

**UNIVERSIDAD NACIONAL AUTÓNOMA DE HONDURAS
FACULTAD DE INGENIERÍA.**

DEPARTAMENTO DE INGENIERÍA EN SISTEMAS.

BASES DE DATOS II (IS-601)

PROYECTO PIZZERIA.

GRUPO “C”

LUCEM ASPICIO

Catedrático: Ing. Eduardo Gross.

Integrantes:

- | | | |
|------------------------------|------|-------------|
| ➤ Rocio Nataly Aguilar | 1847 | 20081000081 |
| ➤ Carlos Osman Suazo Baca | | 20101001701 |
| ➤ Nancy María Herrera Santos | | 20091002525 |
| ➤ Erick Alexander Zelaya | | 20092500066 |
| ➤ Ana Gabriela Gómez | | 20081001281 |

Sección: 14-00

Ciudad Universitaria, miércoles 11 de diciembre del 2013.

Introducción

El siguiente ejercicio pretende dar a conocer, como se realiza un análisis completo un de una Base de datos, esto mediante relaciones las cuales se representan por medio de tablas.

El objetivo principal de este trabajo, es hacer un análisis de una franquicia que se encarga de vender y ofrecer distintos productos a sus clientes, en donde ellos podrán decidir si lo desean consumirlo en el local, pedir a domicilio o si lo desean para llevar.

Se puede mencionar que en el análisis que se realizo se verá un grafico de relaciones entre tablas, cada uno con los atributos y tuplas que se necesitaría.

Gracias a lo anteriormente mencionado; esta práctica servirá de mucha ayuda a nosotros como estudiantes, manera que podamos comprender con claridad, como se debe diseñar una Base de Datos.

Objetivo General.

1. Realizar un análisis de un sistema de información que maneje El control de ventas y pedidos de una franquicia que realice ventas de pizza, bocadillos, y diferentes productos complementarios.

Objetivo Específico.

- Realizar un control de las pizzas que se han vendido.
- Almacenar la información de los pedidos y ventas que han realizado.
- Llevar un control de los servicios de pedidos que los repartidores realizan.
- Administrar la información sobre los repartos realizados por la empresa.

Modelo Conceptual.

COMIDAS A DOMICILIO

Enunciado del problema.

Eat'n Go, una empresa internacional dedicada a la comercialización de comidas rapidez, tiene la intención de ubicar una franquicia en nuestra localidad, siéndole necesario informatizar la gestión del negocio adaptándolo a las características propias de la clientela esperada, Esta empresa se dedica a la venta de pizzas y bocadillos, además de productos complementarios como refrescos, helados, etc.

Según las especificaciones aportadas por esta empresa se sabe que:

SUPUESTO 1: Tanto las pizzas como los bocadillos pueden condimentarse con un número de ingredientes de entre un conjunto de ellos con los que trabaja la empresa.

SUPUESTO 2: Los ingredientes con los que se hacen los bocadillos pueden ser iguales o distintos a aquellos con los que se hacen pizzas.

SUPUESTO 3: El número de ingredientes que intervienen en un artículo que se vende (pizza o bocadillo) no está delimitado, pudiendo realizarse una venta de estos artículos sin ningún ingrediente. Es decir, la empresa vende también las bases de las pizzas y el pan con los que prepara las pizzas y los bocadillos, respectivamente.

SUPUESTO 4: Cada artículo que vende la empresa (Pizza, bocadillos y productos complementarios) tienen un precio base asignado (El precio de estos productos se venden de forma independiente), sin contar los ingredientes que pueden acompañar a alguno de estos tipos de artículos.

SUPUESTO 5: Cada ingrediente tiene un precio para los bocadillos, mientras que para las pizzas todos los ingredientes tienen el mismo precio (Los productos complementarios no llevan ingredientes).

SUPUESTO 6: Los artículos se pueden vender en distinto tamaño, en cuyo caso el precio base es distinta según el tamaño, y el precio de los ingredientes también. Existen actualmente, tres tamaños en los que se venden los bocadillos y las pizzas (Pequeño, Mediano, Grande).

SUPUESTO 7: Las ventas se pueden hacer en tres formas diferentes: para consumir en el local, para recoger en el local y llevar o consumir en el mismo, y para servir a domicilio, en cuyo caso puede incrementarse un cargo añadido por parte de la venta.

SUPUESTO 8: Los artículos complementarios que vende la empresa tienen un precio fijo en base a su tipo, tamaño, sabor, etc.

SUPUESTO 9: Los clientes pueden solicitar un servicio de la empresa (un pedido) tanto personalmente en el local como telefónicamente.

SUPUESTO 10: En los pedidos telefónicos se tomarán los datos completos del cliente: en los de recoger solo el documento nacional de identidad y su nombre completo, y en los de consumir en el local no se tomará ningún dato a no ser que sea necesario por otras razones.

SUPUESTO 11: A la empresa le interesa mantener información de todos los clientes a los que se les recoge información por dos razones:

1. Simplemente a escala informática, de consumo y marketing.
2. Para llevar el control del consumo y favorecer, mediante obsequios, a aquellos clientes que alcancen un cierto consumo

SUPUESTO 12: La empresa organiza, a veces, promociones para sus clientes. Estas se basan en el obsequio de algún regalo o artículo de propaganda sobre la base de:

1. Las unidades consumidas de alguno de los tipos de artículos: pizzas o bocadillos
2. Cargo total alcanzado en los pedidos.

SUPUESTO 12+1: Cuando la empresa obsequia a los clientes con algún artículo de promoción siempre recaba de los mismos toda su información y, además, mantiene información sobre los regalos de promoción que se le entrega para, a ser posible, no hacer entregas duplicadas de los mismos.

SUPUESTO 14: La empresa cuenta con una serie de repartidores encargados del reparto a domicilio y de una batería de scooters para ayudarles con su cometido. Cada reparto se le asigna a un repartidor, el cual tiene asignado un scooter para ser utilizado por distintos repartidores (por su puesto en distintos turnos de trabajo).

SUPUESTO 15: A la empresa le interesa conocer información del coste de los scooters sobre la base del consumo de gasolina de los mismos en la relación de los repartos.

SUPUESTO 16: Los precios de todos los artículos que vende la empresa tienen IVA incluido.

SUPUESTO 17: En los pedidos a domicilio existe un mínimo, en lo referente al valor del pedido, para que este sea servido. Si el valor de los artículos que componen el pedido no alcanza este mínimo y el cliente desea que se le sirva a domicilio, se le cobrará ese mínimo.

SUPUESTO 18: La empresa cuenta con una serie de pizzas y bocadillos “estrellas”, los cuales están formados por un conjunto de ingredientes predeterminados. Estos acumulando los ingredientes que incorporan.

Definición de Objetos.

- **PIZZA:** representa un objeto del mundo real, que representa un producto que vende una franquicia.
- **BOCADILLOS:** representa un objeto del mundo real, que simboliza un producto que vende una franquicia, El cual contiene un número de ingredientes.
- **INGREDIENTES:** Representa a un objeto del mundo real, y que se usan para la elaboración de pizzas y bocadillos.
- **PRODUCTOS_COMPLEMENTARIO:** representa a un objeto del mundo real, el presenta los diferentes productos complementarios que el local ofrece a los clientes.
- **VENTAS:** representa a un objeto del mundo real, el cual son los diferentes productos que ofrece a sus clientes, donde dichas ventas pueden ser ventas a domicilio, para llevar o para consumir en la franquicia.
- **CLIENTE:** representa a un objeto del mundo real, el cual almacena los datos generales de la persona que hace un pedido a la franquicia.
- **PEDIDO:** representa a un objeto del mundo real, el cual almacena el servicio que solicita un cliente.
- **PROMOCIONES:** Objeto del mundo real, que representa un regalo que ofrece la franquicia a sus clientes.
- **MOTOCICLETAS:** Objeto del mundo real, que representa un medio de transporte que es sirve para que los repartidores entreguen los productos a sus clientes.
- **REPARTIDOR:** Objeto del mundo real, que representa una persona que se encarga de repartir o entregar los artículos que el cliente pide a domicilio.
- **SCOOTER:** Objeto del mundo real, que representa un medio de transporte que es sirve para que los repartidores entreguen los productos a sus clientes.
- **REGALO:** Representa un objeto del mundo real y presenta una variedad de regalo que el local ofrece a clientes seleccionados.

