



Universidad Tecnológica de Panamá
Facultad de Ingeniería de Sistemas
Computacionales



Licenciatura en Ingeniería de Sistemas de Información
Sistemas de Base de Datos II

Laboratorio No.2

Modelado de Base de Datos y Normalización

Ing. Henry Lezcano

Integrantes:

Batista, Johel	8-914-587
Pinilla, Miguel	8-975-2460
Riley, Rolando	8-972-1033
Villarreal, Andrés	8-970-1267

Grupo: 1IF-131

Segundo Semestre, Año Académico 2022

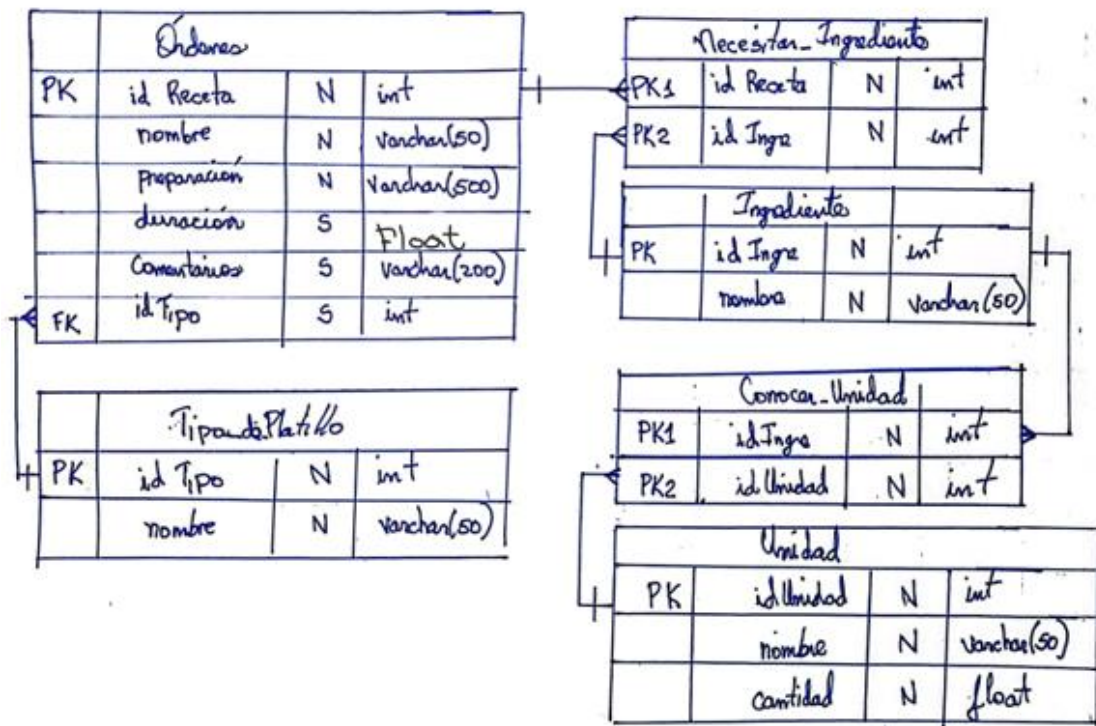
Caso #1

Indicaciones: Para el siguiente enunciado presentar el modelo conceptual E/R, el Modelo Lógico Relacional Normalizado y el Modelo Físico.

Consigna:

En la Cafetería del Edificio No.3 de la Universidad Tecnológica de Panamá para la preparación de un platillo el cocinero requiere de las recetas correspondientes, los ingredientes necesarios, el tipo de platillo que preparara y las unidades de medidas relacionadas. Para los tipos de platos se cuenta con su identificación y su nombre, para las recetas a preparar se cuenta con identificación, el nombre, las preparaciones, la duración y comentarios de esta, para el ingrediente se cuenta con identificación y el nombre, igualmente que para unidades que cuenta con la identificación y el nombre de unidad de medida, también como las cantidades de ingrediente a utilizar. Se requiere que el equipo de analistas diseñe un modelo conceptual Entidad/Relación que represente la preparación las diversas recetas para el platillo que está solicitando el cocinero.

