

Universidad ORT Uruguay

Instituto de Educación

**Obligatorio de
Bases De Datos 1
Segunda entrega**

**Entrega correspondiente a la carrera
Analista en Tecnologías de la Información /
Analista Programador**

**Juan Veiga – 208454
Marcelo Sanchez-214904**

Grupo N2C

Docente: Kleber Marsicano

Setiembre-Diciembre 2016

DECLARACIÓN DE AUTORÍA

Marcelo Sanchez y Juan Veiga declaran que el presente trabajo es de su entera autoría.

Puedo asegurar que:

- El trabajo fue producido en su totalidad mientras realizaba la primera entrega del obligatorio de Bases de Datos 1
- Cuando he consultado el trabajo publicado por otros, lo he atribuido con claridad.
- Cuando cité obras de otros, he indicado las fuentes. Con excepción de estas citas, la obra es enteramente mía.
- En el trabajo, he acusado recibo de las ayudas recibidas.
- Ninguna parte de este trabajo ha sido publicada previamente a su entrega.

Relevamiento (Letra modificada):

Nuestro cliente desea implementar un sistema para gestionar la reserva de habitaciones en hoteles de todo el mundo. Debe permitir al usuario consultar sobre los hoteles, habitaciones disponibles y registrar reservas para las mismas, hacer consultas sobre las mismas, modificarlas o cancelarlas, así como pagar las estadías correspondientes (el pago previo congela el precio dejando como saldo exclusivamente los consumos extra).

El sistema debe mostrar una interfaz en la que se describan los distintos tipos de habitaciones disponibles y los servicios *(siendo estos los tipo de habitación con un regimen)* ofrecidos por cada hotel junto con la opción para registrarse, si es la primera vez que el usuario accede al sistema, o ingresar al mismo con su usuario y contraseña si ya se encuentra registrado, en cuyo caso ingresará con su correo que será la identificación de usuario (email) y su contraseña (clave). Las habitaciones que puedan haber sido reservadas o precontratadas por fuera del sistema (personal o telefónicamente en el propio hotel) no podrán ser reservadas o contratadas por este sistema mientras figuren como “ocupadas”.

Una vez ingresado, el usuario podrá acceder a las siguientes opciones:

- Consulta de hoteles
 - Por Ciudad
 - Por Nombre
 - Por Fecha
 - Por tarifa (rango de precios)
 - Por categoría (cantidad de estrellas)
- Reserva de habitaciones por un período de tiempo (cada hotel puede tener una política diferente de señas)
- Prepago de estadía (congelando el precio a la tarifa vigente al momento de la compra)
 - Con reserva previa
 - Sin reserva previa

De los huéspedes se desea registrar su identificación (ldhuesped) su nombre (nombre) su primer apellido (apaterno) su segundo apellido (amaterno) su tipo de documento (tipo_documento) el número de documento (num_documento) la fecha de nacimiento (fecha_nacimiento) el teléfono (teléfono) y su correo electrónico y si tiene usuario en el sistema, su identificación (email que debe coincidir con un email registrado **para poder**

cambiar el estado de la reserva de pendiente a ejecutada) y el país al que pertenece. El sistema tendrá información sobre las habitaciones ocupadas pero solo podrá identificar al ocupante (responsable) si fue reservada o prepaga a través del sistema. Cada reserva o prepago tiene un id autonumerado único y se debe corresponder a una o más habitaciones de un hotel por un período de tiempo. Las habitaciones tienen una capacidad de camas y puede tener **una** cama matrimonial o no. Cuando un usuario ocupa la o las habitaciones que había reservado esto debe ser registrado en el sistema.

De los hoteles se desea tener la identificación (idahotel) y el nombre (nombre), en que ciudad y país está ubicado, la cantidad de estrellas, la dirección, en que zona/barrio de dicha ciudad está ubicado, un pequeño plano en formato JPG o PNG con su ubicación y las habitaciones que tiene disponibles y ocupadas en un momento dado con sus características básicas (cantidad de camas, cama matrimonial o individuales, aire acondicionado, balcón, vista, amenities, etc.). Se sabe que la tarifa puede variar en algunos hoteles si se decide contratar solo alojamiento, media pensión, pensión completa u all-inclusive. No todos los hoteles ofrecen estas cuatro opciones. También se debe registrar que servicios están disponibles en el hotel, por ejemplo parking privado, piscina abierta y/o cerrada, cajero automático, cambio de divisas, gym, restaurantes, servicio en habitaciones 24hs, lavandería, wifi, etc y si éstos tienen un costo extra. También se quiere saber para cada hotel pertenece a una cadena internacional. Las tarifas tienen una fecha en que entran en vigencia, **en la cual se deberá guardar un histórico** y su validez sigue hasta que entre en vigencia una nueva.

