




Florida universitaria

EJERCICIOS LOWCODE PARTE 1

Profesor: Daniel Lutz

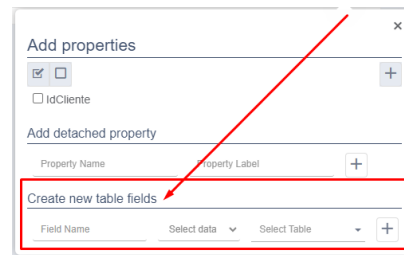
Ejercicio 1

Crear un listado de clientes

1. Crear objeto
 - Acceder al asistente de creación de objetos
 - Nombre de objeto: **Cliente**, descripción: **{{Nombre comercial}}**
 - Nombre de Colección: **Cientes**, descripción: **Cientes**
 - Icono: **client** 
 - Origen de datos: **desde una nueva tabla**
 - Cadena de conexión: main **data** configuration string
 - Nombre de la nueva tabla: **Cientes**
 - Clave primaria: **IdCliente**
2. Generar formulario de cliente añadiendo propiedades al objeto

No incluya espacios, tildes ni caracteres especiales al definir los nombres de los campos.

 - Añadir los siguientes campos **nuevos** a la tabla mediante el asistente (img. 1):
 - NombreComercial. Text (50), requerido
 - RazonSocial, Text (50), requerido
 - Direccion, text (1000)
 - CodPostal, text (50)
 - NIF, text (50)
 - Email, texto (100)
 - Tel, Text (50)
 - FechaFundacion, Date
 - FechaAlta, Datetime
 - IdEstado, Integer, requerido
 - IdEmpleadoAsignado, Integer
 - EnviarInfo, boolean
3. Definir diseño del formulario tal como muestra la imagen 2
 - Definir etiquetas e iconos para los controles.
 - Crear separadores, indicando icono y una clase entre las siguientes disponibles: txt-outstanding, txt-info, txt-notify, txt-tools
 - IdCliente (Código):
 - Tipo de campo: **Identificador**
 - Bloqueado: verdadero



Img. 1

Clientes

CLIENTES

Cliente

Código: #

Nombre: T Razón social: T

Dirección: Código postal: #

Teléfono: E-mail: @ NIF: #

Situación

Estado: Empleado asignado: Permitir enviar información: ☐

Fecha de alta: dd/mm/aaaa --:-- Antigüedad: dd/mm/aaaa

Save

Img. 2

4. Definir la lista en el apartado List setting.
 - Añadir los siguientes campos a la lista
 - Nombre
 - Razón social
 - Email
 - Teléfono

Clientes

CLIENTES

1

| | Nombre | Razón Social | E-Mail | Teléfono |
|--|---------|---------------------------|------------------|-----------|
| | Florida | Florida Universitaria S.A | info@florida.com | 963123456 |
| | Flexygo | Flexygo S.L | info@flexygo.com | 963365587 |

5. Definir acceso a la lista en el menú lateral
 - Ir al paso 7 del asistente y entrar en **show collection configuration**
 - Activar la opción “show in menú” y guardar los cambios.
 - Recargar caché.
 - Reiniciar la aplicación (botón recargar página del navegador o F5).
 - Acceder al listado de clientes desde el menú generado por el sistema.
6. Test
 - Dar de alta un cliente de pruebas: Código = 1, Empleado asignado = 1.
 - Comprobar que el cliente creado sale en la lista.

Ejercicio 2

Listado de empleados

7. Siguiendo el ejemplo del cliente, crear los empleados.
 - Definir objeto Empleado
 - Objeto: **Empleado**, Descripción: **{{Nombre}} {{Apellidos}}**
 - Colección: **Empleados**, Descripción: **Empleados**
 - Icono: **User-3**
 - Origen de datos: **Nueva tabla**
 - Cadena de conexión: **Main Data Configuration String**
 - Clave primaria: **IdEmpleado** (integer).
 - Crear nuevos campos en la tabla del objeto
 - Nombre, text (50)
 - Apellidos, text (50)
 - Dirección, text (1000)
 - TelefonoPersonal, text (50)
 - TelefonoEmpresa, text (50)
 - Email, text (50)
 - IdEstado, integer
 - FechaAlta, Date
 - FechaBaja, Date
 - Imagen, text (max)
 - Definir el diseño del formulario de edición de empleado según la Img.3
 - Definir el formulario de lista con los siguientes campos:
 - Código
 - Nombre
 - Apellidos
 - Email
 - Crear acceso a la lista desde el menú lateral: Habilitar en la configuración de la colección el menú de acceso a la lista
 - Testear alta de empleado

The screenshot shows a web application window titled 'Empleados'. Inside, there's a header 'EMPLEADOS' and a sub-header 'Empleado' with a user icon. The form contains several input fields: 'Código' (with a '#' icon), 'Nombre' (with a 'T' icon, value 'Luis Alberto'), 'Apellidos' (with a 'T' icon, value 'Suárez Díaz'), 'Tel. personal' (with a phone icon), 'Tel. empresa' (with a phone icon), 'E-mail' (with an '@' icon), 'Fecha de alta' (with a calendar icon, value 'dd/mm/aaaa'), 'Fecha de baja' (with a calendar icon, value 'dd/mm/aaaa'), 'Estado' (with a list icon), and 'Dirección' (with a location pin icon). A large 'Imagen' field is on the left. At the bottom left, there is a 'Save' button.

