

```

using Entities;
using System;
using System.Collections.Generic;
using System.Data;
using System.Data.SqlClient;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace AccesoDatos
{
    public class DataPersonas : DataConnection
    {
        public int AltaPersona(Personas personas)
        {
            int resultado = -1;
            string query = @"Insert into Personas (Nombre,
                                                    Apellido,
                                                    Tipo_Documento_ID,
                                                    Nro_documento,
                                                    Tipo_Sexo_ID,
                                                    Nro_Telefono,
                                                    Nro_Alternativo,
                                                    Mail,
                                                    Observaciones,
                                                    Fecha_Alta)

                                                    values (@Nombre,
                                                            @Apellido,
                                                            @Tipo_Documento_ID,
                                                            @Nro_documento,
                                                            @Tipo_Sexo_ID,
                                                            @Nro_Telefono,
                                                            @Nro_Alternativo,
                                                            @Mail,
                                                            @Observaciones,
                                                            @Fecha_Alta)"

            ;
            SqlParameter nombre = new SqlParameter("@Nombre", personas.Nombre);
            SqlParameter apellido = new SqlParameter("@Apellido", personas.Apellido);
            SqlParameter tipo_Documento_ID = new SqlParameter("@Tipo_Documento_ID",
personas.Tipo_Documento_ID);
            SqlParameter nro_documento = new SqlParameter("@Nro_documento",
personas.Nro_documento);
            SqlParameter tipo_Sexo_ID = new SqlParameter("@Tipo_Sexo_ID",
personas.Tipo_Sexo_ID);
            SqlParameter nro_Telefono = new SqlParameter("@Nro_Telefono",
personas.Nro_Telefono);
            SqlParameter nro_Alternativo = new SqlParameter("@Nro_Alternativo",
personas.Nro_Alternativo);
            SqlParameter mail = new SqlParameter("@Mail", personas.Mail);
            SqlParameter observaciones = new SqlParameter("@Observaciones",
personas.Observaciones);
            SqlParameter fecha_Alta = new SqlParameter("@Fecha_Alta", personas.Fecha_Alta);

            SqlCommand cmd = new SqlCommand(query, conexion);
            cmd.Parameters.Add(nombre);
            cmd.Parameters.Add(apellido);
            cmd.Parameters.Add(tipo_Documento_ID);
            cmd.Parameters.Add(nro_documento);
            cmd.Parameters.Add(tipo_Sexo_ID);
            cmd.Parameters.Add(nro_Telefono);

```

```

cmd.Parameters.Add(nro_Alternativo);
cmd.Parameters.Add(mail);
cmd.Parameters.Add(observaciones);
cmd.Parameters.Add(fecha_Alta);

try
{
    OpenConnection();
    resultado = cmd.ExecuteNonQuery();
}
catch (Exception e)
{
    throw new Exception("Error al tratar de registrar nueva Persona", e);
}
finally
{
    CloseConnection();
    cmd.Dispose();
}

return resultado;
}
public int AltaJefe(Personas personas)
{
    int resultado = -1;
    string query = @"Insert into Personas (Nombre,
                                           Apellido,
                                           Tipo_Documento_ID,
                                           Nro_documento,
                                           Tipo_Sexo_ID,
                                           Nro_Telefono,
                                           Mail,
                                           Fecha_Alta)

                                           values (@Nombre,
                                           @Apellido,
                                           @Tipo_Documento_ID,
                                           @Nro_documento,
                                           @Tipo_Sexo_ID,
                                           @Nro_Telefono,
                                           @Mail,
                                           @Fecha_Alta)"
    ;
    SqlParameter nombre = new SqlParameter("@Nombre", personas.Nombre);
    SqlParameter apellido = new SqlParameter("@Apellido", personas.Apellido);
    SqlParameter tipo_Documento_ID = new SqlParameter("@Tipo_Documento_ID",
personas.Tipo_Documento_ID);
    SqlParameter nro_documento = new SqlParameter("@Nro_documento",
personas.Nro_documento);
    SqlParameter tipo_Sexo_ID = new SqlParameter("@Tipo_Sexo_ID",
personas.Tipo_Sexo_ID);
    SqlParameter nro_Telefono = new SqlParameter("@Nro_Telefono",
personas.Nro_Telefono);
    SqlParameter mail = new SqlParameter("@Mail", personas.Mail);
    SqlParameter fecha_Alta = new SqlParameter("@Fecha_Alta", personas.Fecha_Alta);

    SqlCommand cmd = new SqlCommand(query, conexion);
    cmd.Parameters.Add(nombre);
    cmd.Parameters.Add(apellido);
    cmd.Parameters.Add(tipo_Documento_ID);
    cmd.Parameters.Add(nro_documento);
    cmd.Parameters.Add(tipo_Sexo_ID);