Las reservas se identifican por el número de reserva (idreserva) y para ella se debe almacenar, además de la cantidad de personas y habitaciones (tipo) y el período de tiempo si se abonó completamente (**siendo esto un prepago**) o si solo se pagó el monto acordado (**seña**) por la reserva, la fecha en que se reservó (fechaRes) y las observaciones correspondientes (observación). En el caso del prepago se deberá ingresar la información de cada uno de los huéspedes de cada habitación contratada en esa transacción.

Todo pago hecho a través del sistema está identificado por un número autoincremental (idpago) y debe poder identificar a que reserva o prepago corresponde, a ~~que pasajero~~ (~~idpasajero~~) **que usuario (idUserario)** corresponde dicha reserva **o prepago**, la fecha en la que se efectuó dicho pago (fecha), el monto (monto) del mismo.

RESTRICCIONES DE INTEGRIDAD

1. Ri-En DISPONE- El costo tiene que ser mayor o igual a cero.
2. Ri- En DISPONE- Si dispone no tiene costo este será valor 0 no aceptara null.
3. Ri- En DISPONE- El atributo fechaVigencia cuando se guarde en el sistema no deberá ser anterior a la fecha actual de guardado y no acepta null.
4. Ri- En HABITACION- A un mismo numHabitacion no se le puede asignar un número mayor de huéspedes que el valor calculable **plazas** de la misma.
5. Ri. En HABITACION- Una habitación tendrá como máximo una cama matrimonial.
6. Ri- En HABITACION- El atributo plaza es un calculable de la sumatoria de camas simples (1 plaza) y matrimonial (2 plazas).
7. Ri- En OCUPACION- Los atributos fechaDesde y fechaHasta no pueden ser nulos.
8. Ri- En OCUPACION- No se puede ingresar una fechaDesde sino existe una fechaHasta anterior o igual en la escala de tiempo.(sino está cerrado el ciclo anterior).
9. Ri-En OCUPACION- El atributo fechaDesde debe ser menor fechaHasta.
10. Ri-En OCUPACION-fechaDesde tiene que ser igual o mayor a la fecha actual.
11. Ri-En HUESPED- Si el usuario pasa a ser huesped y tiene reserva se ejecuta la misma. Cambia el estado de la misma.(en RESERVA).
12. Ri.-En ASIGNA- Un mismo Huésped no puede ocupar más de una habitación.
13. Ri-En ASIGNA- No se puede asignar más huéspedes que plazas disponibles en una habitación.

14. Ri- En RESERVA- El atributo estado tiene tres valores posibles (Pendiente) (Ejecutada) o (Cancelada). Pendiente (todavía no se utilizó).Ejecutada(ya se utilizó)y Cancelada (queda sin efecto la misma).
15. Ri.-En RESERVA- El atributo idReserva es un entero autoincremental.
16. Ri-En RESERVA- El atributo numeroPersonas es un entero que registra el total de personas de la suma de todos los servicios con ese mismo idReserva.
17. Ri-En RESERVA- El atributo numeroPersonas deberá ser menor o igual a la sumatoria del valor calculable plazas.
18. Ri-En TIENE- No se puede reservar un servicio si no tiene habitaciones disponibles para ese rango de fechas.
19. Ri-En TIENE- fechaSalida y fechaIngreso no pueden ser nulos.
20. Ri-En TIENE-fechaIngreso tiene que ser menor que fechaSalida y mayor o igual a la fecha de la transacción.
21. Ri- En PRACTICÓ- El atributo fechaReserva tiene que ser igual a la fecha del día de la transacción.
22. Ri.-En PRACTICÓ- Señal es un decimal e indica el porcentaje de señal que establece el hotel en el cual se hizo la reserva.
23. Ri-En PRACTICÓ - El costo tiene que ser mayor o igual a cero.
24. Ri- En PRACTICÓ - Si dispone no tiene costo este será valor 0 no aceptara null.
25. Ri.-En EFECTUO- El monto que se abona en pago tiene que ser igual al total a la sumatoria de los servicios contratados.
26. Ri.- En EFECTUO- El atributo fechaPrepago tiene que ser igual a la fecha del día de la transacción.
27. Ri.-En PAGO- El atributo idPago es un entero autoincremental.
28. Ri. -En PAGO- El atributo monto es calculable, si la relación es con RESERVA, el monto es un porcentaje (señal) de la sumatoria de las tarifas y si la relación es con PREPAGO el mismo es el total de la sumatoria de las tarifas
29. Ri- En REGIMEN- El atributo tipo tomara cuatro opciones ALOJAMIENTO-MEDIA PENSION-PENSION COMPLETA - ALL-INCLUSIVE.