Tablas relacionales: intermedias.

- **ARTICULO_PEDIDO:** Esta tabla funciona como la interrelacion entre articulo y pedido y posee los siguientes atributos: ID_ARTICULO, ID_PEDIDO.
- **CLIENTE_REGALO:** Esta tabla funciona como la interrelacion entre cliente y regalo y posee los siguientes atributos: ID_CLIENTE, ID_REGALO.
- **PROMO_REGALO:** Esta tabla funciona como la interrelacion entre promocion y regalo y posee los siguientes atributos: ID_PROMOCION, ID_REGALO.
- **PROMO_ARTICULO:** Esta tabla funciona como la interrelacion entre promocion y articulo y posee los siguientes atributos: ID_PROMOCION, ID_ARTICULO.
- **ARTICULO_VENTA:** Esta tabla funciona como la interrelacion entre venta y articulo y posee los siguientes atributos: ID_VENTA, ID_ARTICULO, CANTIDAD.
- **VENTA_DOMICILIO:** Esta tabla funciona como la interrelacion entre la tabla VENTA y PEDIDO posee los siguientes atributos: ID_VENTA, ID_PEDIDO.
- **REPAR_SCO_ENT:** Esta tabla funciona como la interrelacion entre REPARTIDOR, SCOOTER y ENTREGA posee los siguientes atributos: ID_REPARTIDOR, ID_SCOOTER, ID_ENTREGA.

DIFINICION DE RELACIONES

REPARTIDOR (**ID_REPARTIDOR**, NOMBRE_REPARTIDOR)

PEDIDO (**ID_PEDIDO**, FECHA_PEDIDO, **ID_REPARTIDOR**, **ID_CLIENTE**)

SCOOTER (**ID_SCOOTER**, PRECIO)

ENTREGA (**ID_ENTREGA**, CRONOMETRO, FECHA_ENTREGA)

REPAR_SCO_ENTE (**ID_REPARTIDOR**, **ID_SCOOTER**, **ID_ENTREGA**,)

CLIENTE (**ID_CLIENTE**, CELULAR, PRIMER_NOMBRE, PRIMER_APELLIDO, COLONIA)

VENTA (**ID_VENTA**, TIPO_VENTA, FECHA_VENTA, MONTO)

VENTA_DOMICILIO (**ID_VENTA**, **ID_PEDIDO**)

INGREDIENTE (**ID_INGREDIENTE**, NOMBRE_INGREDIENTE, PRECIO_PIZZA, PRECIO_BOCADILLOS)

ARTICULO (**ID ARTICULO**, TIPO_ARTICULO)

INGREDIENTE_ARTICULO (**ID ARTICULO**, NOMBRE_INGREDIENTE)

ARTICULO_VENTA (**ID ARTICULO**, **ID VENTA**, CANTIDAD)

PIZZA (**ID ARTICULO**, NOMBRE, TAMAÑO, PRECIO)

BOCADILLO (**ID ARTICULO**, NOMBRE, TAMAÑO, PRECIO)

COMPLEMENTO (**ID ARTICULO**, NOMBRE, TAMAÑO, PRECIO)

PRODUCTO_ESTRELLA (**ID ARTICULO**, NOMBREPE)

PROMOCION (**ID PROMOCION**, NOMBRE_PROMOCION, FECHA_INICIO, FECHA_FIN)

REGALO (**ID REGALO**, NOMBRE_REGALO)

PROMOCION_ARTICULO (**ID PROMOCION**, **ID ARTICULO**)

PROMOCION_REGALO (**ID PROMOCION**, **ID REGALO**)

CLIENTE_REGALO (**ID CLIENTE**, **ID REGALO**.)

ARTICULO_PEDIDO (**ID ARTICULO**, **ID PEDIDO**, CANTIDAD)

Diagrama Entidad-Relación.