```

```

cmd.Parameters.Add(nro_Telefono);
cmd.Parameters.Add(mail);
cmd.Parameters.Add(fecha_Alta);

try
{
    OpenConnection();
    resultado = cmd.ExecuteNonQuery();
}
catch (Exception e)
{
    throw new Exception("Error al tratar de registrar nueva Persona", e);
}
finally
{
    CloseConnection();
    cmd.Dispose();
}

return resultado;
}
public int EditarPersona(Personas personas)
{
    int resultado = -1;
    string orden = @"update Personas set Nombre = @Nombre,
                                Apellido = @Apellido,
                                Tipo_Documento_ID = @Tipo_Documento_ID,
                                Nro_documento = @Nro_documento,
                                Tipo_Sexo_ID = @Tipo_Sexo_ID,
                                Nro_Telefono = @Nro_Telefono,
                                Nro_Alternativo = @Nro_Alternativo,
                                Mail = @Mail,
                                Observaciones = @Observaciones
                                Where Persona_ID = @persona_ID"
    ;

    SqlParameter id = new SqlParameter("@persona_ID", personas.Persona_ID);
    SqlParameter nombre = new SqlParameter("@Nombre", personas.Nombre);
    SqlParameter apellido = new SqlParameter("@Apellido", personas.Apellido);
    SqlParameter tipo_Documento_ID = new SqlParameter("@Tipo_Documento_ID",
personas.Tipo_Documento_ID);
    SqlParameter nro_documento = new SqlParameter("@Nro_documento",
personas.Nro_documento);
    SqlParameter tipo_Sexo_ID = new SqlParameter("@Tipo_Sexo_ID",
personas.Tipo_Sexo_ID);
    SqlParameter nro_Telefono = new SqlParameter("@Nro_Telefono",
personas.Nro_Telefono);
    SqlParameter nro_Alternativo = new SqlParameter("@Nro_Alternativo",
personas.Nro_Alternativo);
    SqlParameter mail = new SqlParameter("@Mail", personas.Mail);
    SqlParameter observaciones = new SqlParameter("@Observaciones",
personas.Observaciones);

    SqlCommand cmd = new SqlCommand(orden, conexion);

    cmd.Parameters.Add(id);
    cmd.Parameters.Add(nombre);
    cmd.Parameters.Add(apellido);
    cmd.Parameters.Add(tipo_Documento_ID);
    cmd.Parameters.Add(nro_documento);
    cmd.Parameters.Add(tipo_Sexo_ID);
    cmd.Parameters.Add(nro_Telefono);

```

```

cmd.Parameters.Add(nro_Alternativo);
cmd.Parameters.Add(mail);
cmd.Parameters.Add(observaciones);

try
{
    OpenConnection();
    resultado = cmd.ExecuteNonQuery();
}
catch (Exception e)
{
    throw new Exception("Error al tratar de actualizar datos de la persona", e);
}
finally
{
    CloseConnection();
    cmd.Dispose();
}
return resultado;
}
public Personas Get_Las_Id_Personas(Personas personas)
{
    string query = @"sp_cargar_ultimo_ID_Personas";
    SqlCommand cmd = new SqlCommand(query, conexion);

    try
    {
        OpenConnection();
        SqlDataReader reader = cmd.ExecuteReader();

        if (reader.Read())
        {
            personas.Persona_ID = int.Parse(reader["Persona_ID"].ToString());
        }
        reader.Close();
        cmd.ExecuteNonQuery();
    }
    catch (Exception e)
    {
        throw new Exception("No se pudo extraer el último ID registrado de Personas",
e);
    }
    finally
    {
        CloseConnection();
        cmd.Dispose();
    }

    return personas;
}

#region Consulta cliente único
public Personas GetPersonaUnica(Personas personas)
{
    string query = @"select *
                    From Personas
                    where Persona_ID = @Persona_ID"
;

    SqlParameter persona_ID = new SqlParameter("@Persona_ID", personas.Persona_ID);