30. Ri-En SERVICIO- El atributo costoServicio será mayor que 0.
31. Ri.- En TARIFA- El atributo costo tiene que ser mayor que cero.
32. Ri- En TARIFA- El atributo fechaVigencia cuando se guarde en el sistema deberá ser igual a la fecha actual de guardado.

Modelo Relacional en 3ªFN

Cadena (**nomCadena**)

Hotel (**idHotel**, nomHotel, numEstrellas, totalHab, habDisp⁺ país, ciudad, distrito, dirección, plano, porcentajeSeña)

Pertenece (**idHotel**, nomCadena)

ServicioExtra (**nomServExtra**)

Dispone (**idHotel**, **nomServExtra**, **fechaVigencia**, costo)

TipoHabitacion (**numCamas**, **matrimonial**, totalHab⁺)

Regimen (**tipoRegimen**)

Servicio (**numCamas**, **matrimonial**, **tipoRegimen**)

Tarifa (**numCamas**, **matrimonial**, **tipoRegimen**, **fechaVigencia**, **idHotel**, costoServicio)

Rige (**idHotel**, **numCamas**, **matrimonial**, **tipoRegimen**)

Reserva (**idReserva**, numeroPersonas, estado, email, pago, fechaReserva)

Tiene (**idHotel**, **numCamas**, **matrimonial**, **tipoRegimen**, **idReserva**, cantidad, fechaIngreso, fechaSalida, observaciones)

Prepago (**idReserva**)

Usuario (**email**, contraseña)

Huésped (**idHuesped**, numHabitacion, fechaNac, numDocumento, tipoDocumento, nombre, apellidoPat, apellidoMat, país)

TelefonoHuesped (**idHuesped**, **telefono**)

EmailHuesped (**idHuesped**, **email**)

Es (**email**, idHuesped)

Habitacion (**idHotel**, **numHabitacion**, características, plazas⁺, numCamas, matrimonial)

Ocupacion (**idHotel**, **numHabitacion**, **fechaDesde**, fechaHasta)

Asigna (**idHotel**, **numHabitacion**, **fechaDesde**, **idHuesped**)

Corresponde (idReserva, **idHotel**, **numHabitacion**, **fechaDesde**)

Cuadro de Claves:

Tabla	Primary Key	Unique	Foreign Key → Tabla
Cadena	nomCadena		
Hotel	idHotel		
Pertenece	idHotel		nomCadena → Cadena idHotel → Hotel
ServicioExtra	nomServExtra		
Dispone	{idHotel, nomServExtra, fechaVigencia}	fechaVigencia	idHotel → Hotel nomServExtra → ServicioExtra
TipoHabitacion	{numCamas, matrimonial}		
Regimen	{tipoRegimen}		
Servicio	{numCamas, matrimonial, tipoRegimen}		{numCamas, matrimonial} → TipoHabitacion tipoRegimen → Regimen
Tarifa	{numCamas, matrimonial, tipoRegimen, fechaVigencia, idHotel}		{numCamas, matrimonial} → TipoHabitacion tipoRegimen → Regimen idHotel → Hotel
Rige	{idHotel, numCamas, matrimonial, tipoRegimen}		idHotel → Hotel {numCamas, matrimonial} → TipoHabitacion tipoRegimen → Regimen
Reserva	{idReserva, numeroPersonas, estado, email, pago, fechaReserva}		
Tiene	{idHotel, numCamas, matrimonial, tipoRegimen,		idHotel → Hotel {numCamas, matrimonial} →

