**Bases de Datos**

**Profesora**: Silvia Salgado

**Email**: silvia.salgado@gmail.com

**Programa a utilizar**: Microsoft SQL Server 2012

CLASES 1 Y 2

El Microsoft SQL Server es un Sistema de Gestion de Base deDatos (SGBD o DBMS).

Modelo Relacional inventado por el Dr. Codd.

Algunos de los modelos de gestión relacionales son SQL Server, Oracle, DB2, Informix. Mientras que a MySQL todavía le falta, dado que es opensource.

El motor (engine) es quien permite realizar alta/bajas/modificación de las bases de datos.

Los nombres de las bases de datos en el modelo relacional pueden tener: Letras, Numeros y “\_”. NO SE PUEDE DEJAR ESPACIOS – NO SE PUEDE EMPEZAR CON NUMEROS – NO SE PUEDE UTILIZAR CARACTERES ESPECIALES.

**¿Qué es una tabla?** Es un conjunto de registros (filas o “tuplas” ) y Columnas ( Atributo o Campo).

Tipos de datos que podemos utilizar.

Numéricos: INT / BIGINT / SMALLITN

Caracteres: …CHAR

**DDL** (Data Definition Languaje) es una parte del lenguaje SQL y nos permite: CREATE (crear) / ALTER (modificar) / DROP (borrar).

**NOTA: Siempre para trabajar con SQL SERVER debo ir a TOOL / OPTIONS / DESIGNES y destildar “Prevent Savin”**

--------------------OOOOOO--------------------

**Crear una tabla en código SQL:**

Vamos a “New Query” y escribimos.

Nombre

Tipo

CREATE TABLE Vendedor(ID\_vend INT not null, Nombre char(30) not null, primary key (ID\_vend))

Clave Foránea ( Foreign Key) es la que se utiliza para relacionar una tabla con un registro de otra tabla.

--------------------OOOOOO--------------------

Un Ejemplo de tabla “Factura” en la que se aplica claves foráneas:

CREATE TABLE Factura (

ID\_fact int not null,

Fecha DATE not null,

Cliente int not null,

vendedor int not null,

PRIMARY KEY (ID\_fact),

FOREIGN KEY (Cliente) REFERENCES Clientes(ID\_clientes),

FOREIGN KEY (vendedor) REFERENCES Vendedor(ID\_vend)

)

--------------------OOOOOO--------------------

Para insertar en una taba:

INSERT INTO Vendedor VALUES

(2,'SARASA');

INSERT INTO Vendedor VALUES

(3,'ASÑLDKG');

Esto se repite por cada registro que se quiere agregar.

Entre cada inser se puede utilizar “;” o la palabra “GO”

--------------------OOOOOO--------------------

Para comentar una línea para que no se ejecute usamos “ - - “ (2guionesjuntos)

Si resaltamos líneas, cuando damos execute solo se corren dichas líneas.

--------------------OOOOOO--------------------

Para ejecutar consultas o imprimir en pantalla una tabla se utiliza SELECT.

SELECT \* FROM Clientes

SELECT \* FROM Factura

SELECT \* FROM vendedor

--------------------OOOOOO--------------------

Tabla de productos EJ:

CREATE TABLE Producto(

ID\_prod int not null,

Descripcion Char(20) not null,

Precio int not null,

PRIMARY KEY(ID\_prod),

)

INSERT INTO Producto VALUES( 1, 'coca', 10)

INSERT INTO Producto VALUES( 2, 'fanta', 45)

INSERT INTO Producto VALUES( 3, 'pecsi', 5556)

--------------------OOOOOO--------------------

Para modificar una tabla se usa ALTER y el comando que necesitamos, por ejemplo para agregar una columna a Producto sería:

ALTER TABLE Detalle ADD col\_nueva INT

--------------------OOOOOO--------------------

Tarea dia 16/8 … VETERINARIA

--------------------OOOOOO--------------------

**Clase 2 – 23/8/2013**

Vamos a crear una base de datos de una veterinaria. Las tablas que vamos a necesitar serán:

Cliente

Mascota

Veterinario

Remedios

Turno

NOTA: La notación de los conectores que se utiliza es distinta. Se utiliza una que se denomina “pata de gallo” que indica que puede haber muchos de “xxxxx”. Ej: En la tabla mascota, la flecha donde esta dueño, sería una pata de gallo.

