

# Modelos y bases de datos

Google Data Center





## Tener en cuenta...

- Laboratorio 5  
**Viernes 2 de Mayo**
- Autoestudio 6  
**Jueves 8 de Mayo – 10:00 am**
- Laboratorio 6  
**Viernes 9 de Mayo**
- Proyecto: Diseño físico  
**Sábado 10 de Mayo**

# Proyecto

## AGENDA

S03. FORMULACIÓN DEL PROYECTO. (SA 8 FEB)

S05. DISEÑO CONCEPTUAL. GENERAL. (SA 22 FEB)

S07. DISEÑO CONCEPTUAL. EXTENDIDO. (SA 08 MAR)

S09. DISEÑO LÓGICO. ESTRUCTURA + DECLARATIVAS. (SA 22 MAR)

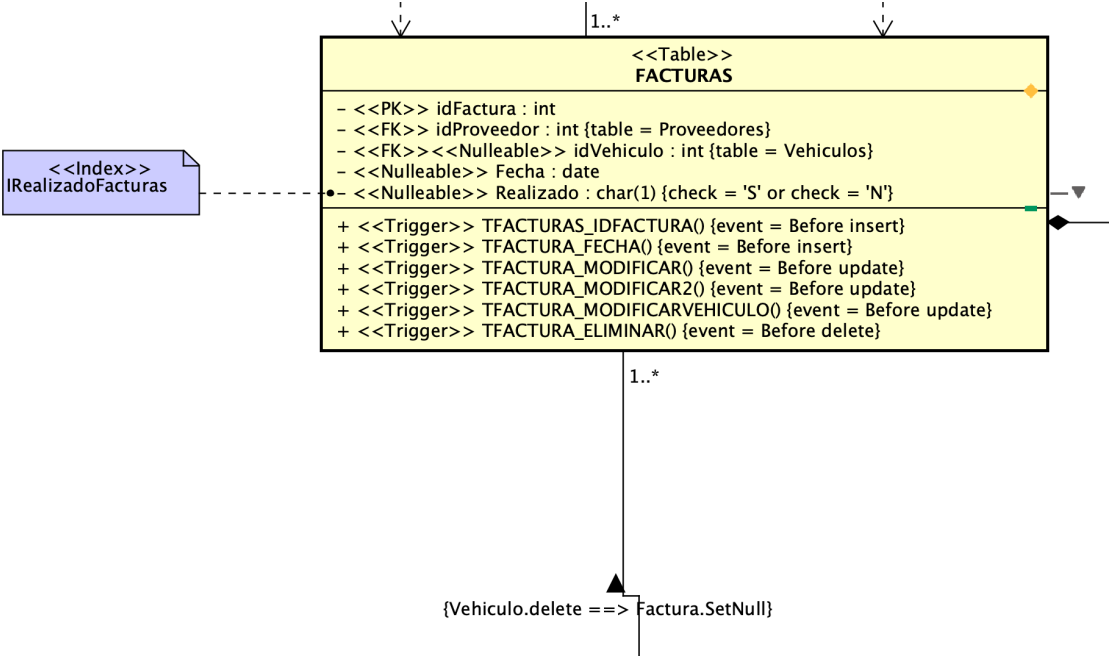
S11 . DISEÑO LÓGICO. ESTRUCTURA + PROCEDIMENTALES. (SA 05 ABR -> SA 12 ABR)

S15. DISEÑO FÍSICO. DATOS + COMPONENTES (SA 10 MAY)

S16. ENTREGA FINAL. PRIMER CICLO (SA 17 MAY).