	idReserva}		TipoHabitacion tipoRegimen → Regimen idReserva → Reserva
Prepago	idReserva		idReserva → Reserva
Usuario	email		
Huesped	idHuesped		{idHuesped, email} → EmailHuesped {idHuesped, telefono} → TelefonoHuesped
TelefonoHuesped	{idHuesped, telefono}		idHuesped → Huesped
EmailHuesped	{idHuesped, email}		idHuesped → Huesped
Es	email		email → Usuario
Habitacion	{idHotel, numHabitacion}		idHotel → Hotel
Ocupacion	{idHotel, numHabitacion, fechaDesde}		idHotel → Hotel {idHotel, numHabitacion} → Habitacion
Asigna	{idHotel, numHabitacion, fechaDesde, idHuesped}		idHotel → Hotel {idHotel, numHabitacion} → Habitacion {idHotel, numHabitacion, fecha desde} → Ocupacion idHuesped → Huesped
Corresponde	{idHotel, numHabitacion, fechaDesde}		idHote → Hotel {idHotel, numHabitacion} → Habitacion {idReserva} → Prepago

```

CREATE TABLE Hotel
(
    Id_Hotel          NUMERIC (5),
    Nom_Hotel         VARCHAR(30) NOT NULL,
    Num_Estrellas     VARCHAR(30) NOT NULL,
    Total_Habitacion  NUMERIC(5) NOT NULL,
    Hab_Disponible    NUMERIC (5),
    Pais              VARCHAR(30),
    Ciudad            VARCHAR(30),
    Distrito          VARCHAR(30),
    Direccion         VARCHAR(50) NOT NULL,
    Plano             VARCHAR (50),
    Porcentaje_Sena   INT,

    Constraint    PK_Hotel    PRIMARY KEY(Id_Hotel)

);

CREATE TABLE Cadena
(
    Nom_Cadena        VARCHAR(30),

    Constraint    PK_Cadena  PRIMARY KEY(Nom_Cadena)

);

CREATE TABLE Pertenece
(
    Nom_Cadena        VARCHAR(30),
    Id_Hotel          NUMERIC(5) NOT NULL,

    Constraint    PK_Pertenece PRIMARY KEY(Id_Hotel),
    Constraint    Fk_Pertenece_Hotel FOREIGN KEY (Id_Hotel) REFERENCES Hotel(Id_Hotel),

);

CREATE TABLE Servicio_Extra
(
    Nom_Serv_Extra    VARCHAR(30),

    Constraint    PK_Servicio_Extra PRIMARY KEY(Nom_Serv_Extra),

);

CREATE TABLE Dispone
(
    Nom_Serv_Extra    VARCHAR(30),
    Id_Hotel          NUMERIC(5),
    Costo             INT NOT NULL,
    Fecha_Vigencia    DATE UNIQUE,

    Constraint    PK_     PRIMARY KEY (Id_Hotel,Nom_Serv_Extra,Fecha_Vigencia),
    Constraint    FK_Dispose_Servicio_Extra FOREIGN KEY (Nom_Serv_Extra) REFERENCES
Servicio_Extra(Nom_Serv_Extra),
    Constraint    FK_Dispose_Hotel FOREIGN KEY (Id_Hotel) REFERENCES Hotel(Id_Hotel)
);

CREATE TABLE Tipo_Habitacion
(
    Num_Camas         NUMERIC(2) NOT NULL,
    Matrimonial       CHAR(2) NOT NULL,