Img. 3

Ejercicio 3

Tablas complementarias en bases de datos

En este ejercicio crearemos tablas donde definir los estados de clientes y empleados, y lo haremos desde el Entorno de diseño de SQL Server

1. Estados de clientes
 - Ir al sql management studio y crear la tabla Clientes_Estados (**IdEstado** integer, Descrip varchar (50))
 - Crear los siguientes estados: 0-Activo, 1-Inactivo, 2-Cerrado, 3-Bloqueado
2. Estados de empleados
 - Ir al sql management studio y crear la tabla Empleados_Estados (**IdEstado** integer, Descrip varchar (50))
 - Crear los siguientes estados: 0-Activo, 1-Inactivo, 2-Baja médica, 3-Despedido
3. Definir integridad referencial
 - Definir clave externa de clientes para el campo IdEstado
 - Definir clave externa de empleados para el campo IdEstado

Ver código T-SQL al final del documento.

Ejercicio 4

Configuración avanzada de controles

1. **Formulario de cliente**
 - Configurar campo: IdCliente (código)
 - Tipo: Identifier
 - Bloqueado
 - Configurar campo: tel (teléfono)
 - Tipo: Phone
 - Configurar campo: Email
 - Tipo E-Mail
 - Configurar campo: IdEstado (Estado)
 - Tipo: Combo.
 - Valor por defecto: 0.
 - Sql= Select IdEstado, Descrip from Clientes_Estados order by Descrip
 - SQL Value field= IdEstado
 - SQL Display field= Descrip
 - Configurar campo: IdEmpleadoAsignado (Empleado asignado)
 - Tipo: DbCombo.
 - Sql: Select IdEmpleado, Nombre, Apellidos, IdEstado, Email from Empleados order by Nombre, Apellidos
 - SQL Value field: IdEmpleado
 - SQL Display field: Nombre

- Direct template: `<i class="flx-icon icon-man-4 icon-margin-right"></i>{{Nombre}} {{Apellidos}} <small> ({{Email|isnull:Sin email}}) </small>`
- Configurar campo: fecha de alta:
 - Valor por defecto: `{{currentDateTime}}`
 - Bloqueado
- Configurar campo: Envío de información
 - Default value: 1

2. Testear alta de cliente creando 3 nuevos clientes

3. Definir los controles en formulario de Empleado

- Configurar campo: IdEmpleado (Código)
 - Tipo: Identifier
 - Bloqueado
- Configurar campos teléfonos
 - Tipo: Phone
- Configurar campo: Email
 - Tipo: E-Mail
- Configurar campo desplegable de estado de empleado (IdEstado)
 - Tipo: Combo.
 - Default value: 0.
 - Sql: `Select IdEstado, Descrip from Empleados_Estados order by Descrip`
 - SQL Value field: IdEstado
 - SQL Display field: Descrip
- Configurar campo: fecha de alta:
 - Default value: `{{currentDateTime}}`
 - Bloqueado
- Configurar campo: Imagen
 - Tipo: Image embeded
 - Tipo de compresión: media

Ayuda T-SQL

Puede ejecutar el script completo desde una **nueva ventana de consulta** contra la base de datos donde ese encuentran creadas las tablas de Clientes y Empleados

```
IF NOT EXISTS (Select 1 from sys.tables where name = 'Empleados_Estados')
CREATE TABLE [dbo].[Empleados_Estados](
    [IdEstado] [int] NOT NULL,
    [Descrip] [varchar](50) NOT NULL,
    CONSTRAINT [PK_Empleados_Estados] PRIMARY KEY CLUSTERED
(
    [IdEstado] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

GO

IF NOT EXISTS (Select 1 from sys.tables where name = 'Clientes_Estados')
CREATE TABLE [dbo].[Clientes_Estados](
    [IdEstado] [int] NOT NULL,
    [Descrip] [varchar](50) NOT NULL,
    CONSTRAINT [PK_Clientes_Estados] PRIMARY KEY CLUSTERED
(
    [IdEstado] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

GO

insert into Clientes_Estados (IdEstado, Descrip)
Values (0,'Activo'), (1,'Inactivo'), (2,'Cerrado definitivo'), (3,'Bloqueado')

insert into Empleados_Estados (IdEstado, Descrip)
Values (0,'Activo'), (1,'Inactivo'), (2,'Baja médica'), (3,'Despedido')

if not exists (select 1 from sys.foreign_keys where
parent_object_id=object_id('Clientes') and Name = 'FK_Clientes_Estados_IdEstado')
BEGIN
    ALTER TABLE Clientes
    ADD CONSTRAINT [FK_Clientes_Estados_IdEstado] FOREIGN KEY ([IdEstado])
REFERENCES [dbo].[Clientes_Estados] ([IdEstado]) ON UPDATE CASCADE
END

if not exists (select 1 from sys.foreign_keys where
parent_object_id=object_id('Empleados') and Name = 'FK_Empleados_Estados_IdEstado')
BEGIN
    ALTER TABLE Empleados
    ADD CONSTRAINT [FK_Empleados_Estados_IdEstado] FOREIGN KEY ([IdEstado])
REFERENCES [dbo].[Empleados_Estados] ([IdEstado]) ON UPDATE CASCADE
END
```