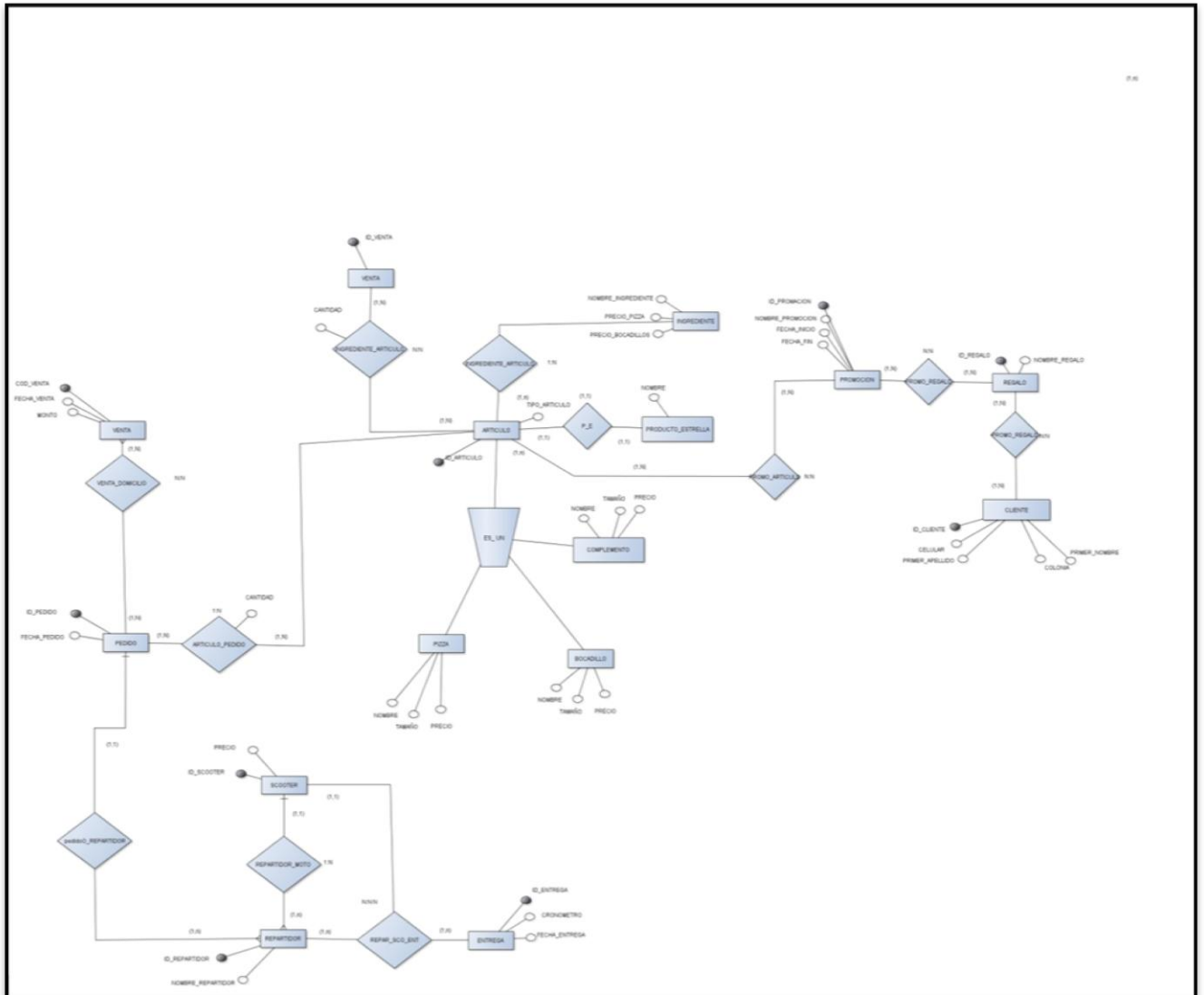
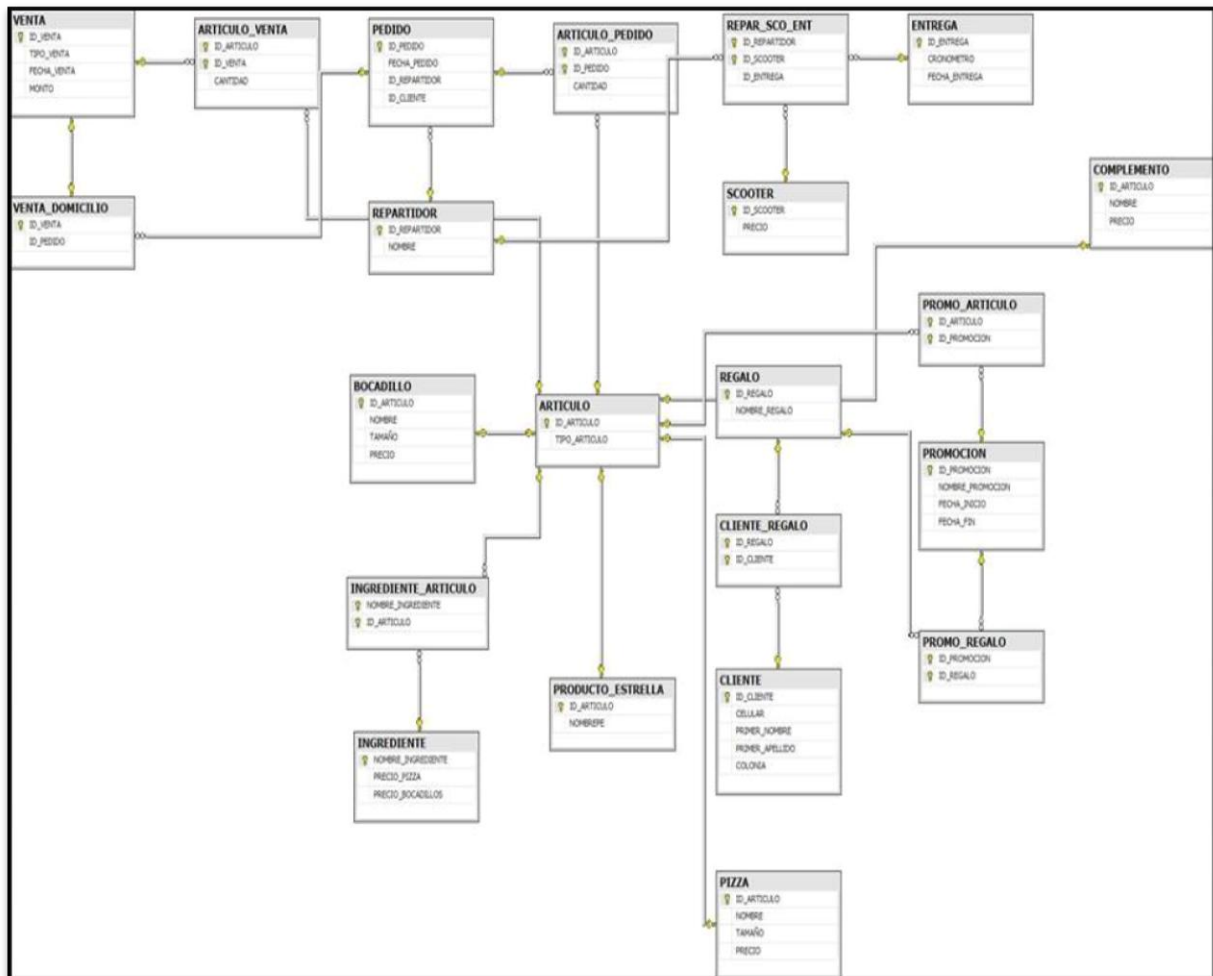


Diagrama Relacional.



Diccionario de datos.

Nombre Entidad	Atributo	Tipo	Longitud	Nulo	Descripción	Ejemplo.
REPARTIDOR	ID_REPARTIDOR	INT		NOT NULL	Almacena el código del repartidor, también es clave primaria de la tabla.	01
	NOMBRE	VARCHAR	15	NOT NULL	Almacena el nombre del repartidor	juan
	PK_R	CONSTRAINT			Constructor de la clave primaria	
PEDIDO	ID_PEDIDO	INT		NOT NULL	Almacena el código del pedido, también es clave primaria de la tabla.	01
	FECHA_PEDIDO	DATETIME		NOT NULL	Almacena la fecha que se hace el pedido	12/5/2010
	ID_REPARTIDOR	INT		NOT NULL	Almacena el código del repartidor, que hace referencia a la tabla repartidor.	05
	ID_CLIENTE	VARCHAR	20	NOT NULL	Almacena el código del cliente, que hace referencia a la tabla cliente.	05
	PK_PEDIDO	CONSTRAINT			Constructor de la clave primaria	
	FK_RP	CONSTRAINT			Constructor de la llave foránea (ID_REPARTIDOR)	
SCOOTER	ID_SCOOTER	VARCHAR	6	NOT NULL	Almacena el código del scooter, también es clave primaria de la tabla.	023
	PRECIO	FLOAT		NOT NULL	Almacena el precio del scooter.	12.5
	PK_SCOOTER	CONSTRAINT			Constructor de la clave primaria	
	CK_PRECIO	CONSTRAINT			Constructor para la condición (PRECIO > 0)	
ENTREGA	ID_ENTREGA	INT		NOT NULL	Almacena el código de la entrega, también es clave primaria de la tabla.	25
	CRONOMETRO	TIME		NOT NULL	Almacena el tiempo de la entrega	15:50
	FECHA_ENTREGA	DATETIME		NOT NULL	Almacena la fecha de la entrega	05/05/2002

Nombre Entidad	Atributo	Tipo	Longitud	Nulo	Descripción	Ejemplo.
	PK_ENTREGA	CONSTRAINT			Constructor de la clave primaria	
REPAR_SCO_ENT	ID_REPARTIDOR	INT			Almacena el código del repartidor, que hace referencia a la tabla repartidor.	
	ID_SCOOTER	VARCHAR	6		Almacena el código del scooter, que hace referencia a la tabla scooter.	
	ID_ENTREGA	INT			Almacena el código de la entrega, que hace referencia a la tabla entrega.	
	PK_RS	CONSTRAINT			Constructor de la clave primaria	
	FK_RSR	CONSTRAINT			Constructor de la llave foránea (ID_REPARTIDOR)	
	FK_RES	CONSTRAINT			Constructor de la llave foránea (ID_SCOOTER)	
	FK_SENTREGA	CONSTRAINT			Constructor de la llave foránea (ID_ENTREGA)	
CLIENTE	ID_CLIENTE	VARCHAR	20	Not null	Almacena el código del cliente	001
	CELULAR	VARCHAR	8		Almacena el numero de celular del cliente	3389751
	PRIMER_NOMBRE	VARCHAR	15		Almacena el nombre del cliente	Juan
	PRIMER_APELLIDO	VARCHAR	15		Almacena el apellido del cliente	Pérez
	COLONIA	VARCHAR	15		Almacena la dirección de la colonia donde vive del cliente	Col Florencia
	PK_CLIENTE	CONSTRAINT			Constructor de la clave primaria	
VENTA	ID_VENTA	INT		NOT NULL	Almacena el identificador de las ventas y es llave primaria	0010
	TIPO_VENTA	VARCHAR	15	NOT NULL	Almacena el tipo de la venta	En el local
	FECHA_VENTA	DATETIME		NOT NULL	Almacena la fecha de la venta	8/8/2010
	MONTO	FLOAT			Almacena el total de venta	150.20
	PK_VENTA	CONSTRAINT			Constructor de llave primaria	

