Mini Proyecto

Base de Datos I

Docente: Maria Andrea RodriguezTastets

Ayudante: Matías Medina Silva

Alumno: Pablo Venegas Olate

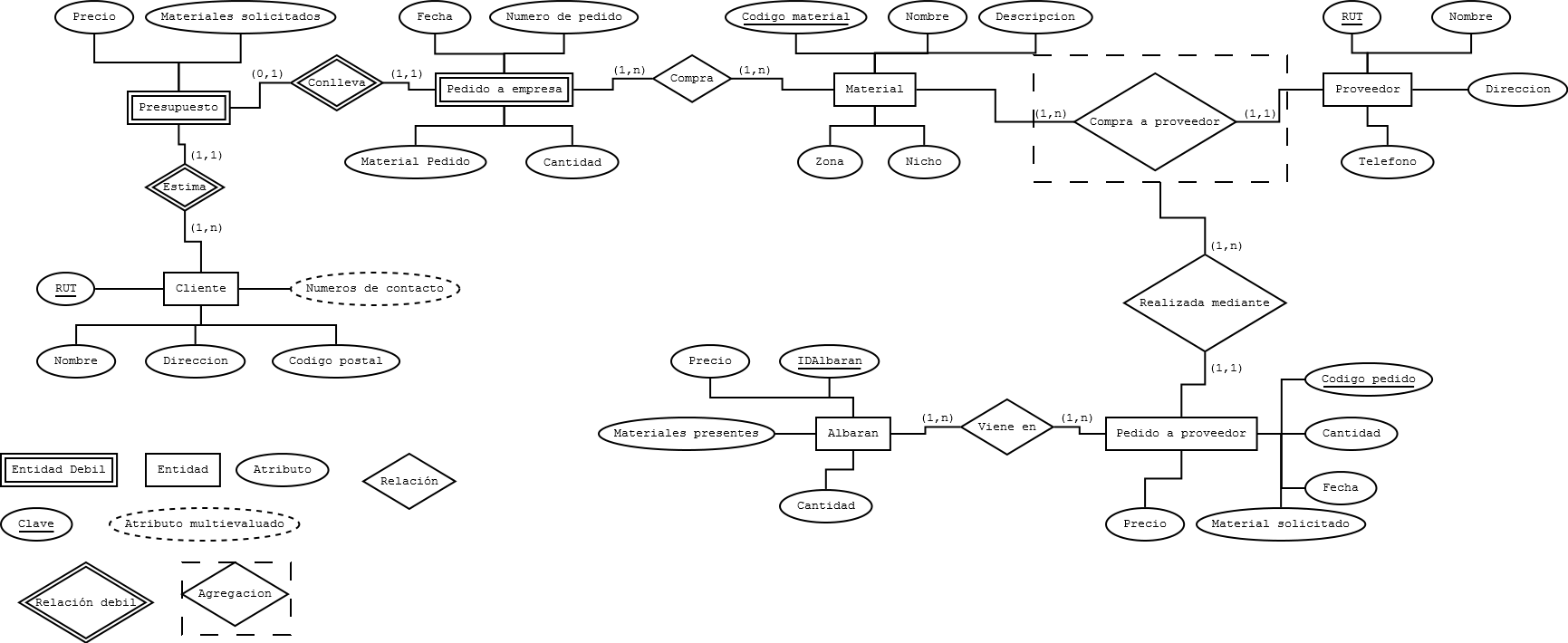
Dpto. de Ingeniería Civil Informática y Ciencias de la Computación

24 Octubre 2016

Introducción.

Mediante el presente “mini-proyecto” se busca reforzar el manejo de principios básicos de Base de Datos, tales como la construcción de un modelo entidad-relación a partir de un enunciado, la construcción de su modelo relacional, además de generar consultas del algebra relacional sobre este mismo modelo, por otro lado se busca crear la base de datos en SQL mediante el lenguaje postgresql y además hacer las mismas consultas del algebra relacional ahora para SQL, generando datos y llenando las tablas creadas.

