

<b>EVALUACIÓN</b>	Parcial 1	<b>GRUPO</b>	N3F	<b>FECHA</b>	20/4/2023
<b>MATERIA</b>	Bases de Datos 2				
<b>CARRERA</b>	Analista en Tecnologías de Información / Analista Programador				
<b>CONDICIONES</b>	- Puntos: 25 - Duración 2 horas - Sin material				

Una empresa de ropa infantil necesita construir una base de datos para soportar su operación. Se detalla las entidades más importantes:

Sucursales(**idsucursal**, descripcion, domicilio, fecha\_inicio\_act)

*Contiene los datos de las distintas sucursales que posee la empresa, un IdSucursal como entero autonumérico, con salto cada 2 y comenzando desde el nro 1 (1, 3,5,7,..). y la Fecha de inicio de actividades donde no se almacena la hora. La misma debe estar entre el periodo 1/1/1980 y 31/12/2100).*

Empleados(**Legajo**, nombre, apellido, fecha\_ingreso, rol, legajo\_jefe, id\_sucursal\_asignada, salario)

*Información sensible de los empleados. El Legajo como un valor entero, la fecha de ingreso sin la hora, el rol que puede ser VENDEDOR, CADETE, CAJERO, ADMINISTRATIVO. El salario no puede ser un valor menor o igual a 0.*

Clientes(**idCliente**, nombre, apellido, rut, telefono, categoria)

Los clientes se identifican con el IdCliente que es entero y autonumérico. El rut no puede contener valores nulos y las categorias pueden ser "MINORISTA" , "MAYORISTA"

Productos(**idProd**, descripcion, stock\_inicial, stock\_actual, precio\_unitario)

El campo IdProd es un valor alfanumérico de longitud 10. El stock\_inicial , si no se informa, es 13. A su vez, no puede ser negativo ni mayor a 157.

Facturas(**idsucursal**, **numTrx**, fecha, idCliente, idVendedor)

En cada sucursal se pueden realizar facturas con numeración independiente. Se graba la fecha y hora de la transacción.

Facturas\_items(**idSucursal**, **numTrx**, **nroLinea**, idProd, cantidad, precio\_unitario, total)

Las líneas de la factura se almacenan en ésta tabla. La columna "cantidad" no puede superar el valor 9999. El total es la cantidad x precio\_unitario

Se pide que escriba el script/instrucciones SQL necesarias para:

#	Enunciado	Puntaje Max
1	Creación de todas las tablas, teniendo en cuenta las restricciones de integridad marcadas para cada una de ellas y cualquier otra restricción de negocio indicada	10
2	Creación de los índices que correspondan en cada tabla	3
3	En la tabla Clientes se debe agregar la columna "Rating" de tipo entero y que solo pueda tener valores entre 1 y 10. Adicionalmente, agregar una columna "Referente" de tipo Varchar(10). No podrá haber dos referentes iguales.	4
4	Crear la instrucción para que en la tabla FACTURA_ITEMS se controle que el total informado no supere al precio_unitario x Cantidad	3
5	Se desea mantener la cantidad de horas trabajadas semanalmente por cada empleado (puede variar en diferentes semanas), para reflejar esto se agregaría un campo llamado HorasTrabajadas a la tabla Empleados. Esta solución está Normalizada? Justifique su respuesta. En caso de no estar de acuerdo plantee una solución alternativa que esté normalizada	5

## SOLUCION PROPUESTA

```
CREATE DATABASE PARCIAL1
```

```
USE PARCIAL1
```

```
CREATE TABLE SUCURSALES(  
idsucursal int identity(1,2) primary key,  
descripcion varchar(50),  
domicilio varchar(100),  
fecha_inicio_act date CHECK(fecha_inicio_act BETWEEN '1-1-1980' AND '31/12/2100')  
)
```

```
CREATE TABLE EMPLEADOS(  
legajo int primary key,  
nombre varchar(50),  
apellido varchar(50),  
fecha_ingreso date,  
rol varchar(50) CHECK (rol IN ('VENDEDOR', 'CADETE', 'CAJERO', 'ADMINISTRATIVO')),  
legajo_jefe int ,  
id_sucursal_asignada int,  
salario money CHECK (salario > 0),  
CONSTRAINT FK_JEFE FOREIGN KEY(legajo_jefe) REFERENCES EMPLEADOS(legajo),  
CONSTRAINT FK_SUC FOREIGN KEY(id_sucursal_asignada) REFERENCES SUCURSALES(idsucursal)  
)
```

```
CREATE TABLE CLIENTES(  
idcliente int identity primary key,  
nombre varchar(50),  
apellido varchar(50),  
rut varchar(15) not null,  
categoria varchar(15) CHECK (categoria IN ('MINORISTA', 'MAYORISTA'))  
)
```

```
CREATE TABLE PRODUCTOS(  
idprod varchar(10) primary key,  
descripcion varchar(50),  
stock_inicial int default 13,  
stock_actual int,  
precio_unitario money,  
constraint CK_STK CHECK(stock_inicial between 1 AND 157)  
)
```

```
CREATE TABLE FACTURAS(  
idsucursal int ,  
numtrx int,  
fecha datetime,  
idCliente int REFERENCES CLIENTES(idcliente),  
idVendedor int REFERENCES EMPLEADOS(legajo),  
CONSTRAINT PK_FACTURAS PRIMARY KEY(idsucursal,numtrx)  
)
```

```
CREATE TABLE FACTURAS_ITEMS(  
idsucursal int ,  
numtrx int ,  
nrolinea int,  
idProducto varchar(10) REFERENCES PRODUCTOS(idProd),  
cantidad int CHECK (cantidad <= 9999),
```

```
precio_unitario money,  
total money,  
CONSTRAINT PK_FAC_ITEMS PRIMARY KEY(idsucursal,numtrx,nrolinea),  
CONSTRAINT FK_PK FOREIGN KEY (idsucursal,numtrx) REFERENCES  
FACTURAS(idsucursal,numtrx)  
)
```

## --EJERCICIO 2

```
CREATE INDEX IDX_1 ON EMPLEADOS(id_sucursal_asignada);  
CREATE INDEX IDX_2 ON FACTURAS(idCliente);  
CREATE INDEX IDX_3 ON FACTURAS(idVendedor);  
CREATE INDEX IDX_4 ON FACTURAS_ITEMS(idproducto);
```

## -- EJERCICIO 3

```
ALTER TABLE CLIENTES  
ADD rating int CHECK(rating BETWEEN 1 and 10)  
  
ALTER TABLE CLIENTES  
ADD referente varchar(10) UNIQUE
```

## --EJERCICIO 4

```
ALTER TABLE FACTURAS_ITEMS  
ADD CONSTRAINT CK_TOTAL CHECK(TOTAL = PRECIO_UNITARIO * cantidad)
```

## --EJERCICIO 5

```
Empleados(Legajo, nombre, apellido, fecha_ingreso, rol, legajo_jefe,  
id_sucursal_asignada, salario, HorasTrabajadas)
```

-- NO ESTA EN 3FN. SE DEBERÍA NORMALIZAR

```
EMPLEADOS(Legajo, nombre, apellido, fecha_ingreso, rol, legajo_jefe,  
id_sucursal_asignada, salario)  
HORAS_TRABAJADAS(legajo_empleado, fecha, cantHorasTrabajadas)  
-----
```