Nombre Entidad	Atributo	Tipo	Longitud	Nulo	Descripción	Ejemplo.
	CK_TIPOV	CONSTRAINT			Constructor para la condición CHECK (TIPO_VENTA IN ('A DOMICILIO', 'PARA LLEVAR', 'EN EL LOCAL'))	
VENTA_DOMICILIO	ID_VENTA	INT		NOT NULL	Almacena el código de la venta, que hace referencia a la tabla venta	002
	ID_PEDIDO	INT		NOT NULL	Almacena el código del pedido, que hace referencia a la tabla pedido.	005
	PK_CVD	CONSTRAINT			Constructor de la clave primaria	
	FK_DOMI_VENTA	CONSTRAINT			Constructor de la llave foránea (ID_VENTA)	
	FK_VD_PEDIDO	CONSTRAINT			Constructor de la llave foránea ID_PEDIDO	
VENTA_DOMICILIO	ID_VENTA	INT		NOT NULL	Almacena el código de la venta, que hace referencia a la tabla venta	002
	ID_PEDIDO	INT		NOT NULL	Almacena el código del pedido, que hace referencia a la tabla pedido.	005
	PK_CVD	CONSTRAINT			Constructor de la clave primaria	
	FK_DOMI_VENTA	CONSTRAINT			Constructor de la llave foránea (ID_VENTA)	
	FK_VD_PEDIDO	CONSTRAINT			Constructor de la llave foránea ID_PEDIDO	
INGREDIENTE	NOMBRE_INGREDIENTE	VARCHAR	20		Almacena el nombre de los ingredientes	especias
	PRECIO_PIZZA	FLOAT			Almacena el precio de los ingredientes_pizza	12.00
	PRECIO_BOCADILLOS	FLOAT			Almacena el precio de los ingredientes_bocadillos	10.00
	PK_INGREDIENTE	CONSTRAINT			Constructor de la clave primaria	
ARTICULO	ID_ARTICULO	INT		NOT NULL	Almacena el código del articulo	001
	PK_ART	CONSTRAINT			Constructor de la clave primaria	

Nombre Entidad	Atributo	Tipo	Longitud	Nulo	Descripción	Ejemplo.
	CK_TA	CONSTRAINT			Constructor de condición: <code>CHECK(TIPO_ARTICULO IN ('PIZZA', 'BOCADILLOS', 'COMPLEMENTO'))</code>	
INGREDIENTE_ARTICULO	NOMBRE_INGREDIENTE	VARCHAR	20		Almacena el nombre de los ingredientes	peperoni
	ID_ARTICULO	INT			Almacena el código del artículo, que hace referencia a la tabla articulo.	012
	PK_IXV	CONSTRAINT			Constructor de la clave primaria	
	FK_IVXV	CONSTRAINT			Constructor de la llave foránea de la tabla INGREDIENTE (NOMBRE_INGREDIENTE)	
	FK_IVXI	CONSTRAINT			Constructor de la llave foránea de la tabla ARTICULO (ID_ARTICULO)	
ARTICULO_VENTA	ID_ARTICULO	INT			Almacena el código del artículo, que hace referencia a la tabla articulo.	017
	ID_VENTA	INT			Almacena el código de la venta, que hace referencia a la tabla venta	
	CANTIDAD	INT			Almacena la cantidad de ese artículo que se vende.	2
	PK_AV	CONSTRAINT			Constructor de la clave primaria	
	PK_AVXA	CONSTRAINT			Constructor de la llave foránea de la tabla INGREDIENTE (NOMBRE_INGREDIENTE)	
	PK_AVXV	CONSTRAINT			Constructor de la llave foránea de la tabla ARTICULO (ID_ARTICULO)	
PIZZA	ID_ARTICULO	INT		NOT NULL	Almacena el código del artículo, que hace referencia a la tabla articulo.	10
	NOMBRE	VARCHAR	30		almacena el nombre de la pizza	Pizza con jamón y queso
	TAMAÑO	VARCHAR	10		almacena el tamaño de la pizza	grande
	PRECIO	FLOAT			almacena el precio de la pizza	189.00

Nombre Entidad	Atributo	Tipo	Longitud	Nulo	Descripción	Ejemplo.
	PK_PIZZA	CONSTRAINT			Constructor de la clave primaria	
	FK_AP	CONSTRAINT			Constructor de la llave foránea de la tabla ARTICULO (ID_ARTICULO)	
	CK_TAMAÑO	CONSTRAINT			Constructor de condición: CHECK (TAMAÑO IN ('GRANDE', 'MEDIANO', 'PEQUEÑO'))	
BOCADILLO	ID_ARTICULO	INT		NOT NULL	Almacena el código del artículo, que hace referencia a la tabla articulo.	050
	NOMBRE	VARCHAR	25		almacena el nombre del bocadillo	palitroques
	TAMAÑO	VARCHAR	10		almacena el tamaño del bocadillo	mediano
	PRECIO	FLOAT			almacena el precio del bocadillo	52.02
	PK_BOCADILLO	CONSTRAINT			Constructor de la clave primaria	
	FK_AB	CONSTRAINT			Constructor de la llave foránea de la tabla ARTICULO (ID_ARTICULO)	
	CK_TAMAÑO B	CONSTRAINT			Constructor de condición: CHECK (TAMAÑO IN ('GRANDE', 'MEDIANO', 'PEQUEÑO'))	
COMPLEMENTO	ID_ARTICULO	INT		NOT NULL	Almacena el código del artículo, que hace referencia a la tabla articulo.	018
	NOMBRE	VARCHAR	20	NOT NULL	almacena el nombre del complemento	alitas
	PRECIO	FLOAT		NOT NULL	almacena el precio del complemento	40.00
	PK_COMPLEMENTO	CONSTRAINT			Constructor de la clave primaria	
	FK_AC	CONSTRAINT			Constructor de la llave foránea de la tabla ARTICULO (ID_ARTICULO)	
PRODUCTO_ESTRELLA	ID_ARTICULO	INT		NOT NULL	Almacena el código del artículo, que hace referencia a la tabla articulo.	005
	NOMBRE PE	VARCHAR	25	NOT NULL	almacena el nombre del producto estrella	PE1

Nombre Entidad	Atributo	Tipo	Longitud	Nulo	Descripción	Ejemplo.
	PK_PE	CONSTRAINT			Constructor de la clave primaria	
PROMOCION	ID_PROMOCION	INT		NOT NULL	Almacena el identificador de una promoción	025
	NOMBRE_PROMOCION	VARCHAR	30		Almacena el nombre de la promoción	
	FECHA_INICIO	DATE			Almacena la fecha de inicio de la promoción	20/10/2010
	FECHA_FIN	DATE			Almacena la fecha final de la promoción	10/1/2011
	PK_PROMOCION	CONSTRAINT			Constructor de la llave primaria	
REGALO	ID_REGALO	INT		NOT NULL	Almacena el identificador de un regalo	01
	NOMBRE_REGALO	VARCHAR	30		Almacena el nombre de un regalo	peluche
	PK_REGALO	CONSTRAINT			Constructor de la llave primaria	
PROMO_ARTICULO	ID_ARTICULO	INT		NOT NULL	Almacena el código del artículo, que hace referencia a la tabla articulo.	1
	ID_PROMOCION	INT		NOT NULL	Almacena el código de la promoción, que hace referencia a la tabla promoción.	1
	PK_PROMO_ART	CONSTRAINT			Constructor de la llave primaria	2
	FK_PROMO_ART_PROMO	CONSTRAINT			Constructor de la llave foránea de la tabla PROMOCION (ID_PROMOCION)	
	FK_PROMO_ART_ART	CONSTRAINT			Constructor de la llave foránea de la tabla ARTICULO (ID_ARTICULO)	
PROMO_REGALO	ID_PROMOCION	INT		NOT NULL	Almacena el código de la promoción, que hace referencia a la tabla promoción.	1
	ID_REGALO	INT		NOT NULL	Almacena el código del regalo, que hace referencia a la tabla regalo.	2
	PK_PROMO_REGALO	CONSTRAINT			Constructor de la llave primaria	