S18. ENTREGA FINAL. SEGUNDO CICLO. (LU 19 MAY)

# Diagrama: Físico - Datos





# Diagrama: Físico - Componentes

PKG_PROVEEDORES
+ adicionarPersona(idProveedor : int, cedula : varchar, nombre : varchar) : void + adicionarEmpresa(idProveedor : int, nit : char, razonSocial : varchar, personaDeContacto : varchar, cedula : varchar) : void + adicionarProveedor(idProveedor : int, direccion : varchar, ciudad : varchar, departamento : varchar, direccionElectronica : varchar, activo : char) : void + adicionarTelefono(idTelefono : int, idProveedor : int, telefono : int) : void + adicionarOfrece(idProveedor : int, idProducto : int, cantidad : int) : void + modificarEmpresa(idProveedor varchar : int, personaDeContacto : varchar, cedula : varchar) : void + modificarProveedor(idProveedor varchar : int, direccion : varchar, ciudad : varchar, departamento : varchar, activo : char) : void + eliminarTelefono(idTelefono : int, idProveedor : int, telefono : int) : void + eliminarOfrece(idProveedor : int, idProducto : int, cantidad : int) : void + consultarProveedor() : Cursor + consultarPersona() : Cursor + consultarEmpresa() : Cursor + consultarOfrece() : Cursor + consultarTelefono() : Cursor

PKG_PRODUCTOS
+ adicionarProducto(idProducto : int, tamaño : varchar, precioCompra : int, precioVenta : int, idClasificacion : int, disponible : char) : void + adicionarClasificacion(idClasificacion : int, categoria : varchar, rangoPeso : varchar, origen : varchar) : void + modificarProducto(precioCompra : int, precioVenta : int, disponible : char) : void + consultarProducto() : Cursor + consultarClasificacion() : Cursor

# Diagrama: Físico - Seguridad

PKG_COORDINADORDECOMPRAS
+ adicionarFactura(idFactura : int, idProveedor : int, idVehiculo : int, fecha : date, realizado : char) : void + adicionarPedido(idPedido : int, idProducto : int, idFactura : int, fechaPedido : date, cantidad : int) : void + adicionarPersona(idProveedor : int, cedula : varchar, nombre : varchar) : void + adicionarEmpresa(idProveedor : int, nit : char, razonSocial : varchar, personaDeContacto : varchar, cedula : varchar) : void + adicionarProveedor(idProveedor : int, direccion : varchar, ciudad : varchar, departamento : varchar, direccionElectronica : varchar) : void + adicionarTelefono(idTelefono : int, idProveedor : int, telefono : int) : void + adicionarOfrece(idProveedor : int, idProducto : int, cantidad : int) : void + modificarFactura(idFactura : int, idVehiculo : int, fecha : date, realizado : char) : void + modificarEmpresa(idProveedor : int, personaDeContacto : varchar, cedula : varchar) : void + modificarProveedor(idProveedor : int, direccion : varchar, ciudad : varchar, departamento : varchar, activo : char) : void + eliminarTelefono(idTelefono : int, idProveedor : int, telefono : int) : void + eliminarOfrece(idProveedor : int, idProducto : int, cantidad : int) : void + consultarVProveedores() : Cursor + consultarFacturas() : Cursor + consultarPedidos() : Cursor + consultarProveedor() : Cursor + consultarOfrece() : Cursor + consultarVehiculos() : Cursor

PKG_JEFEDEPERSONAL
+ adicionarVehiculo(idVehiculo : int, idConductor : int, tamaño : int, placa : varchar, detalle : XMLType) : void + adicionarConductor(idConductor : int, nombre : varchar, cedula : varchar) : void + modificarVehiculo(idVehiculo : int, idConductor : int) : void + eliminarVehiculo(idVehiculo : int, idConductor : int, tamaño : int, placa : varchar, detalle : XMLType) : void + eliminarConductor(idConductor : int, nombre : varchar, cedula : varchar) : void + consultarVehiculo() : Cursor + consultarPedido() : Cursor

