

# UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE CIENCIAS.  
FUNDAMENTOS DE BASES DE DATOS.

## PRÁCTICA 05

LENGUAJE PARA DEFINICIÓN DE DATOS. DDL, (DATA DEFINITION  
LANGUAGE).

---

### EQUIPO: CHIQUESSQL

IVANA IX CHEL BONILLA NEGRETE  
315131994

DYLAN ENRIQUE JUAREZ MARTINEZ  
422117180

DANIEL ROJO MATA  
314297967

---

### PROFESOR:

GERARDO ÁVILES ROSAS

---

### AYUDANTES DE TEORÍA:

GERARDO URIEL SOTO MIRANDA  
VALERIA FERNANDA MANJARREZ ANGELES

---

### AYUDANTES DE LABORATORIO:

RICARDO BADILLO MACÍAS  
ROCÍO AYLIN HUERTA GONZÁLEZ

## I. CORRECCIONES REALIZADAS EN EL MODELO RELACIONAL

- Se reemplazó telefonoPersona por: telefonoPersonalApoyo, telefonoLimpiezaInterna, telefonoServicioComida, telefonoRecepcionista
- Se reemplazó correoPersona por: correoPersonalApoyo, correoLimpiezaInterna, correoServicioComida, correoRecepcionista
- RFC en empleado se cambió a llave candidata, es decir, ya no es llave primaria.
- Se corrigió la tabla atender con solo las llaves de Evento y Personal de Apoyo.
- Se realizaron las tablas para Disponer y Rentar para los tipos de habitaciones.
- Se añadió un ID específico para la forma de pago con tarjeta y otro para la forma de pago en efectivo.

## II. RESTRICCIONES DE DOMINIO DE LA BASE DE DATOS 'HOTELES LUNA'

### ■ Restricciones de dominio para la Tabla Hotel

1. **hotel\_nombre\_no\_vacio** : Se asegura que el nombre del establecimiento no sea la cadena vacía CHECK(nombreEstablecimiento <>'');
2. **hotel\_checkIn\_no\_null** : Se asegura que la hora de check-in no sea NULL CHECK(horaCheckIn IS NOT NULL);
3. **hotel\_checkOut\_no\_null** : Se asegura que la hora de check-out no sea NULL CHECK(horaCheckOut IS NOT NULL);
4. **hotel\_petfriendly\_no\_vacio** : Se asegura que petFriendly no sea la cadena vacía CHECK(petFriendly <>'');
5. **petfriendly\_valores\_permitidos** : Se asegura que la columna petFriendly solo acepte los valores SÍ y NO CHECK (petFriendly IN ('SI', 'NO'));
6. **hotel\_servicio\_no\_vacio** : Se asegura que servicio no sea la cadena vacía CHECK(servicio <>'');
7. **hotel\_numeroInterior\_no\_vacio** : Se asegura que numeroInterior no sea la cadena vacía CHECK(numeroInterior <>'');
8. **hotel\_numeroExterior\_no\_null** : Se asegura que numeroExterior no sea NULL CHECK(numeroExterior IS NOT NULL);
9. **hotel\_colonia\_no\_vacia** : Se asegura que la colonia no sea la cadena vacía CHECK(colonia <>'');
10. **hotel\_calle\_no\_vacia** : Se asegura que la calle no sea la cadena vacía CHECK(calle <>'');
11. **hotel\_estado\_no\_vacio** : Se asegura que el estado no sea la cadena vacía CHECK(estado <>'');

12. **numeroInterior** – Se escribe 'SN' para el caso donde no se tenga el dato  
SET DEFAULT('SN');  
*ALTER TABLE hotel ADD CONSTRAINT numeroExterior\_check CHECK(numeroExterior > 0);*– Se asegura de que el número exterior solo reciba valores positivos

■ **Restricciones de dominio para la Tabla telefonoHotel**

1. **telefonoHotel\_no\_vacio** : para garantizar que el campo telefonoHotel no esté vacío  
CHECK (telefonoHotel <>'');  
2. **telefono\_Hotel\_valido**: Se asegura de recibir 10 dígitos entre 0 y 9 cada uno.

■ **Restricciones de dominio para las Tabla telefono con los tipos de personal**

1. **telefonoTipodePersonal\_no\_vacio** : para garantizar que el campo telefonoPersona no esté vacío. CHECK (telefonoPersona <>'');  
2. **telefonoPersona\_valido**: Se asegura de recibir 10 dígitos entre 0 y 9 cada uno.

