

## Trabajo Práctico 5

### Ejercicio: Estadías

## Esquema DB

ESTADIA<dniCliente, codHotel, cantidadHabitaciones, direccionHotel, ciudadHotel, dniGerente, nombreGerente, nombreCliente, ciudadCliente, fechaInicioHospedaje, cantDiasHospedaje, #Habitacion>

### Restricciones

- Existe un único gerente por hotel. Un gerente podría gerenciar más de un hotel.
- Un cliente puede realizar la estadía sobre más de una habitación del hotel en la misma fecha. Para cada habitación puede reservar diferentes cantidades de días.
- cantidadHabitaciones indica la cantidad de habitaciones existentes en un hotel.
- El código de hotel (codHotel) es único y no puede repetirse en diferentes ciudades.
- Un cliente puede realizar reservas en diferentes hoteles para la misma fecha.
- numHabitacion se puede repetir en distintos hoteles.
- En la misma direccionHotel de una ciudadHotel puede haber más de un hotel funcionando.

## 1. Dependencias Funcionales (DFs) a partir del esquema y las restricciones dadas

Primero modificamos #Habitacion por numHabitacion ya que una de las reglas para cumplir con la Forma Relacional y Primera Forma Normal es que los nombres de las columnas de una tabla deben ser únicos y sin tener, en lo posible, caracteres extraños como #, @, etc.

- **codHotel -> ciudadHotel, direccionHotel, cantidadHabitaciones, dniGerente, nombreGerente**  
El código del hotel es único y determina todos los atributos relacionados con el hotel: ciudad, dirección, cantidad de habitaciones existentes, DNI y nombre del gerente.
- **dniGerente -> nombreGerente**  
Un gerente tiene un único nombre así que nombreGerente depende funcionalmente de dniGerente.
- **dniCliente -> nombreCliente, ciudadCliente**  
El DNI del cliente es único y determina el nombre y la ciudad del cliente.
- **dniCliente, codHotel, fechaInicioHospedaje -> numHabitacion, cantDiasHospedaje**  
Un cliente puede realizar la estadía sobre más de una habitación del hotel en la misma fecha. Entonces la combinación de dniCliente, codHotel y fechaInicioHospedaje determina la habitación y cantidad de días de hospedaje.
- **codHotel, numHabitacion -> cantidadHabitaciones**  
Las habitaciones disponibles en un hotel están determinadas por el código de hotel y el número de habitación.
- **direccionHotel, ciudadHotel -> codHotel**  
En la misma direccionHotel de una ciudadHotel puede haber más de un hotel funcionando, pero combinando direccionHotel y ciudadHotel se determina el código de hotel único.

## 2. Determinación de Clave Candidata

La combinación de **dniCliente**, **codHotel**, **fechaInicioHospedaje**, **numHabitacion** identificarían de manera única una estadía, considerando las restricciones del sistema:

- **dniCliente** identifica al cliente que hace la estadía.
- **codHotel** especifica el hotel donde se hace la estadía.
- **fechaInicioHospedaje** asegura que la estadía está asociada a una fecha.
- **numHabitacion** distingue la habitación que el cliente reservó (ya que un cliente puede reservar varias habitaciones en el mismo hotel y en la misma fecha).

Por estos motivos la **clave candidata** es (**dniCliente**, **codHotel**, **fechaInicioHospedaje**, **numHabitacion**)

## 3. Normalización a Tercera Forma Normal (3FN)

La tabla original fue dividida en 4 tablas como se muestra a continuación, evitando redundancias y dando más consistencia a la base de datos:

### 1 - Tabla **ESTADIA**:

- **dniCliente** (Clave foránea: Referencia al cliente)
- **codHotel** (Clave foránea: Referencia al hotel)
- **fechaInicioHospedaje**
- **numHabitacion**
- **cantDiasHospedaje**
- Clave primaria compuesta (**dniCliente**, **codHotel**, **fechaInicioHospedaje**, **numHabitacion**)

### 2 - Tabla **HOTEL**

- **codHotel** (Clave primaria)
- **direccionHotel**
- **ciudadHotel**
- **cantidadHabitaciones**
- **dniGerente** (Clave foránea: Referencia al gerente del hotel)

**Justificación Clave Primaria:** el atributo **codHotel** es único para cada hotel, ya que lo identifica de manera exclusiva en el sistema, los otros atributos dependen funcionalmente de **codHotel**.

### 3 - Tabla **GERENTE**

- **dniGerente** (Clave primaria)
- **nombreGerente**

**Justificación Clave Primaria:** el atributo **dniGerente** es único para cada gerente y permite una relación clara con los hoteles que administra (clave foránea en la tabla anterior - HOTEL).

### 4 - Tabla **CLIENTE**

- **dniCliente** (Clave primaria)
- **nombreCliente**
- **ciudadCliente**

**Justificación Clave Primaria:** el atributo **dniCliente** identifica de manera única a cada cliente en el sistema y los otros atributos como **nombreCliente** y **ciudadCliente** dependen funcionalmente de **dniCliente**, pero no son lo suficientemente únicos por sí mismos.

---

### **CÓDIGO TABLA DIAGRAMA DB ORIGINAL**

```
Table ESTADIA {  
dniCliente varchar [not null]  
codHotel varchar [not null]  
cantidadHabitaciones int  
direccionHotel varchar  
ciudadHotel varchar  
dniGerente varchar  
nombreGerente varchar  
nombreCliente varchar  
ciudadCliente varchar  
fechaInicioHospedaje date  
cantDiasHospedaje int  
Habitacion int [pk]  
}
```

---

### **CÓDIGO TABLA 3FN**

```
Table CLIENTE {  
dniCliente varchar [pk] // Clave primaria: Identificador único del cliente  
nombreCliente varchar  
ciudadCliente varchar  
}
```

```
Table HOTEL {  
codHotel varchar [pk] // Clave primaria: Código único del hotel  
direccionHotel varchar  
ciudadHotel varchar  
cantidadHabitaciones int  
dniGerente varchar [ref: > GERENTE.dniGerente] // Clave foránea: Referencia al gerente del hotel  
}
```

```
Table GERENTE {  
dniGerente varchar [pk] // Clave primaria: Identificador único del gerente  
nombreGerente varchar  
}
```

```
Table ESTADIA {  
dniCliente varchar [ref: > CLIENTE.dniCliente] // Clave foránea: Referencia al cliente  
codHotel varchar [ref: > HOTEL.codHotel] // Clave foránea: Referencia al hotel  
fechaInicioHospedaje date  
numHabitacion int  
cantDiasHospedaje int  
// Clave candidata compuesta primaria  
indexes {  
(dniCliente, codHotel, fechaInicioHospedaje, numHabitacion) [pk]  
}  
}
```