# Tener en cuenta...

Indices y Vistas	<div>Indices</div> <div>Definición de indices</div> <div>Vistas</div> <div>Definición de vistas</div> <div>XIndicesVistas</div> <div>Eliminación de vistas e indices</div>	<div>IndicesVistasOK</div> <div>Consultas que requieren los indices y las vistas definidos</div>
Componentes	<div>CRUDE</div> <div>Especificación de los paquetes de componentes</div> <div>CRUDI</div> <div>Implementación de los paquetes</div> <div>XCRUD</div> <div>Eliminación de los paquetes</div>	<div>CRUDOK</div> <div>Ingreso de datos correctos usando los procedimientos de los paquetes</div> <div>CRUDNoOK</div> <div>Intento de ingreso de datos erroneos protegidos por los procedimientos</div>

# Tener en cuenta...

Seguridad	ActoresE	SeguridadOK
	Especificación de los paquetes de actores	
	ActoresI	
	Implementación de dichos paquetes	
	Seguridad	
	Definición de roles y permisos	
	XSeguridad	
	Eliminación de paquetes de actores, roles y permisos.	



# Tener en cuenta...

Seguridad	ActoresE	SeguridadOK
	Especificación de los paquetes de actores	
	ActoresI	
	Implementación de dichos paquetes	
	Seguridad	
	Definición de roles y permisos	
	XSeguridad	
	Eliminación de paquetes de actores, roles y permisos.	

# CRUDE

```
--PAQUETE PROVEEDORES:

CREATE OR REPLACE PACKAGE PKG_PROVEEDORES
IS
PROCEDURE adicionarPersona(idProveedor VARCHAR, cedula VARCHAR, nombre VARCHAR);
PROCEDURE adicionarEmpresa(idProveedor VARCHAR, nit CHAR, razonSocial VARCHAR, personaDeContacto VARCHAR, cedula NUMBER);
PROCEDURE adicionarProveedor(idProveedor VARCHAR, direccion VARCHAR, ciudad VARCHAR,
    departamento VARCHAR, direccionElectronica VARCHAR, activo CHAR );
PROCEDURE adicionarTelefono(idTelefono NUMBER, idProveedor NUMBER, telefono NUMBER);
PROCEDURE adicionarOfrece(idProveedor VARCHAR, idProducto VARCHAR, cantidad INT);
PROCEDURE modificarEmpresa(XidProveedor VARCHAR, XpersonaDeContacto VARCHAR, Xcedula VARCHAR);
PROCEDURE modificarProveedorDireccion(XidProveedor VARCHAR, Xdireccion VARCHAR, Xciudad VARCHAR, Xdepartamento VARCHAR);
PROCEDURE modificarProveedorActivo(XidProveedor VARCHAR, Xactivo CHAR );
PROCEDURE eliminarTelefono(XidTelefono NUMBER);
PROCEDURE eliminarOfrece(XidProveedor VARCHAR, XidProducto VARCHAR);
PROCEDURE consultarProveedor(C_PROVEEDOR OUT SYS_REFCURSOR);
PROCEDURE consultarPersona(C_PERSONA OUT SYS_REFCURSOR);
PROCEDURE consultarEmpresa(C_EMPRESA OUT SYS_REFCURSOR);
PROCEDURE consultarOfrece(C_OFRECE OUT SYS_REFCURSOR);
PROCEDURE consultarTelefono(C_TELEFONO OUT SYS_REFCURSOR);
END;
/
```

# CRUDI

```
--(PACKAGE BODY) PROVEEDORES:

CREATE OR REPLACE PACKAGE BODY PKG_PROVEEDORES
IS
    PROCEDURE adicionarPersona(idProveedor VARCHAR, cedula VARCHAR, nombre VARCHAR)IS
    BEGIN
    INSERT INTO Personas VALUES (idProveedor, cedula, nombre);
    COMMIT;
    EXCEPTION
    WHEN OTHERS THEN
    ROLLBACK;
    Raise_application_error(-20050,SQLERRM);
    END adicionarPersona;

    PROCEDURE adicionarEmpresa(idProveedor VARCHAR, nit CHAR, razonSocial VARCHAR, personaDeContacto VARCHAR, cedula NUMBER)IS
    BEGIN
    INSERT INTO Empresas VALUES (idProveedor , nit , razonSocial, personaDeContacto , cedula );
    COMMIT;
    EXCEPTION
    WHEN OTHERS THEN
    ROLLBACK;
    Raise_application_error(-20050,SQLERRM);
    END adicionarEmpresa;
```