■ **Restricciones de dominio para la Tabla Correo Persona**

1. **correoPersona\_no\_vacio** : Restricción CHECK para garantizar que el campo correoElectronico no esté vacío. CHECK (correoElectronico <>'');

■ **Restricciones de dominio para la Tabla Empleado**

1. Estas columnas **RFC ,nombre, paterno, materno,fechaNacimiento, genero, horario, numeroInterior, numeroExterior, colonia, calle, estado, fechaContratacion** no pueden tener valor NULL por lo que agregamos la instruccion Set NOT NULL.  
2. **genero\_valido** : para asegurar que el género sea 'M' o 'H'  
3. **numeroInterior** : Se escribe 'SN' para el caso donde no se tenga el dato  
SET DEFAULT('SN');  
4. **RFC\_check**:Se asegura de que los 13 dígitos del RFC coincidan con la estructura usual.  
CHECK(RFC '[A-Z]4[0-9]6[A-Z]2[0-9]1');  
5. **genero\_valido**: para asegurar que el género sea 'M' o 'H'. CHECK (genero IN ('M', 'H'));  
6. **numeroExterior\_check**:Se asegura de que el número exterior solo reciba valores positivos.CHECK(numeroExterior >0);

■ **Restricciones de dominio para la Tabla Efectivo**

1. **idFormaEfectivo** : La columna idForma no puede ser null por lo que agregamos la instruccion SET NOT NULL

■ **Restricciones de dominio para la Tabla Huesped**

1. Estas columnas **nombre, paterno, materno, fechaNacimiento, genero, idForma** no pueden tener valor null, por lo que agregamos la instruccion SET NOT NULL.  
2. **genero\_valido**: Restricción CHECK para asegurar que el género sea 'M' o 'H'.  
CHECK (genero IN ('M', 'H'));

■ **Restricciones de dominio para la Tabla TelefonoHuesped**

1. **telefonoPersona\_no\_vacio** : para garantizar que el campo telefonoPersona no esté vacío.CHECK (telefonoPersona <>'');

2. **telefonoHuesped\_valido**: Se asegura de recibir 10 dígitos entre 0 y 9 cada uno.

■ **Restricciones de dominio para la Tabla CorreoHuesped**

1. **CorreoPersona\_no\_vacio** : para garantizar que el campo correoElectronico no esté vacío. CHECK (correoElectronico <>"");

■ **Restricciones de dominio de la Tabla Habitacion**

1. Estas columnas **idPersona**, **RFC**, **disponibilidad**, **costoPorNoche**, **numeroDeHabitacion**, **numeroDeCamas** no pueden tener valor nulo, por lo que agregamos la instruccion SET NOT NULL.
2. **disponibilidad\_check** : Restricción para asegurar que disponibilidad sea TRUE o FALSE. CHECK (disponibilidad IN (TRUE, FALSE));
3. **costoPorNoche\_positivo\_check**: Restricción para asegurar que costoPorNoche sea positivo. CHECK (costoPorNoche >= 0);
4. **numeroDeHabitacion\_positivo\_check**: Restricción para asegurar que numeroDeHabitacion sea positivo. CHECK (numeroDeHabitacion >0);

■ **Restricciones de dominio de la Tabla Doble**

1. Estas columnas **idHabitacion**, **idHuesped**, **RFC**, **disponibilidad**, **costoPorNoche**, **numeroDeHabitacion**, **numeroDeCamas** no pueden tener valor nulo, por lo que, agregamos la instruccion SET NOT NULL.
2. **disponibilidad\_check\_doble** : Restricción para asegurar que disponibilidad sea TRUE o FALSE. CHECK (disponibilidad IN (TRUE, FALSE));
3. **costoPorNoche\_positivo\_check\_doble** : Restricción para asegurar que costoPorNoche sea positivo. CHECK (costoPorNoche >= 0);
4. **numeroDeHabitacion\_positivo\_check\_doble** : Restricción para asegurar que numeroDeHabitacion sea positivo. CHECK (numeroDeHabitacion >0);
5. **numeroDeCamas\_doble** : Restricción para asegurar que numeroDeCamas sea mayor o igual a 1. CHECK (numeroDeCamas ≥ 1);