```

```

        Constraint PK_Tipo_Habitacion PRIMARY KEY (Num_Camas,Matrimonial),
    );

CREATE TABLE Regimen
(
    Tipo_Regimen    VARCHAR (30),

    Constraint PK_Regimen PRIMARY KEY (Tipo_Regimen),
);

CREATE TABLE Servicio
(
    Num_Camas        NUMERIC(2),
    Matrimonial      CHAR(2),
    Tipo_Regimen     VARCHAR (30),

    Constraint      PK_Servicio PRIMARY KEY ( Num_Camas,Matrimonial,Tipo_Regimen),
    Constraint      FK_Servicio_Tipo_Habitacion FOREIGN KEY ( Num_Camas,Matrimonial)
REFERENCES Tipo_Habitacion( Num_Camas,Matrimonial),
    Constraint      FK_Servicio_Regimen FOREIGN KEY ( Tipo_Regimen) REFERENCES Regimen(
Tipo_Regimen),
);

CREATE TABLE Tarifa
(
    Num_Camas        NUMERIC(2),
    Matrimonial      CHAR(2),
    Tipo_Regimen     VARCHAR (30),
    Fecha_Vigencia   DATE,
    Id_Hotel          NUMERIC(5),
    Costo            INT,

    Constraint PK_Dedica PRIMARY KEY (Num_Camas,Matrimonial,Tipo_Regimen,
Id_Hotel,Fecha_Vigencia),
    Constraint FK_Tarifa_Tipo_Habitacion FOREIGN KEY (Num_Camas,Matrimonial)
REFERENCES Tipo_Habitacion(Num_Camas,Matrimonial),
    Constraint FK_Tarifa_Regimen FOREIGN KEY (Tipo_Regimen) REFERENCES
Regimen(Tipo_Regimen),
    Constraint FK_Tarifa_Hotel FOREIGN KEY ( Id_Hotel) REFERENCES Hotel( Id_Hotel),
);

CREATE TABLE Rige
(
    Num_Camas NUMERIC(2),
    Matrimonial CHAR(2),
    Tipo_Regimen VARCHAR (30),
    Id_Hotel NUMERIC(5),

    Constraint      PK_Rige PRIMARY KEY (Num_Camas,Matrimonial,Tipo_Regimen, Id_Hotel),
    Constraint      FK_Rige_Hotel FOREIGN KEY (Id_Hotel) REFERENCES Hotel(Id_Hotel),
    Constraint      FK_Rige_Tipo_Habitacion FOREIGN KEY (Num_Camas,Matrimonial)
REFERENCES Tipo_Habitacion(Num_Camas,Matrimonial),
    Constraint      FK_Rige_Regimen FOREIGN KEY (Tipo_Regimen) REFERENCES
Regimen(Tipo_Regimen),
);

```

```
);
```

```
CREATE TABLE Reserva
```

```
(  
    Id_Reserva NUMERIC (5),  
    Num_Personas NUMERIC (2) NOT NULL,  
    Estado VARCHAR (10) NOT NULL,  
    E-Mail VARCHAR (30) NOT NULL,  
    Pago INT,  
    Fecha_Reserva DATE NOT NULL,  
    Constraint PK_Reserva PRIMARY KEY (Id_Reserva),  
);
```

```
CREATE TABLE Tiene
```

```
(  
    Id_Reserva NUMERIC (5),  
    Num_Camas NUMERIC (2),  
    Matrimonial CHAR (2),  
    Tipo_Regimen VARCHAR (30),  
    Id_Hotel NUMERIC (5),  
    Cantidad INT,  
    Fecha_Ingreso DATE NOT NULL,  
    Fecha_Salida DATE NOT NULL,  
    Observaciones VARCHAR (50),  
  
    Constraint PK_Tiene PRIMARY KEY (Num_Camas, Matrimonial, Tipo_Regimen,  
    Id_Hotel, Id_Reserva),  
    Constraint FK_Tiene_Hotel FOREIGN KEY (Id_Hotel) REFERENCES Hotel (Id_Hotel),  
    Constraint FK_Tiene_Tipo_Habitacion FOREIGN KEY (Num_Camas, Matrimonial)  
    REFERENCES Tipo_Habitacion (Num_Camas, Matrimonial),  
    Constraint FK_Tiene_Regimen FOREIGN KEY (Tipo_Regimen) REFERENCES  
    Regimen (Tipo_Regimen),  
    Constraint FK_Tiene_Reserva FOREIGN KEY (Id_Reserva) REFERENCES  
    Reserva (Id_Reserva),  
    CHECK (Fecha_Ingreso <= Fecha_Salida)  
);
```

```
CREATE TABLE Prepago
```

```
(  
    Id_Reserva NUMERIC (5),  
  
    Constraint PK_Prepago PRIMARY KEY (Id_Reserva),  
);
```

```
CREATE TABLE Usuario
```

```
(  
    E-Mail VARCHAR (30) NOT NULL,  
    Contraseña CHAR (8) NOT NULL,  
  
    Constraint PK_Usuario PRIMARY KEY (E-Mail),  
);
```

```
CREATE TABLE Huesped
```

```
(  
    Id_Huesped INT NOT NULL,  
    Nom_Huesped VARCHAR (30) NOT NULL,
```

```

        Apellido_Pat      VARCHAR(30),
        Apellido_Mat      VARCHAR(30),
        Pais              VARCHAR(30),
        Num_Documento     INT UNIQUE,
        TipoDocumento     VARCHAR (20),
        Fecha_Nacimiento  DATE,

        Constraint        PK_Huesped PRIMARY KEY ( Id_Huesped),

);

CREATE TABLE Telefono_Huesped
(
        Id_Huesped      INT,
        Telefono        VARCHAR(30) NOT NULL,

        Constraint      PK_Telefono_Huesped PRIMARY KEY (Id_Huesped,Telefono),
        Constraint      FK_Telefono_Huesped_Huesped FOREIGN KEY (Id_Huesped) REFERENCES
Huesped(Id_Huesped),
);

CREATE TABLE Email_Huesped
(
        Id_Huesped      INT,
        E_Mail          VARCHAR(30) NOT NULL,

        Constraint      PK_Email_Huesped PRIMARY KEY (Id_Huesped,E_Mail),
        Constraint      FK_Email_Huesped_Huesped FOREIGN KEY (Id_Huesped) REFERENCES
Huesped(Id_Huesped),
);

CREATE TABLE Es
(
        Id_Huesped      INT NOT NULL,
        E_Mail          VARCHAR(30) NOT NULL,

        Constraint      PK_Es PRIMARY KEY (E_Mail),
        Constraint      FK_Es_Huesped FOREIGN KEY (Id_Huesped) REFERENCES
Huesped(Id_Huesped),
        Constraint      FK_Es_Usuario FOREIGN KEY (E_Mail) REFERENCES Usuario(E_Mail),
);

CREATE TABLE Habitacion
(
        Id_Hotel        NUMERIC(5),
        Num_Habitacion  INT,
        Caracteristicas VARCHAR(50),
        Plazas          INT,
        Num_Camas       NUMERIC(2),
        Matrimonial     CHAR(2),

        Constraint      PK_Habitacion PRIMARY KEY ( Id_Hotel,Num_Habitacion),
        Constraint      FK_Habitacion_Hotel FOREIGN KEY (Id_Hotel) REFERENCES
Hotel(Id_Hotel),