Modelo Entidad-Relación.

En esta sección se entrega el MER, que muestra de manera grafica como estará relacionada nuestra base de datos.

\*Precio por cada material en Albarán debe ser igual al precio por cada material en Pedido a proveedor, de lo contrario el precio con el que se realiza el pedido no fue respetado por el proveedor.

Modelo Relacional.

En esta sección se detalla el modelo relacional de la base de datos representada en el MER anterior, especificando las claves de cada entidad y además dando a conocer las Foreign Keys respecticas de cada entidad.

Material (CodigoMaterial, Nombre, Descripcion, Zona, Nicho)

Proveedor (RUT, Nombre, Telefono, Direccion)

PedidoAProveedor (CodigoPedido, Cantidad, MaterialSolicitado, Fecha, Precio)

Cliente (RUT, Nombre, Direccion, CodigoPostal)

NumerosDeContacto (RUTCliente, NumeroDeContacto)

-FK: RUTCliente FROM Cliente (RUT)

Presupuesto (RUTCliente, MaterialesSolicitados, Precio,Cantidad)

-FK: RUTCliente FROM Cliente (RUT)

PedidoAEmpresa (RUTCliente, Fecha, NumeroPedido, MaterialPedido, Cantidad)

-FK: RUTCliente FROM Presupuesto (RUTCliente)

Albaran (IDAlbaran, Precio, Cantidad, MaterialesPresentes)

RealizadaMediante (RUTProveedor, CodigoMaterial, CodigoPedido)

-FK: RUTProveedor FROM Proveedor (RUT)

CodigoMaterial FROM Material (CodigoMaterial)

CodigoPedido FROM PedidoAProveedor (CodigoPedido)

Consultas en algebra relacional.

En esta sección se responden las consultas en algebra relacional a partir del modelo relacional antes planteado.

1. Nombre de los proveedores que han suministrado hormigón armado a la compañía desde el 1 de enero de 2000.
2. Total de unidades (metros cuadrados) de pavimento de gres vendidas en el último mes.
3. Nombre de los clientes que han solicitado presupuestos superiores a 60.000 $.

Código SQL.

En esta sección se da a conocer el código utilizado para crear las tablas de la base de datos e inmediatamente después insertar los datos que son necesarios para poder responder las consultas en SQL que se nos planteaban anteriormente, solo se insertaron los datos que se ocuparían en las tablas que se ven afectadas por las consultas, al menos en esta instancia eran un tanto irrelevante la generación de todos los datos.

DROP TABLE Material CASCADE;

DROP TABLE Proveedor CASCADE;

DROP TABLE PedidoAProveedor CASCADE;

DROP TABLE Cliente CASCADE;

DROP TABLE Telefonos CASCADE;

DROP TABLE Presupuesto CASCADE;

DROP TABLE PedidoAEmpresa CASCADE;

DROP TABLE Albaran CASCADE;

DROP TABLE RealizadaMediante CASCADE;

CREATE TABLE Material

(CodigoMaterial varchar(50),

Nombre varchar(50),

Descripcion varchar(50),

Zona varchar(50),

Nicho varchar(50),

primary key (CodigoMaterial));

CREATE TABLE Proveedor

(RUT varchar(50),

Nombre varchar(50),

Telefono varchar(50),

Direccion varchar(50),

primary key (RUT));

CREATE TABLE PedidoAProveedor

(CodigoPedido varchar(50),

Cantidad int,

MaterialSolicitado varchar(50),

Precio varchar(50),

Fecha date,

primary key (CodigoPedido));

CREATE TABLE Cliente

(RUT varchar(50),

Nombre varchar(50),

Direccion varchar(50),

CodigoPostal varchar(50),

primary key (RUT));