■ **Restricciones de dominio de la Tabla Cuadruple**

1. Estas columnas **idPersona**, **RFC**, **disponibilidad**, **costoPorNoche**, **numeroDeHabitacion**, **numeroDeCamas** no pueden tener valor nulo, por lo que agregamos la instruccion SET NOT NULL.
2. **disponibilidad\_check\_doble** : Restricción para asegurar que disponibilidad sea TRUE o FALSE. CHECK (disponibilidad IN (TRUE, FALSE));
3. **costoPorNoche\_positivo\_check\_doble** : Restricción para asegurar que costoPorNoche sea positivo. CHECK (costoPorNoche >= 0);
4. **numeroDeHabitacion\_positivo\_check\_doble** : Restricción para asegurar que numeroDeHabitacion sea positivo. CHECK (numeroDeHabitacion >0);
5. **numeroDeCamas\_doble** : Restricción para asegurar que numeroDeCamas sea mayor o igual a 2. CHECK (numeroDeCamas ≥ 2);

■ **Restricciones de dominio de la Tabla PentHouse**

1. Estas columnas **idPersona**, **RFC**, **disponibilidad**, **costoPorNoche**, **numeroDeHabitacion**, **numeroDeCamas** no pueden tener valor nulo, por lo que agregamos la instruccion SET NOT NULL.
  2. **disponibilidad\_check\_doble** : Restricción para asegurar que disponibilidad sea TRUE o FALSE. CHECK (disponibilidad IN (TRUE, FALSE));
  3. **costoPorNoche\_positivo\_check\_doble** :Restricción para asegurar que costoPorNoche sea positivo. CHECK (costoPorNoche >= 0);
  4. **numeroDeHabitacion\_positivo\_check\_doble** : Restricción para asegurar que numeroDeHabitacion sea positivo. CHECK (numeroDeHabitacion >0);
  5. **numeroDeCamas\_doble** : Restricción para asegurar que numeroDeCamas sea mayor o igual a 2. CHECK (numeroDeCamas ≥ 2);
- **Restricciones de dominio de la Tabla ServicioComida**
1. Estas columnas **nombre**, **apellidoPaterno**, **apellidoMaterno**, **fechaNacimiento**, **genero**, **RFC**, **numeroExterior**, **colonia**, **calle**, **estado**, **idHotel** no pueden tener valor nulo, por lo que, agregamos la instruccion SET NOT NULL.
  2. **numeroInterior**: se escribe "SN" donde no se tenga el dato. SET DEFAULT('SN');
  3. **RFC\_check**: Se asegura de que los 13 dígitos del RFC coincidan con la estructura usual. CHECK(RFC '[A-Z]4[0-9]6[A-Z]2[0-9]1');
  4. **genero\_valido**: para asegurar que el género sea 'M' o 'H'. CHECK (genero IN ('M', 'H'));
  5. **numeroExterior\_check**: Se asegura de que el número exterior solo reciba valores positivos. CHECK(numeroExterior >0);
- **Restricciones de dominio de la Tabla Recepcionista**
1. Estas columnas **nombre**, **apellidoPaterno**, **apellidoMaterno**, **fechaNacimiento**, **genero**, **RFC**, **numeroExterior**, **colonia**, **calle**, **estado**, **idHotel** no pueden tener valor nulo, por lo que agregamos la instruccion SET NOT NULL.
  2. **numeroInterior**: se escribe "SN". en caso de que no se tenga el dato. SET DEFAULT('SN')
  3. **RFC\_check**: Se asegura de que los 13 dígitos del RFC coincidan con la estructura usual. CHECK(RFC '[A-Z]4[0-9]6[A-Z]2[0-9]1');
  4. **genero\_valido**: para asegurar que el género sea 'M' o 'H'. CHECK (genero IN ('M', 'H'));
  5. **numeroExterior\_check**: Se asegura de que el número exterior solo reciba valores positivos. CHECK(numeroExterior >0);
- **Restricciones de dominio de la Tabla PersonalDeApoyo**
1. Estas columnas **nombre**, **apellidoPaterno**, **apellidoMaterno**, **fechaNacimiento**, **genero**, **RFC**, **numeroExterior**, **colonia**, **calle**, **estado**, **idHotel** no pueden tener valor nulo, por lo que agregamos la instruccion SET NOT NULL.
  2. **numeroInterior**: se escribe "SN". en caso de que no se tenga el dato. SET DEFAULT('SN')
  3. **RFC\_check**: Se asegura de que los 13 dígitos del RFC coincidan con la estructura usual. CHECK(RFC '[A-Z]4[0-9]6[A-Z]2[0-9]1');
  4. **genero\_valido**: para asegurar que el género sea 'M' o 'H'. CHECK (genero IN ('M', 'H'));
  5. **numeroExterior\_check**: Se asegura de que el número exterior solo reciba valores positivos. CHECK(numeroExterior >0);

