

**INSTRUCTIVO:** El proyecto será una aplicación en Java / MySQL. Se debe crear las tablas de la base de datos, y las correspondientes clases. Utilice una clase para la conexión y confeccione la interfaz gráfica (GUI). El grupo deberá presentar el proyecto en 3 etapas.

- 1- Armado de la base de datos completa, relaciones. Proyecto en Netbeans, y clase de conexión. ABM de la clase Huesped (HuespedData) pruebas desde el main.
- 2- Desarrollo de todas las clases principales del dominio (ABM), consultas SQL embebidas. Interfaces gráficas completas.
- 3- Implementación de todos los métodos necesarios, listados, ABM de clases relacionadas y aplicación terminada.

Se deben presentar las clases con atributos y métodos necesarios (abstracción). Setters y Getters (encapsulamiento). Constructor.

Se evaluará el aporte de todos los miembros al proyecto, en la construcción de las clases. Se subirán los commit al repositorio público (GitHub).

## HOTEL IDEAL

### Relaciones

- El *Hotel* tiene un conjunto fijo de **Habitaciones** numeradas del 1 al 200 en donde los huéspedes pueden alojarse. Se encuentran en un piso, y tienen un estado *Ocupada (1)* o *Libre (0)*.
- Una habitación es de una categoría o **Tipo de Habitación** específico. A este *Tipo de Habitación* (Estándar simple, doble, Triple, o Suite Lujo) pertenecen muchas habitaciones.
- Un **huésped** hace muchas **reservas**, y cada reserva corresponde a una habitación. Así, una reserva vincula un huésped con una habitación, y tiene fechas inicio y fin, días, monto a pagar, y un estado.

### Clases

- Los huéspedes son quienes podrán realizar sus reservas según disponibilidad en las fechas que correspondan. **ABM Huésped** tiene sus datos personales (nombre, DNI, Domicilio, correo, celular).
- **Crear la Clase Tipo de Habitación** tendrá sus propias características, las relevantes para nuestro sistema de gestión de reservas son: código, *cantidad de personas* (máxima), cantidad de camas, tipo de camas (Simples, Queen, King Size), precio por noche, que por simplicidad se supondrá único (sin promociones o tarifas distintas para temporada alta o baja).
- Un método *cambiarPrecio*, busca un Tipo de Habitación y actualiza su precio.
- **ABM de Habitaciones:** Se deben poder gestionar el alta baja y modificación de Habitaciones de un tipo. Ejemplo. Si antes tenía 30 de tipo estándar, luego de agregar las habitaciones 201 al 205 de ese tipo, el conteo sumará 35 en total.
- **ABM Reserva.** En cada *Reserva* se toma nota de: *datos del Huésped, Tipo de Habitación, cantidad de personas*, fechas de entrada y salida, importe total de estadía. Metodos Buscar una Reserva por Huésped, Cancelar reserva
- El proceso de hacer una *Reserva* se organizará. El método *Crear Reserva*:
  1. Se introduce por teclado *fechaEntrada* y *fechaSalida* y *cantidad de personas*.
  2. Se busca un “*tipo de habitación*” para esa cantidad de personas y el precio.
  3. Se devuelve una lista de las habitaciones que estén Libres, de ese “*Tipo Habitación*”.

**Calcular monto estadía:** en la reserva, se calcula como precio por noche (de Tipo Habitación) por cantidad de días de estadía ingresado. Devuelve el monto\*.

4. Se construye la reserva con *fechaCheckin*, *fechaCheckout*, *Huésped*, *Habitación*, *monto\**, *estado=1*.
5. La Habitación se marca *Ocupada(1)*, en la fecha de salida vuelve a su estado *Libre*.

- Método “*finReserva*” recibe un *Huésped*, permite buscar una reserva se marca, de *Activa(1)* a *Inactiva(0)*. Se busca la habitación y se marca *Libre(0)*.
- Búsqueda de Reservas por huésped o fechas: devuelve una Reserva.
- Mostrar Habitaciones clasificadas por Tipo de Habitación, y su estado actual (*Libre/Ocupada*).
- Un informe de Huéspedes por dni como campo de búsqueda.