Nombre Entidad	Atributo	Tipo	Longitud	Nulo	Descripción	Ejemplo.
	FK_ID_PROM	CONSTRAINT			Constructor de la llave foránea de la tabla: PROMOCION (ID_PROMOCION)	
	FK_REGALO	CONSTRAINT			Constructor de la llave foránea de la tabla: REGALO (ID_REGALO)	
CLIENTE_REGALO	ID_REGALO	INT		NOT NULL	Almacena el código del regalo, que hace referencia a la tabla regalo.	1
	ID_CLIENTE	VARCHAR	20	NOT NULL	Almacena el código del cliente, que hace referencia a la tabla cliente.	0401-2000-56584
	PK_CLI_REG	CONSTRAINT			Constructor de la llave primaria	
	FK_CLI_REG_CLI	CONSTRAINT			Constructor de la llave foránea de la tabla: CLIENTE (ID_CLIENTE)	
	FK_CLI_REG_REG	CONSTRAINT			Constructor de la llave foránea de la tabla: REGALO (ID_REGALO)	
ARTICULO_PEDIDO	ID_ARTICULO	INT		NOT NULL	Almacena el código del artículo, que hace referencia a la tabla articulo.	3
	ID_PEDIDO	INT		NOT NULL	Almacena el código del pedido, que hace referencia a la tabla pedido.	1
	CANTIDAD	INT			Almacena la cantidad del artículo que se pidió.	5
	PK_ART_PED	CONSTRAINT			Constructor de la llave primaria	
	FK_ART_PED_ART	CONSTRAINT			Constructor de la llave foránea de la tabla ARTICULO (ID_ARTICULO)	
	FK_ART_PED_PED	CONSTRAINT			Constructor de la llave foránea de la tabla PEDIDO (ID_PEDIDO)	

ELABORACION DE BASE DE DATOS DDL Y DML.

```
CREATE DATABASE PIZZERIA
USE PIZZERIA
```

```
CREATE TABLE REPARTIDOR(
    ID_REPARTIDOR INT IDENTITY (1,1) NOT NULL,
    NOMBRE VARCHAR(15) NOT NULL,
    CONSTRAINT PK_R PRIMARY KEY(ID_REPARTIDOR)
)
```

```
INSERT INTO REPARTIDOR VALUES ('ERICK')
INSERT INTO REPARTIDOR VALUES ('CARLOS')
INSERT INTO REPARTIDOR VALUES ('ANA')
INSERT INTO REPARTIDOR VALUES ('FEDERICO')
INSERT INTO REPARTIDOR VALUES ('ELSA PO')
```

```
-----
CREATE TABLE PEDIDO
(
    ID_PEDIDO INT IDENTITY(1,1) NOT NULL,
    FECHA_PEDIDO DATETIME NOT NULL,
    ID_REPARTIDOR INT NOT NULL,
    ID_CLIENTE VARCHAR(20) NOT NULL,
    CONSTRAINT PK_PEDIDO
    PRIMARY KEY(ID_PEDIDO),
    CONSTRAINT FK_RP
    FOREIGN KEY(ID_REPARTIDOR)
    REFERENCES REPARTIDOR(ID_REPARTIDOR) ON DELETE CASCADE ON UPDATE
    CASCADE
)
```

```
INSERT INTO PEDIDO VALUES('12-10-2013', 1, '0703-1992-00523')
INSERT INTO PEDIDO VALUES('15-10-2013', 2, '0703-1992-00523')
INSERT INTO PEDIDO VALUES('18-10-2013', 3, '0703-1994-00421')
INSERT INTO PEDIDO VALUES('20-10-2013', 4, '0703-1994-00421')
INSERT INTO PEDIDO VALUES('22-10-2013', 1, '0703-1994-00421')
INSERT INTO PEDIDO VALUES('25-10-2013', 3, '0703-1994-00421')
INSERT INTO PEDIDO VALUES('31-10-2013', 2, '0503-1999-00524')
INSERT INTO PEDIDO VALUES('31-10-2013', 4, '0503-1999-00524')
INSERT INTO PEDIDO VALUES('5-11-2013', 5, '0801-1991-02542')
INSERT INTO PEDIDO VALUES('7-11-2013', 5, '0801-1991-02542')
INSERT INTO PEDIDO VALUES('10-11-2013', 4, '0801-1991-02542')
```

```
-----
CREATE TABLE SCOOTER
(
    ID_SCOOTER VARCHAR(6) NOT NULL,
    PRECIO FLOAT NOT NULL,

    CONSTRAINT PK_SCOOTER
    PRIMARY KEY(ID_SCOOTER),
    CONSTRAINT CK_PRECIO
    CHECK(PRECIO > 0)
)
```

```

INSERT INTO SCOOTER VALUES ('ALER', 12150)
INSERT INTO SCOOTER VALUES ('ASDF', 12051)
INSERT INTO SCOOTER VALUES ('ZXC5', 13050)
INSERT INTO SCOOTER VALUES ('QWE65', 13225)
INSERT INTO SCOOTER VALUES ('94EQS', 120.00)

```

```

-----
CREATE TABLE ENTREGA
(
    ID_ENTREGA INT IDENTITY(1,1) NOT NULL,
    CRONOMETRO TIME NOT NULL,
    FECHA_ENTREGA DATETIME NOT NULL,

    CONSTRAINT PK_ENTREGA PRIMARY KEY(ID_ENTREGA)
)

```

```

INSERT INTO ENTREGA VALUES ('0:20', '12-10-2013')
INSERT INTO ENTREGA VALUES ('0:19', '15-10-2013')
INSERT INTO ENTREGA VALUES ('0:25', '18-10-2013')
INSERT INTO ENTREGA VALUES ('0:24', '20-10-2013')
INSERT INTO ENTREGA VALUES ('0:26', '22-10-2013')
INSERT INTO ENTREGA VALUES ('0:25', '25-10-2013')
INSERT INTO ENTREGA VALUES ('0:30', '31-10-2013')
INSERT INTO ENTREGA VALUES ('0:21', '31-10-2013')
INSERT INTO ENTREGA VALUES ('0:20', '5-11-2013')
INSERT INTO ENTREGA VALUES ('0:26', '7-11-2013')
INSERT INTO ENTREGA VALUES ('0:25', '10-1-2013')

```

```

-----
CREATE TABLE REPAR_SCO_ENT
(
    ID_REPARTIDOR INT,
    ID_SCOOTER VARCHAR(6),
    ID_ENTREGA INT,

    CONSTRAINT PK_RS
    PRIMARY KEY(ID_REPARTIDOR, ID_SCOOTER),
    CONSTRAINT FK_RSR FOREIGN KEY(ID_REPARTIDOR) REFERENCES
REPARTIDOR(ID_REPARTIDOR) ON DELETE CASCADE ON UPDATE CASCADE,
    CONSTRAINT FK_RES FOREIGN KEY(ID_SCOOTER) REFERENCES
SCOOTER(ID_SCOOTER) ON DELETE CASCADE ON UPDATE CASCADE,
    CONSTRAINT FK_SENTREGA FOREIGN KEY(ID_ENTREGA) REFERENCES
ENTREGA(ID_ENTREGA) ON DELETE CASCADE ON UPDATE CASCADE,
)

```

```

INSERT INTO REPAR_SCO_ENT VALUES (1, 'ALER', 1)
INSERT INTO REPAR_SCO_ENT VALUES (5, 'ALER', 3)
INSERT INTO REPAR_SCO_ENT VALUES (3, 'ALER', 5)
INSERT INTO REPAR_SCO_ENT VALUES (4, 'ASDF', 7)
INSERT INTO REPAR_SCO_ENT VALUES (2, 'ASDF', 8)
INSERT INTO REPAR_SCO_ENT VALUES (3, 'ASDF', 6)
INSERT INTO REPAR_SCO_ENT VALUES (1, 'ZXC5', 2)
INSERT INTO REPAR_SCO_ENT VALUES (5, 'ZXC5', 4)
INSERT INTO REPAR_SCO_ENT VALUES (2, 'ZXC5', 9)
INSERT INTO REPAR_SCO_ENT VALUES (4, 'QWE65', 11)
INSERT INTO REPAR_SCO_ENT VALUES (4, '94EQS', 10)