### ■ Restricciones de dominio de la Tabla LimpiezaInterna

1. Estas columnas **nombre,apellidoPaterno,apellidoMaterno,fechaNacimiento,genero,RFC,numeroExterior,colonia,calle,estado,idHotel** no pueden tener valor nulo, por lo que agregamos la instruccion SET NOT NULL.
2. **numeroInterior**: se escribe "SN". en caso de que no se tenga el dato. SET DEFAULT('SN')
3. **RFC\_check**:Se asegura de que los 13 dígitos del RFC coincidan con la estructura usual. CHECK(RFC '[A-Z]4[0-9]6[A-Z]2[0-9]1');
4. **genero\_valido**: para asegurar que el género sea 'M' o 'H'. CHECK (genero IN ('M', 'H'));
5. **numeroExterior\_check**:Se asegura de que el número exterior solo reciba valores positivos.CHECK(numeroExterior >0);

### ■ Restricciones de dominio de la Tabla Salon

1. Estas columnas **idSalon, horario, nombreSalon, capacidad, numeroExterior, colonia, calle, estado** no pueden tener valor nulo, por lo que, agregamos la instruccion SET NOT NULL.
2. **capacidad\_check**:Se asegura de que la capacidad solo reciba valores positivos. CHECK(capacidad >0);
3. **numeroExterior\_check**:Se asegura de que el número exterior solo reciba valores positivos. CHECK(numeroExterior >0);

### ■ Restricciones de dominio de la Tabla Evento

1. Estas columnas **idEvento, nombreSalon, numAsistentes, fechaEvento, horaInicio, horaFin, idSalon** no pueden tener valor nulo, por lo que, agregamos la instruccion SET NOT NULL.
2. **numAsistentes\_check**: Se asegura de que el número de asistentes solo reciba valores positivos. CHECK(numAsistentes >0);

### ■ Restricciones de dominio de la Tabla Encuesta

1. Estas columnas **idEncuesta, servicioRecibido, limpieza, recomendacion, volverHospedar, idPersona** no pueden tener valor nulo, por lo que, agregamos la instruccion SET NOT NULL.
2. **check\_servicioRecibido**: Restricción de evaluación del servicio recibido. CHECK (servicioRecibido BETWEEN 1 AND 5);
3. **check\_limpieza**: Restricción de evaluación de la limpieza. CHECK (limpieza BETWEEN 1 AND 5);
4. **check\_recomendacion**: Restricción de evaluación de la recomendación. CHECK (recomendacion BETWEEN 1 AND 5);
5. **check\_volverHospedar**: Restricción de evaluación de volver a hospedarse. CHECK (volverHospedar BETWEEN 1 AND 5);
6. **comentario**: Valor predeterminado para comentario. SET DEFAULT '';

### ■ Restricciones de dominio de la Tabla membresia

1. Estas columnas **numeroMembresia,activa,idPersona** no pueden tener valor nulo,por lo que, agregamos la instruccion SET NOT NULL.

2. **activa\_check** : restriccion para activa (Solo 'SI' o 'NO')  
CHECK (activa IN ('SI', 'NO'));
- **Restricciones de dominio de la Tabla Encuesta**
    1. Estas columnas **idForma**, **numeroTarjeta**, **vencimiento** no pueden tener valor nulo,por lo que, agregamos la instruccion SET NOT NULL.
    2. **numerotarjeta\_valido**: Se asegura de recibir 16 dígitos entre 0 y 9 cada uno.
  - **Restricciones de dominio de la tabla Pago**
    1. Estas columnas **idPago**, **NumeroNinos**, **NumeroAdultos**, **NumeroMascotas**, **NumeroNoches** no pueden tener valor nulo,por lo que, agregamos la instruccion SET NOT NULL.
    2. **NumeroNinos\_check**: Se asegura de que el número de niños solo reciba valores positivos.  
CHECK(NumeroNinos >0);
    3. **NumeroAdultos\_check**: Se asegura de que el número de adultos solo reciba valores positivos.  
CHECK(NumeroAdultos >0);
    4. **NumeroMascotas\_check**: Se asegura de que el número de mascotas solo reciba valores positivos. CHECK(NumeroMascotas >0);
    5. **NumeroNoches\_check**: Se asegura de que el número de noches solo reciba valores positivos.  
CHECK(NumeroNoches >0);
  - **Restricciones de dominio de la Tabla ServicioExtra**
    1. Estas columnas **idServicio**, **numeroPlatos**, **idPago** no pueden tener valor nulo,por lo que, agregamos la instruccion SET NOT NULL.
    2. **numeroPlatos\_check**: Se asegura de que el número de platos solo reciba valores mayores o iguales a cero.CHECK(numeroPlatos >= 0);
  - **Restricciones de dominio de la Tabla Disponer**
    1. Estas columnas **IdHabitacion**,**IdHotel** no pueden tener valor nulo,por lo que, agregamos la instruccion SET NOT NULL.
  - **Restricciones de dominio de las Tabla Atender para las distintas habitaciones**
    1. Estas columnas **IdEvento**,**IdPersona** no pueden tener valor nulo,por lo que, agregamos la instruccion SET NOT NULL.
  - **Restricciones de dominio de las Tabla Rentar para las distintas habitaciones**
    1. Estas columnas **IdPersona** no pueden tener valor nulo,por lo que, agregamos la instruccion SET NOT NULL.

