# TrabajoPráctico Nro2

1. Crear tablas de Nivel 0.
2. Crear tablas de Nivel 1.
3. Definir para todos los casos restricciones por Unique, Not NULL, Check, claves foráneas y claves primarias.

**Tablas de nivel 0**

CREATETABLEPAISES

(

PAISINT,

descripcionVARCHAR(100)NOTNULL,

CONSTRAINTPK\_PAISPRIMARYKEY (PAIS),

CONSTRAINTUNIQUE\_DESCUNIQUE (descripcion)

)

CREATETABLEPRODUCTOS

(

PRODUCTOINT,

descripcionVARCHAR(100)NOTNULL,

precioFLOATNOTNULL,

CONSTRAINTPK\_PRODUCTOPRIMARYKEY (PRODUCTO)

)

**Tablas de nivel 1**

CREATE TABLE STOCKS

(

FECHADATETIME,

PRODUCTO INT,

cantidadINTNOTNULL,

CONSTRAINTPK\_FECHA\_PRODUCTOPRIMARYKEY (FECHA,PRODUCTO),

CONSTRAINTFK\_PRODUCTOFOREIGNKEY (PRODUCTO)REFERENCESPRODUCTOS(PRODUCTO)

)

CREATE TABLE PROVINCIAS

(

PROVINCIA INT,

descripcion VARCHAR(100)NOTNULL,

PAIS INT,

CONSTRAINT PK\_PROVINCIA PRIMARYKEY (PROVINCIA),

CONSTRAINT FK\_PAIS FOREIGNKEY (PAIS)REFERENCES PAISES(PAIS)

)

CREATE TABLE LOCALIDADES

(

LOCALIDAD INT,

descripcion VARCHAR(100)NOTNULL,

PROVINCIA INT,

CONSTRAINT PK\_LOCALIDAD PRIMARYKEY (LOCALIDAD),

CONSTRAINT FK\_PROVINCIA\_LOCALIDADES FOREIGNKEY (PROVINCIA) REFERENCES PROVINCIAS(PROVINCIA)

)

CREATETABLE PROVEEDORES

(

PROVEEDORINT,

descripcionVARCHAR(100)NOTNULL,

cuitVARCHAR(45)NOTNULL,

telefonoVARCHAR(45),

LOCALIDADINT,

direccionVARCHAR(100)NOTNULL,

fechaDATETIMENOTNULL,

CONSTRAINTPK\_PROVEEDORPRIMARYKEY (PROVEEDOR),

CONSTRAINTUNIQUE\_cuitUNIQUE (cuit),

CONSTRAINTFK\_LOCALIDAD\_PROVEEDORESFOREIGNKEY (LOCALIDAD)REFERENCESLOCALIDADES(LOCALIDAD)

)

CREATETABLECLIENTES

(

CLIENTE INT,

descripcion VARCHAR(100)NOTNULL,

telefono VARCHAR(45),

nro\_documento INT NOTNULL,

TIPO\_DOCUMENTO INT,

direccion VARCHAR(100)NOTNULL,

LOCALIDAD INT,

C\_FISCAL INT,

CONSTRAINT PK\_CLIENTE PRIMARYKEY (CLIENTE),

CONSTRAINT FK\_TIPO\_DOCUMENTO\_CLIENTES FOREIGNKEY (TIPO\_DOCUMENTO) REFERENCES TIPOS\_DOCUMENTOS(TIPO\_DOCUMENTO),

CONSTRAINT FK\_LOCALIDAD\_CLIENTES FOREIGNKEY (LOCALIDAD)REFERENCESLOCALIDADES(LOCALIDAD),

CONSTRAINTFK\_C\_FISCAL\_CLIENTESFOREIGNKEY (C\_FISCAL)REFERENCESCONDICION\_FISCAL(C\_FISCAL)

)

CREATETABLECOMPRAS\_CABECERA

(

fechaDATETIMENOTNULL,

PROVEEDORINT,

TIPOVARCHAR(3),

FACTURAINT,

CONSTRAINTPK\_TIPO\_FACTURA\_CCPRIMARYKEY (TIPO,FACTURA),

CONSTRAINTCK\_TIPO\_CCCHECK (TIPO='A'ORTIPO='B'ORTIPO='C'),

CONSTRAINTFK\_PROVEEDOR\_CCFOREIGNKEY (PROVEEDOR)REFERENCESPROVEEDORES(PROVEEDOR)

)

CREATETABLECOMPRAS\_DETALLE

(

PRODUCTOINT,

TIPOVARCHAR(3),

FACTURAINT,

cantidadINTNOTNULL,

precioFLOATNOTNULL,

CONSTRAINTPK\_PTF\_CDPRIMARYKEY (PRODUCTO,TIPO,FACTURA),

CONSTRAINTFK\_PRODUCTO\_CDFOREIGNKEY (PRODUCTO)REFERENCESPRODUCTOS(PRODUCTO),

CONSTRAINTFK\_TIPO\_FACTURA\_CDFOREIGNKEY (TIPO,FACTURA)REFERENCESCOMPRAS\_CABECERA(TIPO,FACTURA),

)

CREATETABLE VENTAS\_CABECERA

(

fechaDATETIMENOTNULL,

CLIENTEINT,

TIPOVARCHAR(3),

FACTURAINT,

CONSTRAINTPK\_TIPO\_FACTURA\_VCPRIMARYKEY (TIPO,FACTURA),

CONSTRAINTFK\_CLIENTE\_VCFOREIGNKEY (CLIENTE)REFERENCESCLIENTES(CLIENTE),

CONSTRAINTCK\_TIPO\_VCCHECK (TIPO='A'ORTIPO='B'ORTIPO='C')

)

CREATETABLE VENTAS\_DETALLE

(

PRODUCTOINT,

TIPOVARCHAR(3),

FACTURAINT,

cantidadINTNOTNULL,

precioFLOATNOTNULL,

CONSTRAINTPK\_PRODUCTO\_TIPO\_FACTURA\_VDPRIMARYKEY (PRODUCTO,TIPO,FACTURA),

CONSTRAINTFK\_PRODUCTO\_VDFOREIGNKEY (PRODUCTO)REFERENCESPRODUCTOS(PRODUCTO),

CONSTRAINTFK\_TIPO\_FACTURA\_VDFOREIGNKEY (TIPO,FACTURA)REFERENCESVENTAS\_CABECERA(TIPO,FACTURA)

)

1. Definir **Trigger**. Estudiar y documentar las tablas que podrían utilizar este objeto, detallando la acción a seguir de acuerdo al evento.

Los triggers o disparadores son objetos de la base de datos que ejecutan **acciones** cuando se producen ciertos **eventos como**inserciones, modificaciones, borrados, creación de tablas, etc.)

**Trigger** para la tabla **VENTAS\_DETALLE**, para la acción INSERT, cuando se inserta un registro en VENTAS\_DETALLE debe hacer un UPDATE en la tabla STOCK, modificando la cantidad del PRODUCTO insertado en VENTAS\_DETALLE. (Cantidad insertada en VENTAS\_DETALLE debe ser en STOCKS: Cantidad actual – cantidad insertada en VENTAS\_DETALLE).

**Trigger** para la tabla **COMPRAS\_DETALLE**, para la acción INSERT, cuando se inserta un registro debe hacer un UPDATE en la tabla STOCK, modificando la cantidad del PRODUCTO insertado en COMPRAS\_DETALLE. (Cantidad insertada en COMPRAS\_DETALLE debe ser en STOCKS: Cantidad actual + cantidad insertada en COMPRAS\_DETALLE).