Paquetes del Proyecto

* Conector
* Conexión
* Controlador
* Img
* Modelo
* Pdf
* Vista
  + formularioAcceso.java

Formulario Acceso

Determinar cómo dimensión 700 x 500;

Configuración dentro del constructor

this.setResizable(false);

this.setLocationRelativeTo(null);

this.setTitle("Acceso al Sistema");

this.setSize(700, 500);

Diseño

Panel izquierdo – Rosa

Panel derecho - Blanco

Posible base de datos

¡Claro! Aquí tienes un ejemplo simplificado de cómo podrías generar una base de datos para un centro de enseñanza utilizando el lenguaje SQL y el modelo relacional:

1. Identificar las entidades:
   * Estudiantes
   * Profesores
   * Cursos
   * Materias
2. Definir las tablas:

sqlCopy code

-- Tabla Estudiantes CREATE TABLE Estudiantes ( ID INT PRIMARY KEY, Nombre VARCHAR(50), Apellido VARCHAR(50), FechaNacimiento DATE, Direccion VARCHAR(100) ); -- Tabla Profesores CREATE TABLE Profesores ( ID INT PRIMARY KEY, Nombre VARCHAR(50), Apellido VARCHAR(50), Especialidad VARCHAR(50) ); -- Tabla Cursos CREATE TABLE Cursos ( ID INT PRIMARY KEY, Nombre VARCHAR(50), Descripcion VARCHAR(100), ProfesorID INT, FOREIGN KEY (ProfesorID) REFERENCES Profesores(ID) ); -- Tabla Materias CREATE TABLE Materias ( ID INT PRIMARY KEY, Nombre VARCHAR(50), Descripcion VARCHAR(100) );

1. Establecer las relaciones:
   * La tabla **Cursos** tiene una clave externa **ProfesorID** que referencia el ID del profesor responsable del curso.
   * No se establecen relaciones específicas entre las demás tablas en este ejemplo.
2. Definir restricciones y validaciones:
   * Puedes agregar restricciones adicionales en las columnas, como NOT NULL o rangos específicos para los datos, según tus necesidades.
3. Normalizar la base de datos:
   * Asegúrate de seguir las reglas de normalización para evitar redundancias y problemas de actualización.
4. Crear las tablas:
   * Utiliza las sentencias CREATE TABLE para crear las tablas en tu sistema de gestión de bases de datos.
5. Realizar consultas y operaciones:
   * Utiliza sentencias SQL como INSERT, UPDATE, SELECT y DELETE para interactuar con la base de datos y realizar operaciones como insertar estudiantes, actualizar información, recuperar datos, etc.

Este es solo un ejemplo básico para ilustrar cómo podrías estructurar una base de datos para un centro de enseñanza. Recuerda adaptarlo según tus necesidades específicas y considerar otros aspectos importantes, como la seguridad de la base de datos y las consultas necesarias para obtener información relevante para tu centro de enseñanza.

Basándonos en la descripción proporcionada, podemos identificar las siguientes entidades y relaciones:

Entidades:

- Alumno (ID, Nombre, Apellidos, Dirección, Población, DNI, Fecha de nacimiento, Código postal, Teléfono)

- Profesor (ID, Nombre, Apellidos, Dirección, Población, DNI, Fecha de nacimiento, Código postal, Teléfono)

- Asignatura (ID, Código, Nombre, Número de horas semanales)

- Curso (ID, Código, Nombre)

- Aula (ID, Código, Piso, Número de pupitres)

- Tutoría (ID, ProfesorID, CursoID)

Relaciones:

- Matrícula (AlumnoID, AsignaturaID, Nota, Incidencias)

- Impartición (AsignaturaID, ProfesorID)

- AsignaturasAula (AsignaturaID, AulaID)

- Horario (AsignaturaID, AulaID, Fecha, Hora)

Teniendo en cuenta estas entidades y relaciones, podemos diseñar el esquema de la base de datos utilizando el lenguaje SQL. Aquí tienes un ejemplo:

```sql

-- Tabla Alumnos

CREATE TABLE Alumnos (

ID INT PRIMARY KEY,

Nombre VARCHAR(50),

Apellidos VARCHAR(50),

Dirección VARCHAR(100),

Población VARCHAR(50),

DNI VARCHAR(10),

FechaNacimiento DATE,

CódigoPostal VARCHAR(10),

Teléfono VARCHAR(20)

);

-- Tabla Profesores

CREATE TABLE Profesores (

ID INT PRIMARY KEY,

Nombre VARCHAR(50),

Apellidos VARCHAR(50),

Dirección VARCHAR(100),

Población VARCHAR(50),

DNI VARCHAR(10),

FechaNacimiento DATE,

CódigoPostal VARCHAR(10),

Teléfono VARCHAR(20)

);

-- Tabla Asignaturas

CREATE TABLE Asignaturas (

ID INT PRIMARY KEY,

Código VARCHAR(10),

Nombre VARCHAR(100),

HorasSemanales INT

);

-- Tabla Cursos

CREATE TABLE Cursos (

ID INT PRIMARY KEY,

Código VARCHAR(10),

Nombre VARCHAR(100)

);

-- Tabla Aulas

CREATE TABLE Aulas (

ID INT PRIMARY KEY,

Código VARCHAR(10),

Piso INT,

NúmeroPupitres INT

);

-- Tabla Tutorías

CREATE TABLE Tutorías (

ID INT PRIMARY KEY,

ProfesorID INT,

CursoID INT,

FOREIGN KEY (ProfesorID) REFERENCES Profesores(ID),

FOREIGN KEY (CursoID) REFERENCES Cursos(ID)

);

-- Tabla Matrícula

CREATE TABLE Matrícula (

AlumnoID INT,

AsignaturaID INT,

Nota DECIMAL(4, 2),

Incidencias VARCHAR(500),

FOREIGN KEY (AlumnoID) REFERENCES Alumnos(ID),

FOREIGN KEY (AsignaturaID) REFERENCES Asignaturas(ID)

);

-- Tabla Impartición

CREATE TABLE Impartición (

AsignaturaID INT,

ProfesorID INT,

FOREIGN KEY (AsignaturaID) REFERENCES Asignaturas(ID),

FOREIGN KEY (ProfesorID) REFERENCES Profesores(ID)

);

-- Tabla AsignaturasAula

CREATE TABLE AsignaturasAula (

AsignaturaID INT,

AulaID INT,

FOREIGN KEY (AsignaturaID) REFERENCES Asignaturas(ID),

FOREIGN KEY (AulaID) REFERENCES Aulas(ID