqlCommand(Id, con))

{

using (SqlDataAdapter adapter = new SqlDataAdapter(com))

{

adapter.Fill(datatable);

}

}

}

return datatable;

}

public bool InsertEmployee(Employees employee)

{

int rows;

using (SqlConnection con = new SqlConnection(myConn))

{

con.Open();

using (SqlCommand com = new SqlCommand(InsertQuery, con))

{

// com.Parameters.AddWithValue("@EmployeeID", employee.EmpId);

com.Parameters.AddWithValue("@EmpName", employee.EmpName);

com.Parameters.AddWithValue("@Job", employee.Job);

com.Parameters.AddWithValue("@Date", employee.Date);

com.Parameters.AddWithValue("@Address", employee.Address);

com.Parameters.AddWithValue("@ContactNo", employee.ContactNo);

rows = com.ExecuteNonQuery();

}

}

return (rows > 0) ? true : false;

}

public bool UpdateEmployee(Employees employee)

{

int rows;

using (SqlConnection con = new SqlConnection(myConn))

{

con.Open();

using (SqlCommand com = new SqlCommand(UpdateQuery, con))

{

com.Parameters.AddWithValue("@EmpName", employee.EmpName);

com.Parameters.AddWithValue("@Job", employee.Job);

com.Parameters.AddWithValue("@Date", employee.Date);

com.Parameters.AddWithValue("@Address", employee.Address);

com.Parameters.AddWithValue("@ContactNo", employee.ContactNo);

com.Parameters.AddWithValue("@EmpId", employee.EmpId);

rows = com.ExecuteNonQuery();

}

}

return (rows > 0) ? true : false;

}

public bool DeleteEmployee(Employees employee)

{

int rows;

using (SqlConnection con = new SqlConnection(myConn))

{

con.Open();

using (SqlCommand com = new SqlCommand(DeleteQuery, con))

{

com.Parameters.AddWithValue("@EmpId", employee.EmpId);

rows = com.ExecuteNonQuery();

}

}

return (rows > 0) ? true : false;

}

}

}

