



Simulacro prueba de desempeño Java #2

Enunciado del Ejercicio

Desarrolla una aplicación en Java que utilice JDBC para interactuar con una base de datos MySQL, destinada a la gestión de un hospital. La aplicación deberá permitir el manejo de pacientes, médicos, citas y especialidades médicas. Cada cita estará asociada a un solo médico y un solo paciente, y cada médico tendrá una especialidad.

Modelo Entidad-Relación

Especialidad

- id_especialidad (Clave primaria, entero, autoincremental)
- nombre (Cadena)
- descripcion (Cadena)

Médico

- id_medico (Clave primaria, entero, autoincremental)
- nombre (Cadena)
- apellidos (Cadena)
- id_especialidad (Clave foránea, entero, referencia a Especialidad)



Paciente

- id_paciente (Clave primaria, entero, autoincremental)
- nombre (Cadena)
- apellidos (Cadena)
- fecha_nacimiento (Fecha)
- documento_identidad (Cadena)

Cita

- id_cita (Clave primaria, entero, autoincremental)
- id_paciente (Clave foránea, entero, referencia a Paciente)
- id_medico (Clave foránea, entero, referencia a Médico)
- fecha_cita (Fecha)
- hora_cita (Hora)
- motivo (Cadena)

Relaciones

- Una Especialidad puede ser ejercida por varios Médicos.
- Un Médico puede atender múltiples Citas.
- Un Paciente puede tener múltiples Citas.

Operaciones CRUD

Desarrolla operaciones CRUD para cada entidad (Especialidad, Médico, Paciente, Cita) utilizando JDBC:

- Create: Permite añadir nuevas especialidades, médicos, pacientes y citas a la base de datos.
- Read: Consulta registros existentes, por ejemplo, busca citas por fecha, pacientes por documento de identidad, o todos los médicos de una especialidad.



- Update: Actualiza registros, por ejemplo, cambia la fecha de una cita, los datos de un paciente, o la especialidad de un médico.
- Delete: Elimina registros, por ejemplo, elimina una especialidad, un médico, un paciente, o una cita.