CREATE TABLE Telefonos

(RUTCliente varchar(50),

Telefono varchar(50),

foreign key (RUTCliente) references Cliente(RUT));

CREATE TABLE Presupuesto

(RUTCliente varchar(50),

MaterialSolicitado varchar(50),

Precio varchar(50),

Cantidad int,

foreign key (RUTCliente) references Cliente(RUT));

CREATE TABLE PedidoAEmpresa(RUTCliente varchar(50),

Fecha date,

NumeroPedido varchar(50),

Cantidad int,

MaterialPedido varchar(50),

foreign key (RUTCliente) references Cliente(RUT));

CREATE TABLE Albaran

(IDAlbaran varchar(50),

Precio varchar(50),

Cantidad int,

MaterialesPresentes varchar(50),

primary key (IDAlbaran));

CREATE TABLE RealizadaMediante

(RUTProveedor varchar(50),

CodigoMaterial varchar(50),

Codigo\_Pedido varchar(50),

foreign key (RUTProveedor) references Proveedor(RUT),

foreign key (CodigoMaterial) references Material(CodigoMaterial),

foreign key (Codigo\_Pedido) references PedidoAProveedor(CodigoPedido));

INSERT INTO "cliente" (RUT,Nombre,Direccion,CodigoPostal) VALUES

('11.111.111-1','Danny Rand','Hells Kitchen','1111'),

('22.222.222-2','Matt Murdock','Hells Kitchen','2222'),

('33.333.333-3','John Smith','Hells Kitchen','3333'),

('44.444.444-4','Jessica Jones','Hells Kitchen','4444'),

('55.555.555-5','Luke Cage','Hells Kitchen','5555');

INSERT INTO "presupuesto" (RUTCliente,materialsolicitado,precio,cantidad) VALUES

('11.111.111-1','acero','50000','10'),

('11.111.111-1','madera','80000','10'),

('11.111.111-1','pavimento','70000','10'),

('11.111.111-1','metal','40000','10'),

('33.333.333-3','madera','10000','1'),

('33.333.333-3','madera','50000','5'),

('33.333.333-3','madera','61000','6'),

('33.333.333-3','madera','70000','7'),

('44.444.444-4','acero','50000','1'),

('55.555.555-5','madera','50000','1');

INSERT INTO "pedidoaempresa" (RUTCliente,fecha,NumeroPedido,cantidad,MaterialPedido) VALUES

('11.111.111-1','2016/08/01','1','10','pavimento de gres'),

('11.111.111-1','2016/10/20','2','10','pavimento de gres'),

('44.444.444-4','2016/10/20','1','10','pavimento de gres'),

('44.444.444-4','2016/10/20','2','10','acero'),

('55.555.555-5','2016/08/02','1','10','pavimento de gres');

INSERT INTO "proveedor" (RUT,nombre,telefono,direccion) VALUES

('9.999.999-9','Tony Stark','111111','Winterfell'),

('8.888.888-8','Ned Stark','2222222','Winterfell'),

('7.777.777-7','Bran Stark','3333333','Winterfell'),

('6.666.666-6','Sansa Stark','4444444','Winterfell'),

('5.555.555-5','Arya Stark','5555555','Winterfell');

INSERT INTO "pedidoaproveedor" (CodigoPedido,Cantidad,MaterialSolicitado,precio,fecha) VALUES

('1','12','hormigon armado','500','2000/01/12'),

('2','10','hormigon armado','500','1999/12/25'),

('3','5','acero externo','500','1999/12/25'),

('4','20','acero externo','500','2000/01/12'),

('5','10','hormigon armado','500','1999/12/25'),

('6','10','hormigon armado','500','2000/01/12'),

('7','15','hormigon armado','500','1999/12/25');

INSERT INTO "material" (CodigoMaterial,Nombre,Descripcion,Zona,Nicho) VALUES

('001','hormigon armado','hormigon que viene armado de fabrica','A','1'),

('002','acero externo','acero que viene del exterior','B','2');

INSERT INTO "realizadamediante" (RUTProveedor,CodigoMaterial,Codigo\_Pedido) VALUES

('9.999.999-9','001','1'),

('9.999.999-9','001','2'),

('8.888.888-8','002','3'),

('8.888.888-8','002','4'),

('7.777.777-7','001','5'),

('6.666.666-6','001','6'),

('5.555.555-5','001','7');

Consultas en SQL.