Modelo Físico



Implementación del Caso #1 en PL/SQL

Tabla TipoDePlatillo

```
SQL> create table TipoDePlatillo (  
  2  idTipo int NOT NULL,  
  3  nombre varchar2(50) NOT NULL,  
  4  CONSTRAINT pk_TipoDePlatillo PRIMARY KEY (idTipo)  
  5  );  
  
Table created.
```

```
SQL> create table TipoDePlatillo (  
  
  2  idTipo int NOT NULL,  
  
  3  nombre varchar2(50) NOT NULL,  
  
  4  CONSTRAINT pk_TipoDePlatillo PRIMARY KEY (idTipo)  
  
  5);
```

Tabla Unidad

```
SQL> create table Unidad (  
  2  idUnidad int NOT NULL,  
  3  nombre varchar2(50) NOT NULL,  
  4  cantidad float NOT NULL,  
  5  CONSTRAINT pk_Unidad PRIMARY KEY (idUnidad)  
  6  );  
  
Table created.
```

```
SQL> create table Unidad (  
  
  2  idUnidad int NOT NULL,  
  
  3  nombre varchar2(50) NOT NULL,  
  
  4  cantidad float NOT NULL,
```

5 CONSTRAINT pk_Unidad PRIMARY KEY (idUnidad)

6);

Tabla Ingrediente

```
SQL> create table Ingrediente (  
  2  IdIngre int NOT NULL,  
  3  nombre varchar2(50) NOT NULL,  
  4  CONSTRAINT pk_Ingrediente PRIMARY KEY (IdIngre)  
  5  );  
  
Table created.
```

SQL> create table Ingrediente (

2 IdIngre int NOT NULL,

3 nombre varchar2(50) NOT NULL,

4 CONSTRAINT pk_Ingrediente PRIMARY KEY (IdIngre)

5);

Tabla Conocer_unidad

```
SQL> create table Conocer_unidad (  
  2  IdIngre int NOT NULL,  
  3  idUnidad int NOT NULL,  
  4  CONSTRAINT fk_Ingrediente FOREIGN KEY (IdIngre) REFERENCES Ingrediente(IdIngre),  
  5  CONSTRAINT fk_Unidad FOREIGN KEY (idUnidad) REFERENCES Unidad(idUnidad)  
  6  );  
  
Table created.
```

SQL> create table Conocer_unidad (

2 IdIngre int NOT NULL,

3 idUnidad int NOT NULL,

4 CONSTRAINT fk_Ingrediente FOREIGN KEY (IdIngre) REFERENCES
Ingrediente(IdIngre)

5);

Tabla Receta

```
SQL> create table Receta (  
  2 idReceta int NOT NULL,  
  3 nombre varchar2(50) NOT NULL,  
  4 preparacion varchar2(500) NOT NULL,  
  5 duracion float,  
  6 comentarios varchar(200),  
  7 idTipo int,  
  8 CONSTRAINT pk_Receta PRIMARY KEY (idReceta),  
  9 CONSTRAINT fk_TipoDePlatillo FOREIGN KEY (idTipo) REFERENCES TipoDePlatillo(idTipo)  
 10 );
```

Table created.

```
SQL> create table Receta (  
  
  2 idReceta int NOT NULL,  
  
  3 nombre varchar2(50) NOT NULL,  
  
  4 preparacion varchar2(500) NOT NULL,  
  
  5 duración float,  
  
  6 comentarios varchar (200),  
  
  7 idTipo int,  
  
  8 CONSTRAINT pk_Receta PRIMARY KEY (idReceta),  
  
  9  CONSTRAINT fk_TipoDePlatillo FOREIGN KEY (idTipo) REFERENCES  
TipoDePlatillo(idTipo)  
  
 10 ).
```

Table created.