```

```
);
```

```
CREATE TABLE Ocupacion
```

```
(  
    Id_Hotel      NUMERIC(5),  
    Num_Habitacion INT,  
    Fecha_Desde   DATE,  
    Fecha_Hasta   DATE,
```

```
Constraint PK_Ocupacion PRIMARY KEY (Id_Hotel,Num_Habitacion,Fecha_Desde),
```

```
Constraint FK_Ocupacion_Hotel FOREIGN KEY (Id_Hotel) REFERENCES
```

```
Hotel(Id_Hotel),
```

```
Constraint FK_Habitacion_Habitacion FOREIGN KEY (Id_Hotel,Num_Habitacion)
```

```
REFERENCES Habitacion(Id_Hotel,Num_Habitacion),
```

```
);
```

```
CREATE TABLE Asigna
```

```
(  
    Id_Hotel      NUMERIC(5),  
    Num_Habitacion INT,  
    Fecha_Desde   DATE,  
    Id_Huesped     INT,
```

```
Constraint PK_Asigna PRIMARY KEY (
```

```
Id_Hotel,Num_Habitacion,Fecha_Desde,Id_Huesped),
```

```
Constraint FK_Asigna_Hotel FOREIGN KEY (Id_Hotel) REFERENCES Hotel(Id_Hotel),
```

```
Constraint FK_Asigna_Habitacion FOREIGN KEY (Id_Hotel,Num_Habitacion)
```

```
REFERENCES Habitacion(Id_Hotel,Num_Habitacion),
```

```
Constraint FK_Asigna_Ocupacion FOREIGN KEY  
(Id_Hotel,Num_Habitacion,Fecha_Desde) REFERENCES
```

```
Ocupacion(Id_Hotel,Num_Habitacion,Fecha_Desde),
```

```
);
```

```
CREATE TABLE Corresponde
```

```
(  
    Id_Hotel      NUMERIC(5),  
    Num_Habitacion INT,  
    Fecha_Desde   DATE,  
    Id_Reserva     NUMERIC (5),
```

```
Constraint PK_Corresponde PRIMARY KEY (Id_Hotel,Num_Habitacion,Fecha_Desde),
```

```
Constraint FK_Corresponde_Hotel FOREIGN KEY (Id_Hotel) REFERENCES
```

```
Hotel(Id_Hotel),
```

```
Constraint FK_Corresponde_Habitacion FOREIGN KEY (Id_Hotel,Num_Habitacion)
```

```
REFERENCES Habitacion(Id_Hotel,Num_Habitacion),
```

```
Constraint FK_Corresponde_Reserva FOREIGN KEY (Id_Reserva) REFERENCES
```

```
Reserva(Id_Reserva),
```

```
);
```