En esta seccion se dan a conocer los códigos para las consultas que se realizaron en SQL a partir de las del algebra relacional y las imágenes mostrando que funcionan, además de presentar las tablas con los datos ingresados para cada caso.

1. Nombre de los proveedores que han suministrado hormigón armado a la compañía desde el 1 de enero de 2000.

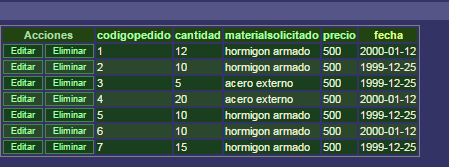
SELECT proveedor.nombre

from proveedor , realizadamediante, pedidoaproveedor

where pedidoaproveedor.materialsolicitado = 'hormigon armado'and

pedidoaproveedor.codigopedido = realizadamediante.codigo\_pedido and

realizadamediante.rutproveedor = proveedor.rut and pedidoaproveedor.fecha > '2000-01-01'



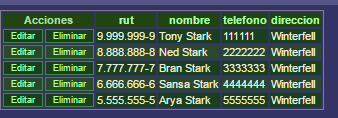
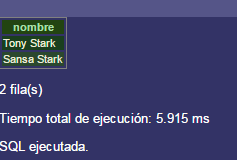


Tabla Proveedor.

Tabla PedidoAProveedor.



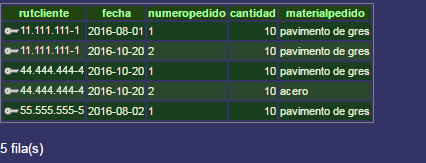
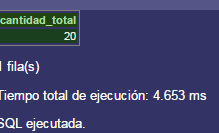
**Tabla RealizaMediante. Tabla Resultado.**

1. Total de unidades (metros cuadrados) de pavimento de gres vendidas en el último mes.

select SUM(cantidad) as cantidad\_total

from pedidoaempresa

where pedidoaempresa.materialpedido='pavimento de gres' and CURRENT\_DATE-fecha<30

****

**Tabla PedidoAEmpresa. Tabla Resultado.**

C:\Users\PLS\Google Drive\Universidad\Base de datos\MiniProyecto\Consulta2\Screenshot_56.png

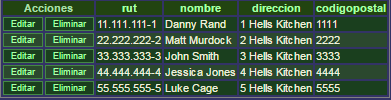
**Tabla Resultado (sin filtrar).**

1. Nombre de los clientes que han solicitado presupuestos superiores a $60.000.

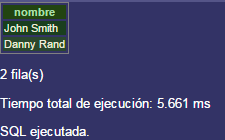
select distinct Cliente.Nombre

from Cliente, Presupuesto

where Presupuesto.Precio>'60000' and Presupuesto.rutcliente= cliente.rut



**Tabla Clientes.**



**Tabla Presupuesto.**

**Tabla Resultado.**

Conclusión.

A lo largo del presente proyecto se logra dar con una forma de repasar de manera práctica y rápida contenidos ya evaluados y además de ayudarnos a ejercitar con el manejo de la poderosa herramienta que es SQL, ya que con ella es posible llevar nuestra base de datos a una forma más “física”, de manera que podemos ver como se representan los datos en las “tablas”, familiarizarnos mas con la sintaxis y poder hacer las consultas sobre los datos, de manera fácil y rápida, gracias a lo poderosa que es esta herramienta.