Necesitar_ingrediente

```
SQL> create table Necesitar_ingrediente(  
  2 idReceta int NOT NULL,  
  3 IdIngre1 int NOT NULL,  
  4 CONSTRAINT fk_Receta FOREIGN KEY (idReceta) REFERENCES Receta(idReceta),  
  5 CONSTRAINT fk_Ingrediente1 FOREIGN KEY (IdIngre1) REFERENCES Ingrediente(IdIngre)  
  6 );  
  
Table created.
```

```
SQL> create table Necesitar_ingrediente(  
  
  2 idReceta int NOT NULL,  
  
  3 IdIngre1 int NOT NULL,  
  
  4 CONSTRAINT fk_Receta FOREIGN KEY (idReceta) REFERENCES  
Receta(idReceta),  
  
  5 CONSTRAINT fk_Ingrediente1 FOREIGN KEY (IdIngre1) REFERENCES  
Ingrediente(IdIngre),  
  
  6 );  
  
Table created.
```

Caso #2

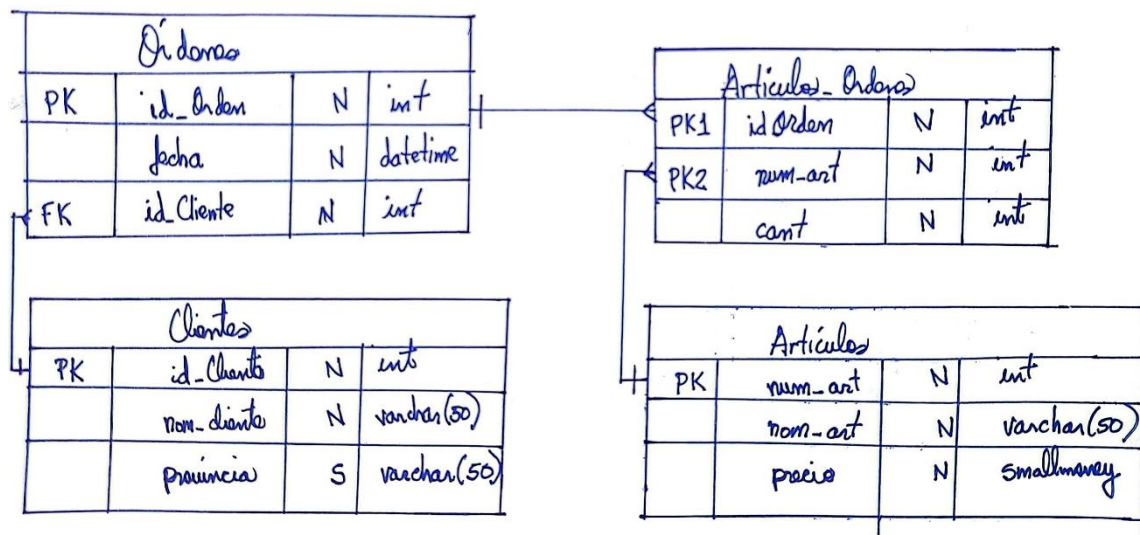
Indicaciones: Una compañía Deportiva mantiene una tabla de datos que no cumple con ninguna regla de normalización. Para el caso dado, aplicar cada una de las reglas normalización para encontrar el modelo correcto de base de datos, considerando los datos proporcionados por la compañía.

Adicionalmente realice la implementación correspondiente.

ordenes (id_orden, fecha, id_cliente, nom_cliente, estado, num_art, nom_art, cant, precio)

Id_orden	Fecha	Id_cliente	Nom_cliente	Provincia	Num_art	nom_art	cant	Precio
2301	23/02/2011	101	Martin	Chiriqui	3786	Red	3	35,00
2301	23/02/2011	101	Martin	Chiriqui	4011	Raqueta	6	65,00
2301	23/02/2011	101	Martin	Chiriqui	9132	Paq-3	8	4,75
2302	25/02/2011	107	Herman	Colon	5794	Paq-6	4	5,00
2303	27/02/2011	110	Pedro	Herrera	4011	Raqueta	2	65,00
2303	27/02/2011	110	Pedro	Herrera	3141	Funda	2	10,00

Modelo Físico



Implementación del Modelo Físico del Caso #2 en PL/SQL

Tabla Clientes

```
SQL> create table Clientes (  
  2  id_cliente int NOT NULL,  
  3  nom_cliente varchar2(50) NOT NULL,  
  4  provincia varchar2(50),  
  5  CONSTRAINT pk_Clientes PRIMARY KEY (id_cliente)  
  6 );  
  
Table created.
```

```
SQL> create table Clientes (  
  
  2  id_Cliente int NOT NULL,  
  
  3  nom_Cliente varchar2(50) NOT NULL,  
  
  4  provincia varchar2(50),  
  
  5  CONSTRAINT pk_Clientes PRIMARY KEY (id_Cliente)  
  
  6 );
```