|  |
| --- |
| Cliente |
| ID\_Cliente  Nombre  Domicilio |

|  |
| --- |
| Turno |
| Turno\_ID  Fecha  Hora  Mascota  Veterinario  Diagnostico  Comentario |

|  |
| --- |
| Mascota |
| Mascota\_ID  NombreMascota  Dueño |

|  |
| --- |
| Veterinario |
| Vet\_ID  NombreVet |

|  |
| --- |
| Diagonostico |
| ID\_Diag  Nombre |

|  |
| --- |
| RemTur |
| TurnoID  RemID |

|  |
| --- |
| Laboratorio |
| ID\_Lab  Nombre  Direccion  Tel |

|  |
| --- |
| Remedios |
| ID\_Rem  Nombre  Laboratorio |

**Pk**

CODIGO DE LA BASE DE DATOS:

CREATE TABLE Cliente (

Cli\_ID INT not null,

Nombre CHAR(20) not null,

Domicilio CHAR(30),

PRIMARY KEY(Cli\_ID)

);

GO

CREATE TABLE Veterinario (

Vet\_ID INT NOT NULL,

Nombre CHAR(20)NOT NULL,

PRIMARY KEY(Vet\_ID)

);

GO

CREATE TABLE Laboratorio(

Lab\_ID INT NOT NULL,

Nombre CHAR(20) NOT NULL,

Direccion CHAR(30),

Telefono CHAR(20),

PRIMARY KEY(Lab\_ID)

);

GO

CREATE TABLE Mascota(

Masc\_ID INT NOT NULL,

Nombre CHAR(30) NOT NULL,

Dueño INT NOT NULL,

PRIMARY KEY(Masc\_ID),

FOREIGN KEY (Dueño) REFERENCES Cliente(CLi\_ID)

);

GO

CREATE TABLE Remedio(

Rem\_ID INT NOT NULL,

Nombre CHAR(30) NOT NULL,

Lab INT NOT NULL,

PRIMARY KEY(Rem\_ID),

FOREIGN KEY (Lab) REFERENCES Laboratorio(Lab\_ID)

);

GO

CREATE TABLE Diagnostico(

Diag\_ID INT NOT NULL,

Nombre CHAR(30),

PRIMARY KEY(Diag\_ID)

);

GO

CREATE TABLE Turno(

Turno\_ID INT NOT NULL,

Fecha DATE NOT NULL,

Hora CHAR(10) NOT NULL,

Mascota INT NOT NULL,

Veterinario INT NOT NULL,

Diagnostico INT NOT NULL,

Coment CHAR(300),

PRIMARY KEY (Turno\_ID),

FOREIGN KEY(Mascota) REFERENCES Mascota(Masc\_ID),

FOREIGN KEY (Veterinario)REFERENCES Veterinario(Vet\_ID),

FOREIGN KEY (Diagnostico)REFERENCES Diagnostico(Diag\_ID)

);

GO

CREATE TABLE RemTur(

Turno\_ID INT NOT NULL,

Rem\_ID INT NOT NULL,

PRIMARY KEY(Turno\_ID, Rem\_ID),

FOREIGN KEY(Turno\_ID)REFERENCES Turno(Turno\_ID),

FOREIGN KEY(Rem\_ID) REFERENCES Remedio(Rem\_ID)

);

GO

--------------------OOOOOO--------------------

Ejercicios de la [guía de ejercicios](guia%20de%20ejercicios%20Nro%201%20DABD%202013%20Da%20Vinci.doc)

Parte a introducción:

2)

CREATE TABLE Alumno (

Alu\_ID INT NOT NULL,

Nombre CHAR(30)NOT NULL,

PRIMARY KEY (Alu\_ID)

);

GO

CREATE TABLE Curso(

Curso\_ID INT NOT NULL,

Aula CHAR(5),

PRIMARY KEY(Curso\_ID)

);

GO

CREATE TABLE AluCur(

Curso INT NOT NULL,

Alumno INT NOT NULL,

PRIMARY KEY(Curso, Alumno),

FOREIGN KEY (Curso) REFERENCES Curso(Curso\_ID),

FOREIGN KEY (Alumno) REFERENCES Alumno(Alu\_ID)

);