```

```
SqlCommand cmd = new SqlCommand(query, conexion);
```

```
cmd.Parameters.Add(persona_ID);
```

```
try
```

```
{
```

```
    OpenConnection();
```

```
    SqlDataReader reader = cmd.ExecuteReader();
```

```
    if (reader.Read())
```

```
    {
```

```
        personas.Persona_ID = int.Parse(reader["Persona_ID"].ToString());
```

```
        personas.Nombre = reader["Nombre"].ToString();
```

```
        personas.Apellido = reader["Apellido"].ToString();
```

```
        personas.Tipo_Documento_ID =
```

```
int.Parse(reader["Tipo_Documento_ID"].ToString());
```

```
        personas.Nro_documento = reader["Nro_documento"].ToString();
```

```
        personas.Tipo_Sexo_ID = int.Parse(reader["Tipo_Sexo_ID"].ToString());
```

```
        personas.Nro_Telefono = reader["Nro_Telefono"].ToString();
```

```
        personas.Nro_Alternativo = reader["Nro_Alternativo"].ToString();
```

```
        personas.Mail = reader["Mail"].ToString();
```

```
        personas.Observaciones = reader["Observaciones"].ToString();
```

```
    }
```

```
    reader.Close();
```

```
    cmd.ExecuteNonQuery();
```

```
}
```

```
catch (Exception e)
```

```
{
```

```
    throw new Exception("No se pudo realizar el cálculo requerido", e);
```

```
}
```

```
finally
```

```
{
```

```
    CloseConnection();
```

```
    cmd.Dispose();
```

```
}
```

```
return personas;
```

```
}
```

```
public Personas GetPersona(Personas personas)
```

```
{
```

```
    string query = @"select Nombre, Persona_ID
```

```
        From Personas
```

```
        where Personas.Persona_ID = @Persona_ID"
```

```
;
```

```
    SqlParameter persona_Id = new SqlParameter("@Persona_ID", personas.Persona_ID);
```

```
    SqlCommand cmd = new SqlCommand(query, conexion);
```

```
    cmd.Parameters.Add(persona_Id);
```

```
try
```

```
{
```

```
    OpenConnection();
```

```
    SqlDataReader reader = cmd.ExecuteReader();
```

```
    if (reader.Read())
```

```
    {
```

```
        personas.Nombre = reader["Nombre"].ToString();
```

```
        personas.Persona_ID = int.Parse(reader["Persona_ID"].ToString());
```

```
    }
```

```
    reader.Close();
```

```
    cmd.ExecuteNonQuery();
```

```
}
```

```

        catch (Exception e)
        {
            throw new Exception("No se pudo realizar el cálculo requerido", e);
        }
        finally
        {
            CloseConnection();
            cmd.Dispose();
        }

        return personas;
    }
}
public Personas BuscarCoincidencias(int Persona_ID, string Nro_Documento, Personas
personas)
{
    string query = @"select Persona_ID, Nro_documento from Personas
                    where Persona_ID = @Persona_ID and Nro_Documento =
@Nro_Documento";

    SqlParameter persona_ID = new SqlParameter("@Persona_ID", Persona_ID);
    SqlParameter nro_Documento = new SqlParameter("@Nro_Documento", Nro_Documento);

    SqlCommand cmd = new SqlCommand(query, conexion);

    cmd.Parameters.Add(persona_ID);
    cmd.Parameters.Add(nro_Documento);

    try
    {
        OpenConnection();
        SqlDataReader reader = cmd.ExecuteReader();

        if (reader.Read())
        {
            personas.Persona_ID = int.Parse(reader["Persona_ID"].ToString());
            personas.Nro_documento = reader["Nro_documento"].ToString();
        }
        reader.Close();
        cmd.ExecuteNonQuery();
    }
    catch (Exception)
    {
        throw;
    }
    finally
    {
        CloseConnection();
        cmd.Dispose();
    }

    return personas;
}

#endregion

#region Persona para mostrar plan asignado que tiene (si es que tiene)
public DataSet GetPersonaPlan(string buscar, Personas personas)
{
    string query = @"SELECT Personas.Persona_ID, Clientes.Cliente_ID, Nombre,
Apellido, Nro_Documento, Nro_Telefono, Mail

```

```
FROM Personas
```

```
inner join Clientes
on Clientes.Persona_ID =
```

```
Personas.Persona_ID
```

```
where Nro_Documento = @Documento"
```

```
;
```

```
SqlParameter documento = new SqlParameter("@Documento", buscar);
```

```
SqlCommand cmd = new SqlCommand(query, conexion);
```

```
cmd.Parameters.Add(documento);
```

```
DataSet ds = new DataSet();
```

```
SqlDataAdapter da = new SqlDataAdapter();
```

```
try
```

```
{
```

```
    OpenConnection();
```

```
    cmd.ExecuteNonQuery();
```

```
    da.SelectCommand = cmd;
```

```
    da.Fill(ds);
```

```
}
```

```
catch (Exception e)
```

```
{
```

```
    throw new Exception("Error al traer datos de cliente", e);
```

```
}
```

```
finally
```

```
{
```

```
    CloseConnection();
```

```
    cmd.Dispose();
```

```
}
```

```
return ds;
```

```
}
```

```
#endregion
```

```
#region Consulta Profesores
```

```
public DataTable BuscaProfesores(Personas personas)
```

```
{
```

```
    string query = @"sp_cargar_Profesores_Actuales";
```

```
    SqlCommand cmd = new SqlCommand(query, conexion);
```

```
    DataTable dt = new DataTable();
```

```
    SqlDataAdapter da = new SqlDataAdapter();
```

```
try
```

```
{
```

```
    OpenConnection();
```

```
    cmd.ExecuteNonQuery();
```

```
    da.SelectCommand = cmd;
```

```
    da.Fill(dt);
```

```
}
```

```
catch (Exception)
```

```
{
```

```
    throw;
```

```
}
```

```
finally
```

```
{
```

```
    CloseConnection();
```

```
        cmd.Dispose();
    }
    return dt;
}

#endregion
}
```