

Taller de Computación IV

Ejercicio 2

Duración: tres clases.

Objetivo: crear las clases necesarias para interactuar con la base de datos utilizando ODBC. Capturar los posibles errores y lanzarlos hacia la capa superior. Crear un formulario de pruebas que permita testear los distintos métodos de acceso a los datos.

Conceptos aplicados:

- Namespace o espacio de nombre
- Creación de clases
- Clase entidad
- Clase de acceso a datos ODBC con los objetos OdbcConnection, OdbcCommand y OdbcDataReader
- Control de errores de ejecución con la estructura Try...Catch...Finally
- Desarrollo orientado a Interfaces

Recuerde! para ver ejemplos de este ejercicio y los que siguen, bajen del aula virtual el link **T4 Ejemplos**.



Tenga en cuenta TODO ESTO, antes de comenzar!

- Utilice la clase **daConexion** para obtener una conexión a la base de datos.
- **Primero analice y comprenda la lógica de los métodos Insertar y BuscarPorClavePrimaria de la clase daEmpleado y pruebe éstos métodos ejecutando el proyecto y luego usando el debugger.**
Luego complete los métodos restantes: Actualizar, Eliminar y Buscar. Para incluir esta clase, haga clic con el botón derecho sobre el proyecto ControlEmpleadosDataODBC y seleccione la opción Add Existing Item y agregue el archivo daEmpleado.cs.
- La base de datos se encuentra ControlEmpleadosLib\Base.

- A partir de ahora y hasta el fin de cuatrimestre, trabaje con el proyecto guardado en una carpeta local desde su máquina. **No use su carpeta de la red de Ort.**

Parte 1:

1. **Lógica de negocio:** Completar la clase entidad **EmpleadoEntity** que se encuentra en el proyecto ControlEmpleadosEntity y agregarle las siguientes properties (ver los campos de la tabla Empleado para conocer el tipo de dato de cada atributo):
 - a. Legajo
 - b. Apellido
 - c. Nombre
 - d. Telefono
 - e. Email
 - f. Sueldo (tenga en cuenta que el sueldo es un entero sin signo)
 - g. FechaNacimiento

Ejemplo de creación de una Property:

```
private string legajo;  
  
public string Legajo  
{  
    get { return legajo; }  
    set { legajo = value; }  
}
```

El tipo de dato de la propiedad fecha de nacimiento es de tipo DateTime.

Nota: para convertir un valor en DateTime o cualquier otra conversión utilice la clase Convert. Esta clase incluye varios métodos para conversión de datos.

Importante: recuerde siempre crear el constructor de la clase e inicializar las variables.

2. Agregar los siguientes métodos y programar la lógica necesaria:
 - a. TieneEmail: este método retorna true si el empleado tiene email y false para el caso contrario. Recuerde que este dato puede ser nulo en la base de datos.
 - b. TieneFechaNacimiento: este método retorna true si el empleado tiene fecha de nacimiento y false para el caso contrario. Recuerde que este dato puede ser nulo en la base de datos.
3. **Acceso a Datos:** para crear la conexión a la base de datos se debe configurar el DSN (Data Source Name).
 - a. En Inicio/Ejecutar escriba odbcad32 y presione enter (o desde el Panel de Control de Windows).
 - b. Seleccione la solapa "DSN de Usuario" y presione el botón "Agregar".
 - c. De la lista, seleccione la opción "Microsoft Access Driver (*.mdb)" y presione "Finalizar".
 - d. Complete "Nombre del origen de datos" con el nombre "ControlEmpleados".
 - e. Presione el botón "Seleccionar", ubique la base de datos "Taller4.mdb" en la carpeta ControlEmpleadosLib\Base\ y selecciónela haciendo doble clic sobre la misma.
 - f. Presione la opción "Aceptar" para finalizar la configuración.
4. Complete el string de conexión de la clase **daConexion**, método **GetOpenedConnection** para poder conectarse a la base de datos. En el string de conexión, debe hacer referencia al DSN creado en el punto 3.
5. Completar la clase de acceso a datos **daEmpleado** (está proyecto ControlEmpleadosDataODBC) con los siguientes métodos utilizando los ya implementados como ejemplo:

- Buscar: búsqueda por legajo, apellido y/o nombre. (tome como ejemplo BuscarPorClavePrimaria). **La búsqueda se hará por los tres filtros (son excluyentes), tenga en cuenta el uso de LIKE en la sentencia SQL y use el comodín “%” concatenado al valor al parámetro.** Por ejemplo:

```
parametro.valor = "%" + legajo + "%"
```

- Actualizar (tome como ejemplo Insertar y EjecutarComando)
 - Eliminar (tome como ejemplo Insertar y EjecutarComando)
6. Crear el WebForm “Test.aspx” en el proyecto ControlEmpleados y agregarle 5 botones. Cada botón deberá ejecutar un método de la clase daEmpleado. El objetivo de este formulario es probar que los métodos de la capa de datos funcionen correctamente. En el evento clic de cada botón crear los objetos necesarios y completarlos con datos manualmente (“hardcodeados”).
 7. Tenga en cuenta el uso de daEmpleado. Ejemplo:
daEmpleado empleado = new daEmpleado();

Parte 2:

8. Luego de completar **TODA** la parte 1, cree otro DSN llamado **ControlEmpleadosSqlSvr**, pruebe conectarse a la base de datos Sql Server del servidor con esta información de conexión:

Driver a usar: Sql Server
Nombre del servidor: a-srv-moe
Nombre de Base de datos: Taller4
Usuario: taller4
Password: taller4

Tenga en cuenta que esta base de datos la usan TODOS los alumnos de Taller4. No elimine información de otro grupo.

9. Modifique el string de conexión de la clase daConexion por:

Dsn=ControlEmpleadosSqlSvr;uid=taller4;pwd=taller4;db=Taller4

10. En este paso vamos a crear un stored procedure para la búsqueda por clave primaria.
 - a. Ejecute la aplicación **S:\Query Analyzer\isqlw.exe** para conectarse a la consola del SQL Server.
 - b. Complete los datos de conexión mencionados en el **punto 8** (seleccione **SQL Server Authentication**).
 - c. Para visualizar el **Object Browser** (barra ubicada a la izquierda que permite navegar por las bases de datos) presione **F8**.
 - d. De la lista desplegable que se encuentra en la barra debajo del menú, seleccione la base de datos **Taller4**.
 - e. Para crear el stored procedure utilice el siguiente ejemplo:

CREATE PROCEDURE Grupo01BuscarPorClavePrimaria

@Legajo varchar (10)

AS

SELECT * FROM Empleado WHERE EmpleadoLegajo = @Legajo

GO

Y luego presione **F5** para crearlo en la base de datos.

IMPORTANTE: el nombre del stored procedure debe ser Grupo + el número de grupo al que pertenece. Recuerde que la base de datos es compartida por todos los alumnos del curso.

- f. Para ejecutar el stored procedure desde la aplicación, modifique la constante **SQLSearchByPrimaryKey** que se encuentra en **daEmpleado** de la siguiente manera:

```
private const string SQLSearchByPrimaryKey = "EXEC  
Grupo01BuscarPorClavePrimaria ?";
```

- g. Ejecute la aplicación y pruebe el funcionamiento.