```

```

-----
CREATE TABLE CLIENTE
(
    ID_CLIENTE VARCHAR(20) NOT NULL,
    CELULAR VARCHAR(8),
    PRIMER_NOMBRE VARCHAR(15) NOT NULL,
    PRIMER_APELLIDO VARCHAR(15) NOT NULL,
    COLONIA VARCHAR(15),

    CONSTRAINT PK_CLIENTE
    PRIMARY KEY (ID_CLIENTE)

)

INSERT INTO CLIENTE VALUES ('0703-1992-00523', '32028853', 'STEPHANIE',
'ROMERO', 'LAS COLINAS')
INSERT INTO CLIENTE VALUES ('0801-1991-02542', '32634981', 'IRIS', 'HERRERA',
'LAS COLINAS')
INSERT INTO CLIENTE VALUES ('0503-1999-00524', '93127411', 'GLADYS',
'TORRES', 'LOS GIRASOLES')
INSERT INTO CLIENTE VALUES ('0401-2000-56584', '99813271', 'ARMANDO',
'PAREDES', 'LA KENNEDY')
INSERT INTO CLIENTE VALUES ('0903-1980-54867', '22270264', 'RODRIGO',
'RODRIGUEZ', 'LAS UVAS')
INSERT INTO CLIENTE VALUES ('0109-1987-25467', '22270910', 'MARYURY',
'ROMERO', 'LAS JOYAS')
INSERT INTO CLIENTE VALUES ('0703-1994-00421', '99490988', 'VALERIA',
'ROMERO', 'LA MODESTO')

```

```

-----
CREATE TABLE VENTA(
    ID_VENTA INT IDENTITY(1,1) NOT NULL,
    TIPO_VENTA VARCHAR(15) NOT NULL,
    FECHA_VENTA DATETIME NOT NULL,
    MONTO FLOAT,
    CONSTRAINT PK_VENTA
    PRIMARY KEY (ID_VENTA),
    CONSTRAINT CK_TIPOV
    CHECK (TIPO_VENTA IN ('A DOMICILIO', 'PARA LLEVAR', 'EN EL LOCAL'))

)

INSERT INTO VENTA VALUES ('A DOMICILIO', '2-10-2013', 900) --1
INSERT INTO VENTA VALUES ('PARA LLEVAR', '9-10-2013', 100)
INSERT INTO VENTA VALUES ('EN EL LOCAL', '14-10-2013', 200)
INSERT INTO VENTA VALUES ('PARA LLEVAR', '20-10-2013', 300)
INSERT INTO VENTA VALUES ('A DOMICILIO', '22-10-2013', 400) --5
INSERT INTO VENTA VALUES ('EN EL LOCAL', '25-10-2013', 540)
INSERT INTO VENTA VALUES ('EN EL LOCAL', '31-10-2013', 600)
INSERT INTO VENTA VALUES ('PARA LLEVAR', '5-11-2013', 700)
INSERT INTO VENTA VALUES ('A DOMICILIO', '5-11-2013', 800) --9
INSERT INTO VENTA VALUES ('A DOMICILIO', '5-11-2013', 750) --10

```

```

-----
CREATE TABLE VENTA_DOMICILIO(

```

```

        ID_VENTA INT NOT NULL,
        ID_PEDIDO INT NOT NULL,
        CONSTRAINT PK_CVD
        PRIMARY KEY (ID_VENTA),
        CONSTRAINT FK_DOMI_VENTA
        FOREIGN KEY (ID_VENTA)
        REFERENCES VENTA (ID_VENTA) ON DELETE CASCADE ON UPDATE CASCADE,
        CONSTRAINT FK_VD_PEDIDO FOREIGN KEY (ID_PEDIDO) REFERENCES
PEDIDO (ID_PEDIDO)
)
INSERT INTO VENTA_DOMICILIO VALUES (1,1)
INSERT INTO VENTA_DOMICILIO VALUES (2,1)
INSERT INTO VENTA_DOMICILIO VALUES (4,2)
INSERT INTO VENTA_DOMICILIO VALUES (3,3)

```

```

CREATE TABLE INGREDIENTE (
    NOMBRE_INGREDIENTE VARCHAR(20) NOT NULL,
    PRECIO_PIZZA FLOAT NOT NULL,
    PRECIO_BOCADILLOS FLOAT NOT NULL,
    CONSTRAINT PK_INGREDIENTE PRIMARY KEY (NOMBRE_INGREDIENTE)
)

```

```

INSERT INTO INGREDIENTE VALUES ('CHORIZO', 31.50, 30.00)
INSERT INTO INGREDIENTE VALUES ('PEPPERONI', 54.00, 30.00)
INSERT INTO INGREDIENTE VALUES ('ANCHOAS', 29.50, 30.00)
INSERT INTO INGREDIENTE VALUES ('POLLO', 45.50, 30.00)
INSERT INTO INGREDIENTE VALUES ('SOYA', 22.50, 30.00)
INSERT INTO INGREDIENTE VALUES ('PIMIENTO', 20.00, 30.00)
INSERT INTO INGREDIENTE VALUES ('TOCINO', 45.00, 30.00)
INSERT INTO INGREDIENTE VALUES ('QUESO MOZARELLA', 42.00, 30.00)
INSERT INTO INGREDIENTE VALUES ('PIÑA', 24.00, 30.00)
INSERT INTO INGREDIENTE VALUES ('HONGOS', 41.50, 30.00)

```

```

CREATE TABLE ARTICULO (
    ID_ARTICULO INT IDENTITY(1,1),
    TIPO_ARTICULO VARCHAR(15) NOT NULL,

    CONSTRAINT PK_ART PRIMARY KEY (ID_ARTICULO),
    CONSTRAINT CK_TA CHECK (TIPO_ARTICULO IN
('PIZZA', 'BOCADILLOS', 'COMPLEMENTO'))
)
INSERT INTO ARTICULO VALUES ('PIZZA')
INSERT INTO ARTICULO VALUES ('PIZZA')
INSERT INTO ARTICULO VALUES ('BOCADILLOS')
INSERT INTO ARTICULO VALUES ('COMPLEMENTO')
INSERT INTO ARTICULO VALUES ('COMPLEMENTO')
INSERT INTO ARTICULO VALUES ('BOCADILLOS')
INSERT INTO ARTICULO VALUES ('BOCADILLOS')
INSERT INTO ARTICULO VALUES ('COMPLEMENTO')
INSERT INTO ARTICULO VALUES ('PIZZA')
INSERT INTO ARTICULO VALUES ('COMPLEMENTO')
INSERT INTO ARTICULO VALUES ('PIZZA')
INSERT INTO ARTICULO VALUES ('BOCADILLOS')

```

```

INSERT INTO ARTICULO VALUES ('BOCADILLOS')
INSERT INTO ARTICULO VALUES ('COMPLEMENTO')
INSERT INTO ARTICULO VALUES ('PIZZA')
INSERT INTO ARTICULO VALUES ('BOCADILLOS')
INSERT INTO ARTICULO VALUES ('COMPLEMENTO')
INSERT INTO ARTICULO VALUES ('PIZZA')
INSERT INTO ARTICULO VALUES ('COMPLEMENTO')
INSERT INTO ARTICULO VALUES ('PIZZA')
INSERT INTO ARTICULO VALUES ('BOCADILLOS')
INSERT INTO ARTICULO VALUES ('COMPLEMENTO')
INSERT INTO ARTICULO VALUES ('PIZZA')
INSERT INTO ARTICULO VALUES ('BOCADILLOS')
INSERT INTO ARTICULO VALUES ('COMPLEMENTO')
INSERT INTO ARTICULO VALUES ('COMPLEMENTO')
INSERT INTO ARTICULO VALUES ('COMPLEMENTO')