Inserción de Datos en la Tabla Clientes

```
INSERT INTO clientes(id_cliente, nom_cliente, provincia)  
  
VALUES(101,'Martin','Chiriqui');  
  
INSERT INTO clientes(id_cliente, nom_cliente, provincia)  
  
VALUES(107,'Herman','Colon');  
  
INSERT INTO clientes(id_cliente, nom_cliente, provincia)  
  
VALUES(110,'Pedro','Herrera');
```



```

SQL> INSERT INTO clientes(id_cliente, nom_cliente, provincia)
  2  VALUES(101,'Martin','Chiriqui');

1 row created.

SQL> INSERT INTO clientes(id_cliente, nom_cliente, provincia)
  2  VALUES(107,'Herman','Colon');

1 row created.

SQL> INSERT INTO clientes(id_cliente, nom_cliente, provincia)
  2  VALUES(110,'Pedro','Herrera');

1 row created.

```

Tabla Articulos

```

SQL> create table Articulos (
  2  num_art int NOT NULL,
  3  nom_art varchar2(50) NOT NULL,
  4  precio float NOT NULL,
  5  CONSTRAINT pk_Articulos PRIMARY KEY (num_art)
  6 );

Table created.

```

```

SQL> create table Articulos (
  2  num_art int NOT NULL,
  3  nom_art varchar2(50) NOT NULL,
  4  precio float NOT NULL,
  5  CONSTRAINT pk_Articulos PRIMARY KEY (num_art)
  6 );

```

Inserción de Datos en la Tabla Artículos

```

INSERT INTO articulos(num_art, nom_art, precio)
VALUES(3786,'Red',35.00);

INSERT INTO articulos(num_art, nom_art, precio)

```

VALUES(4011,'Raqueta',65.00);

INSERT INTO articulos(num_art, nom_art, precio)

VALUES(9132,'Paq-3',4.75);

INSERT INTO articulos(num_art, nom_art, precio)

VALUES(5794,'Paq-6',5.00);

INSERT INTO articulos(num_art, nom_art, precio)

VALUES(3141,'Funda',10.00);

```
SQL> INSERT INTO articulos(num_art, nom_art, precio)
      2  VALUES(3786,'Red',35.00);

1 row created.

SQL> INSERT INTO articulos(num_art, nom_art, precio)
      2  VALUES(4011,'Raqueta',65.00);

1 row created.

SQL> INSERT INTO articulos(num_art, nom_art, precio)
      2  VALUES(9132,'Paq-3',4.75);

1 row created.

SQL> INSERT INTO articulos(num_art, nom_art, precio)
      2  VALUES(5794,'Paq-6',5.00);

1 row created.

SQL> INSERT INTO articulos(num_art, nom_art, precio)
      2  VALUES(3141,'Funda',10.00);

1 row created.
```

Tabla Ordenes

```
SQL> create table Ordenes (
      2  id_Orden int NOT NULL,
      3  fecha date NOT NULL,
      4  id_Cliente int NOT NULL,
      5  CONSTRAINT pk_Ordenes PRIMARY KEY (id_Orden),
      6  CONSTRAINT fk_Clientes FOREIGN KEY (id_Cliente) REFERENCES Clientes(id_Cliente)
      7  );

Table created.
```

```
SQL> create table Ordenes (
2 id_Orden int NOT NULL,
3 fecha date NOT NULL,
4 id_Cliente int NOT NULL,
5 CONSTRAINT pk_Ordenes PRIMARY KEY (id_Orden),
6 CONSTRAINT fk_Clientes FOREIGN KEY (id_Cliente) REFERENCES
Clientes(id_Cliente)
7 );
```