## III. DOMINIO DE ATRIBUTOS Y RESTRICCIONES

- NumeroInterior: VARCHAR(10)
- NumeroExterior: INTEGER
- Colonia: VARCHAR(60)
- Calle: VARCHAR(70)
- Estado: VARCHAR(50)
- NumeroHabitacionesOcupadas: INTEGER
- NombreEstablecimiento: VARCHAR(50)
- Telefono: INTEGER
- NumeroHabitacionesDisponibles: INTEGER
- Numerototaldehabitaciones: INTEGER
- IDHotel: VARCHAR(20)
- Servicio: VARCHAR(100)
- Petfriendly: VARCHAR(2)
- HoraCheckIn: TIME
- HoraCheckOut: TIME
- IDEmpleado: VARCHAR(20)
- FechaContratacion: DATE
- RFC: VARCHAR(13)
- Horario: TIME
- NumeroHabitacionesaCargo: INTEGER
- IDHabitacion: VARCHAR(10)
- Disponibilidad: VARCHAR(13)
- CostoPorNoche: FLOAT
- NumeroDeCamas: INTEGER
- NumeroHabitacion: INTEGER
- FechaNacimiento: DATE
- CorreoElectronico: VARCHAR(50)
- Genero: VARCHAR(1)
- IDPersona: VARCHAR(20)
- ApellidoPaterno: VARCHAR(50)
- ApellidoMaterno: VARCHAR(50)
- Nombre: VARCHAR(50)
- Nacionalidad: VARCHAR(50)
- FechaEvento: DATE
- HoraInicio: TIME
- HoraFin: TIME
- NombreSalon: VARCHAR(50)
- NumAsistentes: INTEGER
- NombreHuesped: VARCHAR(50)
- NumMembresía: INTEGER
- IDEvento: VARCHAR(15)
- RFCPersonalapoyo: VARCHAR(13)
- HorarioAtencion: TIME
- Capacidad: INTEGER
- IDSalon: VARCHAR(10)
- NombreSalon: VARCHAR(50)
- Recomendacion: INTEGER
- Comentario: VARCHAR(150)
- VolverHospedar: INTEGER
- IDEncuesta: VARCHAR(10)
- ServicioRecibido: INTEGER
- Limpieza: INTEGER
- Precio: FLOAR
- IDServicio: VARCHAR(10)
- NumeroPlatos: INTEGER
- NumeroNiños: INTEGER
- NumeroAdultos: INTEGER



- NumeroMascotas: INTEGER
- NumeroNoches: INTEGER
- Preciofinal: FLOAT
- IDPago: VARCHAR(15)
- IDForma: VARCHAR(8)
- NumeroTarjeta: INTEGER
- FechaVencimiento: DATE
- Activa: VARCHAR(2)
- VisitasPorAño: INTEGER

Algunas restricciones:

- No habrá valores de tipo NULL.
- La evaluación en las columnas *Recomendacion*, *ServicioRecibido*, *Limpieza* y *VolverHospedar* será de tipo entero entre 1 a 5.
- Para el atributo *NumeroInterior* se permitirán valores enteros o 'SN' para el caso donde no se tenga el dato.
- El atributo Genero solo permitirá los valores 'M' (mujer) o 'H' (hombre).
- Petfriendly y activa solo permitirá los valores 'SI' Y 'NO'.