```

```

-----
CREATE TABLE INGREDIENTE_ARTICULO (
    NOMBRE_INGREDIENTE VARCHAR(20),
    ID_ARTICULO INT,

    CONSTRAINT PK_IXV
    PRIMARY KEY (NOMBRE_INGREDIENTE, ID_ARTICULO),
    CONSTRAINT FK_IVXV
    FOREIGN KEY (NOMBRE_INGREDIENTE)
    REFERENCES INGREDIENTE (NOMBRE_INGREDIENTE),
    CONSTRAINT FK_IVXI
    FOREIGN KEY (ID_ARTICULO)
    REFERENCES ARTICULO (ID_ARTICULO)
)

```

```

INSERT INTO INGREDIENTE_ARTICULO VALUES ('ANCHOAS', 1)
INSERT INTO INGREDIENTE_ARTICULO VALUES ('PIÑA', 1)
INSERT INTO INGREDIENTE_ARTICULO VALUES ('TOCINO', 2)
INSERT INTO INGREDIENTE_ARTICULO VALUES ('HONGOS', 2)
INSERT INTO INGREDIENTE_ARTICULO VALUES ('SOYA', 3)
INSERT INTO INGREDIENTE_ARTICULO VALUES ('PIMIENTO', 2)

```

```

-----
CREATE TABLE ARTICULO_VENTA (
    ID_ARTICULO INT,
    ID_VENTA INT,
    CANTIDAD INT,
    CONSTRAINT PK_AV
    PRIMARY KEY (ID_ARTICULO, ID_VENTA),
    CONSTRAINT PK_AVXA
    FOREIGN KEY (ID_ARTICULO)
    REFERENCES ARTICULO (ID_ARTICULO),
    CONSTRAINT PK_AVXV
    FOREIGN KEY (ID_VENTA)
    REFERENCES VENTA (ID_VENTA)
)

```

```

INSERT INTO ARTICULO_VENTA VALUES (1, 1, 2)
INSERT INTO ARTICULO_VENTA VALUES (3, 1, 1)
INSERT INTO ARTICULO_VENTA VALUES (27, 1, 1)
INSERT INTO ARTICULO_VENTA VALUES (2, 2, 3)
INSERT INTO ARTICULO_VENTA VALUES (26, 2, 2)

```

```
-----  
-----  
  
CREATE TABLE PIZZA(  
  
    ID_ARTICULO INT NOT NULL,  
    NOMBRE VARCHAR(30),  
    TAMAÑO VARCHAR(10),  
    PRECIO FLOAT,  
  
    CONSTRAINT PK_PIZZA  
    PRIMARY KEY(ID_ARTICULO),  
    CONSTRAINT FK_AP  
    FOREIGN KEY(ID_ARTICULO)  
    REFERENCES ARTICULO(ID_ARTICULO) ON DELETE CASCADE,  
    CONSTRAINT CK_TAMAÑO  
    CHECK (TAMAÑO IN('GRANDE', 'MEDIANO', 'PEQUEÑO'))  
)  
  
INSERT INTO PIZZA VALUES(1, 'PIZZA MARGARITA', 'MEDIANO', 225.00)  
INSERT INTO PIZZA VALUES(9, 'PIZZA CUATRO ESTACIONES', 'GRANDE', 400.00)  
INSERT INTO PIZZA VALUES(2, 'CAPRICHOSA', 'MEDIANO', 225.00)  
INSERT INTO PIZZA VALUES(15, 'CUATRO QUESOS', 'MEDIANO', 225.00)  
INSERT INTO PIZZA VALUES(11, 'NAPOLITANA', 'PEQUEÑO', 150.00)  
INSERT INTO PIZZA VALUES(18, 'HAWAIANA', 'GRANDE', 400)  
INSERT INTO PIZZA VALUES(19, 'DE PEPPERONI', 'GRANDE', 400)  
INSERT INTO PIZZA VALUES(23, 'VEGETARIANA', 'MEDIANO', 225)
```

```
-----  
-----  
  
CREATE TABLE BOCADILLO(  
  
    ID_ARTICULO INT NOT NULL,  
    NOMBRE VARCHAR(25) NOT NULL,  
    TAMAÑO VARCHAR(10) NOT NULL,  
    PRECIO FLOAT,  
  
    CONSTRAINT PK_BOCADILLO PRIMARY KEY(ID_ARTICULO),  
    CONSTRAINT FK_AB FOREIGN KEY(ID_ARTICULO) REFERENCES  
ARTICULO(ID_ARTICULO) ON DELETE CASCADE,  
    CONSTRAINT CK_TAMAÑOB CHECK (TAMAÑO IN('GRANDE', 'MEDIANO',  
'PEQUEÑO'))  
)  
  
INSERT INTO BOCADILLO VALUES(3, 'PANINI POLLO', 'MEDIANO', 65.00)  
INSERT INTO BOCADILLO VALUES(6, 'ENSALADA CESAR', 'PEQUEÑO', 40.00)  
INSERT INTO BOCADILLO VALUES(7, 'LASAGNNA', 'MEDIANO', 66.00)  
INSERT INTO BOCADILLO VALUES(20, 'CHILENA', 'GRANDE', 75.00)  
INSERT INTO BOCADILLO VALUES(12, 'SPAGETTI', 'PEQUEÑO', 39.00)  
INSERT INTO BOCADILLO VALUES(13, 'PANINI SUPREMO', 'GRANDE', 79.00)  
INSERT INTO BOCADILLO VALUES(16, 'CALZONES', 'MEDIANO', 55.50)  
INSERT INTO BOCADILLO VALUES(24, 'SOPA DE FRIJOLES', 'MEDIANO', 35.00)
```

```
-----  
-----  
  
CREATE TABLE COMPLEMENTO(  
  
    ID_ARTICULO INT NOT NULL,
```

```

    NOMBRE VARCHAR(20) NOT NULL,
    PRECIO FLOAT NOT NULL,

    CONSTRAINT PK_COMPLEMENTO
    PRIMARY KEY(ID_ARTICULO),
    CONSTRAINT FK_AC
    FOREIGN KEY(ID_ARTICULO)
    REFERENCES ARTICULO(ID_ARTICULO) ON DELETE CASCADE,
)

INSERT INTO COMPLEMENTO VALUES(4, 'ALITAS', 52.00 )
INSERT INTO COMPLEMENTO VALUES(5, 'PAN DE AJO', 51.00)
INSERT INTO COMPLEMENTO VALUES(8, 'REFRESCO', 31.00)
INSERT INTO COMPLEMENTO VALUES(10, 'SUNDAY', 59.00)
INSERT INTO COMPLEMENTO VALUES(14, 'PALITROQUES', 38.00)
INSERT INTO COMPLEMENTO VALUES(19, 'PECHURRICAS', 33.00)
INSERT INTO COMPLEMENTO VALUES(22, 'ENCANELADOS', 35.00)
INSERT INTO COMPLEMENTO VALUES(25, 'PAPAS FRITAS', 25.00)
INSERT INTO COMPLEMENTO VALUES(26, 'PIE DE MANZANA', 50.00)
INSERT INTO COMPLEMENTO VALUES(27, 'PIE DE LIMON', 40.00)

-----

CREATE TABLE PRODUCTO_ESTRELLA(
    ID_ARTICULO INT NOT NULL,
    NOMBREPE VARCHAR(25) NOT NULL,

    CONSTRAINT PK_PE
    PRIMARY KEY(ID_ARTICULO),
    CONSTRAINT FK_PEARTICULO
    FOREIGN KEY(ID_ARTICULO)
    REFERENCES ARTICULO(ID_ARTICULO)
)

INSERT INTO PRODUCTO_ESTRELLA VALUES(1, 'PE1')
INSERT INTO PRODUCTO_ESTRELLA VALUES(3, 'PE2')
INSERT INTO PRODUCTO_ESTRELLA VALUES(7, 'PE3')
INSERT INTO PRODUCTO_ESTRELLA VALUES(5, 'PE4')

-----

CREATE TABLE PROMOCION (
    ID_PROMOCION INT IDENTITY(1,1) NOT NULL,
    NOMBRE_PROMOCION VARCHAR(30),
    FECHA_INICIO DATE,
    FECHA_FIN DATE,
    CONSTRAINT PK_PROMOCION PRIMARY KEY(ID_PROMOCION),
)

INSERT INTO PROMOCION VALUES('COMBOMAX', '12-03-2013', '12-05-2013');
INSERT INTO PROMOCION VALUES(' FAMILCOMBO', '12-05-2013', '12-11-2013');
INSERT INTO PROMOCION VALUES('NAVIDEÑO', '11-11-2013', '12-12-2013');
INSERT INTO PROMOCION VALUES('COMELONES', '12-07-2013', '12-12-2013');

CREATE TABLE REGALO (