Inserción de Datos en la Tabla Ordenes

```
INSERT INTO ordenes(id_orden, fecha, id_cliente1)
VALUES(2301, TO_DATE('23/February/2011','DD/MM/YYYY'), 101);

INSERT INTO ordenes(id_orden, fecha, id_cliente1)
VALUES(2302, TO_DATE('25/February/2011','DD/MM/YYYY')
,107);

INSERT INTO ordenes(id_orden, fecha, id_cliente1)
VALUES(2303, TO_DATE('27/February/2011','DD/MM/YYYY'),110);
```

```
SQL> INSERT INTO Ordenes (id_Orden, fecha, id_Cliente)
2 VALUES(2301,TO_DATE('23/February/2011','DD/MM/YYYY'),101);
1 row created.

SQL> INSERT INTO Ordenes (id_Orden, fecha, id_Cliente)
2 VALUES(2302,TO_DATE('25/January/2011','DD/MM/YYYY'),107);
1 row created.

SQL> INSERT INTO Ordenes (id_Orden, fecha, id_Cliente)
2 VALUES(2303,TO_DATE('27/February/2011','DD/MM/YYYY'), 110);
1 row created.
```

Tabla Arts_ordenes

```
SQL> create table Arts_ordenes (  
2 id_Orden int NOT NULL,  
3 num_art int NOT NULL,  
4 cant int NOT NULL,  
5 CONSTRAINT fk_Ordenes FOREIGN KEY (id_Orden) REFERENCES Ordenes(id_Orden),  
6 CONSTRAINT fk_Articulos FOREIGN KEY (num_art) REFERENCES Articulos(num_art)  
7 );  
  
Table created.
```

```
SQL> create table Arts_ordenes (  
  
2 id_Orden int NOT NULL,  
  
3 num_art int NOT NULL,  
  
4 cant int NOT NULL,  
  
5 CONSTRAINT fk_Ordenes FOREIGN KEY (id_Orden) REFERENCES  
Ordenes(id_Orden),  
  
6 CONSTRAINT fk_Articulos FOREIGN KEY (num_art) REFERENCES  
Articulos(num_art)  
  
7 );
```

Inserción de Datos en la Tabla Arts_ordenes

```
INSERT INTO Arts_ordenes (id_orden, num_art1, cant)
```

```
VALUES(2301,3786,3);
```

```
INSERT INTO Arts_ordenes (id_orden, num_art1, cant)
```

```
VALUES(2302,4011,6);
```

```
INSERT INTO Arts_ordenes (id_orden, num_art1, cant)
```

```
VALUES(2303,9132,8);
```

```
INSERT INTO Arts_ordenes (id_orden, num_art1, cant)
```

```
VALUES(2302,5794,4);
```

INSERT INTO Arts_ordenes (id_orden, num_art1, cant)

VALUES(2303,4011,2);

INSERT INTO Arts_ordenes (id_orden, num_art1, cant)

VALUES(2303,3141,2);

```
SQL> INSERT INTO Arts_ordenes(id_orden, Num_art, cant)
2  VALUES(2301,3786,3);

1 row created.

SQL> INSERT INTO Arts_ordenes(id_orden, Num_art, cant)
2  VALUES(2301,4011,6);

1 row created.

SQL> INSERT INTO Arts_ordenes(id_orden, Num_art, cant)
2  VALUES(2301,9132,8);

1 row created.

SQL> INSERT INTO Arts_ordenes(id_orden, Num_art, cant)
2  VALUES(2302,5794,4);

1 row created.

SQL> INSERT INTO Arts_ordenes(id_orden, Num_art, cant)
2  VALUES(2303,4011,2);

1 row created.

SQL> INSERT INTO Arts_ordenes(id_orden, Num_art, cant)
2  VALUES(2303,3141,2);

1 row created.
```

Comentarios o Consideraciones Finales

Reto considerable fue la posibilidad de desarrollar los casos planteados en la Asignación #1 a través de ya una implementación física en la que se consideraron múltiples aspectos fundamentales de las cardinalidades de las Relaciones presentes entre las Tablas de una Base de Datos, en este caso; empero especificar que las mismas fueron planteadas con los debidos procedimientos y estándares previamente mencionados en clase, para su posterior implementación física en PL/SQL, desarrollado por la empresa Oracle Inc.