

Taller POO Avanzado

Sistema básico de reservas

Eres la persona encargada de administrar un sistema simple de reservas para salas de reuniones. El objetivo es construir una aplicación en Java que permita crear, almacenar, listar y eliminar reservas, usando correctamente:

- Paquetes
- Modificadores de acceso
- Encapsulación (getters y setters)
- Colecciones de objetos
- Excepciones básicas
- Bloques try/catch

Como administrador del sistema, quiero gestionar reservas de salas (crear, listar y eliminar) para mantener un control sencillo de las reservas activas.

Requisitos

1. Crear una clase Reserva en un paquete llamado model.
 - Atributos privados:
id (int), cliente (String), fecha (String), duracionHoras (int)
 - Constructor que inicialice todos los atributos.
 - **Getters y setters** para todos los campos.
 - Método mostrarInfo() que imprima los datos de la reserva.
2. Crear una clase SistemaReservas en un paquete llamado service.
 - Debe contener una colección ArrayList<Reserva>.
 - Métodos:
 - agregarReserva(Reserva r)
 - eliminarReserva(int id)
 - listarReservas()
 - contarReservas()
 - Debe utilizar **getters** para acceder a los datos de las reservas.
3. Crear una clase Main en un paquete app.

- Crear al menos **dos reservas**.
- Agregarlas al sistema.
- Usar **al menos un setter** para modificar datos de una reserva.
- Listar las reservas.
- Mostrar el total de reservas.
- Eliminar una reserva por ID usando try/catch para capturar posibles errores.
- Volver a listar reservas.

4. Manejo de errores:

- Debes envolver la eliminación de reservas en un bloque try/catch.
- Si el ID no existe, el programa debe mostrar un mensaje claro sin detenerse.

5. La salida en consola debe ser clara y comprensible.

Entrega

Debes entregar en un repositorio de GitHub

1. El proyecto con **tres paquetes** (model, service, app).
2. Código que compile y ejecute sin errores.
3. Una captura de pantalla de la salida del programa mostrando:
 - Reservas creadas
 - Uso de setters
 - Eliminación
 - Conteo final