```



```
ID_REGALO INT IDENTITY(1,1) NOT NULL,
NOMBRE_REGALO VARCHAR(30) NOT NULL,
CONSTRAINT PK_REGALO PRIMARY KEY(ID_REGALO)
)

INSERT INTO REGALO VALUES ('PELUCHE');
INSERT INTO REGALO VALUES ('LLAVERO');
INSERT INTO REGALO VALUES ('GORRA');
INSERT INTO REGALO VALUES ('CAMISAS');
INSERT INTO REGALO VALUES ('POSTRES');

CREATE TABLE PROMO_ARTICULO(
ID_ARTICULO INT NOT NULL,
ID_PROMOCION INT NOT NULL,
CONSTRAINT PK_PROMO_ART PRIMARY KEY(ID_ARTICULO, ID_PROMOCION),
CONSTRAINT FK_PROMO_ART_PROMO FOREIGN KEY(ID_PROMOCION) REFERENCES
PROMOCION(ID_PROMOCION),
CONSTRAINT FK_PROMO_ART_ART FOREIGN KEY(ID_ARTICULO) REFERENCES
ARTICULO(ID_ARTICULO)
)

INSERT INTO PROMO_ARTICULO VALUES (1,2);
INSERT INTO PROMO_ARTICULO VALUES (2,3);
INSERT INTO PROMO_ARTICULO VALUES (3,1);

CREATE TABLE PROMO_REGALO(
ID_PROMOCION INT NOT NULL,
ID_REGALO INT NOT NULL,
CONSTRAINT PK_PROMO_REGA PRIMARY KEY(ID_PROMOCION, ID_REGALO),
CONSTRAINT FK_ID_PROM FOREIGN KEY(ID_PROMOCION) REFERENCES
PROMOCION(ID_PROMOCION),
CONSTRAINT FK_REGALO FOREIGN KEY(ID_REGALO) REFERENCES REGALO(ID_REGALO)
);

INSERT INTO PROMO_REGALO VALUES (1,2);
INSERT INTO PROMO_REGALO VALUES (2,3);
INSERT INTO PROMO_REGALO VALUES (3,4);
INSERT INTO PROMO_REGALO VALUES (4,1);

CREATE TABLE CLIENTE_REGALO(
ID_REGALO INT NOT NULL,
ID_CLIENTE VARCHAR(20) NOT NULL,
CONSTRAINT PK_CLI_REG PRIMARY KEY(ID_CLIENTE, ID_REGALO),
CONSTRAINT FK_CLI_REG_CLI FOREIGN KEY(ID_CLIENTE) REFERENCES
CLIENTE(ID_CLIENTE),
CONSTRAINT FK_CLI_REG_REG FOREIGN KEY(ID_REGALO) REFERENCES
REGALO(ID_REGALO)
)

INSERT INTO CLIENTE_REGALO VALUES (1, '0401-2000-56584');
INSERT INTO CLIENTE_REGALO VALUES (1, '0109-1987-25467');
INSERT INTO CLIENTE_REGALO VALUES (2, '0401-2000-56584');
INSERT INTO CLIENTE_REGALO VALUES (2, '0109-1987-25467');
INSERT INTO CLIENTE_REGALO VALUES (3, '0109-1987-25467');

CREATE TABLE ARTICULO_PEDIDO(
ID_ARTICULO INT NOT NULL,
```

```

ID_PEDIDO INT NOT NULL,
CANTIDAD INT,
CONSTRAINT PK_ART_PED PRIMARY KEY (ID_ARTICULO, ID_PEDIDO),
CONSTRAINT FK_ART_PED_ART FOREIGN KEY (ID_ARTICULO) REFERENCES
ARTICULO (ID_ARTICULO) ON DELETE CASCADE,
CONSTRAINT FK_ART_PED_PED FOREIGN KEY (ID_PEDIDO) REFERENCES
PEDIDO (ID_PEDIDO) ON DELETE CASCADE
)

INSERT INTO ARTICULO_PEDIDO VALUES (1,1,1)
INSERT INTO ARTICULO_PEDIDO VALUES (3,1,5)
INSERT INTO ARTICULO_PEDIDO VALUES (2,1,3)
INSERT INTO ARTICULO_PEDIDO VALUES (1,2,2)

```

CONSULTAS:

```

-----CONSULTAS-----
-----
USE PIZZERIA

/* CONSULTA 1*/
CREATE PROCEDURE CONSULTA1
AS
BEGIN
SELECT P.NOMBRE, COUNT(NOMBRE_INGREDIENTE) AS CANTIDAD_INGREDIENTES, P.PRECIO
FROM PIZZA P , INGREDIENTE_ARTICULO AI
WHERE P.ID_ARTICULO = AI.ID_ARTICULO
GROUP BY P.NOMBRE, P.PRECIO
HAVING COUNT(NOMBRE_INGREDIENTE) = 2
END

/*CONSULTA 2*/
CREATE PROCEDURE CONSULTA2
AS
BEGIN
SELECT SUM(V.MONTO) AS "TOTAL VENTAS (NO PIZZAS)" FROM VENTA
V, ARTICULO_VENTA AV
WHERE V.ID_VENTA = AV.ID_VENTA AND AV.ID_ARTICULO NOT IN (SELECT
ID_ARTICULO FROM PIZZA)
END
/*CONSULTA 3*/
CREATE PROCEDURE CONSULTA3
AS
BEGIN
SELECT C.PRIMER_NOMBRE, COUNT(P.ID_PEDIDO) AS "CANTIDAD DE PEDIDOS" FROM
VENTA_DOMICILIO VD, CLIENTE C, PEDIDO P
WHERE C.ID_CLIENTE = P.ID_CLIENTE AND P.ID_PEDIDO = VD.ID_PEDIDO AND
P.FECHA_PEDIDO >= '01/12/2013'
GROUP BY C.PRIMER_NOMBRE
ORDER BY "CANTIDAD DE PEDIDOS"
END
/*CONSULTA 4*/
CREATE PROCEDURE CONSULTA4
AS
BEGIN
SELECT R.NOMBRE, R.ID_SCOOTER FROM REPARTIDOR R
WHERE R.ID_REPARTIDOR IN

```

```
(SELECT ID_REPARTIDOR FROM PEDIDO WHERE ID_PEDIDO = 111 AND FECHA_PEDIDO =  
'08/06/1999')  
END  
/*CONSULTA 5*/  
CREATE PROCEDURE CONSULTA5  
AS  
BEGIN  
SELECT R.NOMBRE, COUNT(P.ID_PEDIDO) AS "CANTIDAD DE PEDIDOS" FROM REPARTIDOR  
R, PEDIDO P  
WHERE R.ID_REPARTIDOR = P.ID_PEDIDO AND R.ID_REPARTIDOR = 222 AND  
P.FECHA_PEDIDO >= '01/24/2013'  
GROUP BY R.NOMBRE  
ORDER BY "CANTIDAD DE PEDIDOS"  
END
```