# ACTORESE

```
---PAQUETE GERENTE:  
CREATE OR REPLACE PACKAGE PKG_GERENTE  
IS  
    PROCEDURE consultarVPreciosProductos(C_PRECIOSPRODUCTOS OUT SYS_REFCURSOR);  
    PROCEDURE consultarVComprasMensuales(C_COMPRASMENSUALES OUT SYS_REFCURSOR);  
    PROCEDURE consultarProduccionProveedor(C_PRODUCCIONPROVEEDOR OUT SYS_REFCURSOR);  
END;  
/
```

# ACTORES I

```
--(PACKAGE BODY)GERENTE :|  
  
CREATE OR REPLACE PACKAGE BODY PKG_GERENTE  
IS  
    PROCEDURE consultarVPreciosProductos(C_PRECIOSPRODUCTOS OUT SYS_REFCURSOR) IS  
    BEGIN  
        OPEN C_PRECIOSPRODUCTOS FOR SELECT * FROM PRECIOSPRODUCTOS;  
        COMMIT;  
    EXCEPTION  
    WHEN OTHERS THEN  
        ROLLBACK;  
        Raise_application_error(-20084,SQLERRM);  
    END consultarVPreciosProductos;  
  
    PROCEDURE consultarVComprasMensuales(C_COMPRASMENSUALES OUT SYS_REFCURSOR) IS  
    BEGIN  
        OPEN C_COMPRASMENSUALES FOR SELECT * FROM COMPRASMENSUALES;  
        COMMIT;  
    EXCEPTION  
    WHEN OTHERS THEN  
        ROLLBACK;  
        Raise_application_error(-20085,SQLERRM);  
    END consultarVComprasMensuales;
```

# SEGURIDAD

```
---DEFINICION DE ROLES Y PERMISOS:
```

```
---ROLES---:
```

```
CREATE ROLE GERENTES;  
CREATE ROLE COORDINADORDE_COMPRAS;
```

```
---PERMISOS--:
```

```
--1.Para 'GERENTE':
```

```
GRANT EXECUTE  
ON PKG_GERENTE  
TO GERENTE;
```

```
--2.Para 'COORDINADORDECOMPRAS':
```

```
GRANT EXECUTE  
ON PKG_COORDINADORDECOMPRAS  
TO COORDINADORDECOMPRAS;
```



DTD

```
<?xml version="1.0"?>
<!DOCTYPE Bienes [
  <!--ELEMENT Bienes (Producto | Servicio | Alquiler)*-->
  <!--ELEMENT Producto (Descripcion?, Componentes?, Recomendaciones?, Proveedores)-->
  <!--ELEMENT Descripcion (#PCDATA)-->
  <!--ELEMENT Componentes (Componente+)-->
  <!--ELEMENT Componente (#PCDATA)-->
  <!--ELEMENT Proveedores (Proveedor+)-->
  <!--ELEMENT Proveedor EMPTY-->
  <!--ELEMENT Servicio (Insumos, Recomendaciones+)-->
  <!--ELEMENT Insumos (Insumo+)-->
  <!--ELEMENT Insumo EMPTY-->
  <!--ELEMENT Recomendaciones (#PCDATA)-->
  <!--ELEMENT Alquiler EMPTY-->
  <!--ATTLIST Producto codigo ID #REQUIRED-->
  <!--ATTLIST Producto nombre CDATA #REQUIRED-->
  <!--ATTLIST Producto precioVenta CDATA #IMPLIED-->
  <!--ATTLIST Producto existencias CDATA #REQUIRED-->
  <!--ATTLIST Producto precioCompra CDATA #IMPLIED-->
  <!--ATTLIST Servicio codigo ID #REQUIRED-->
  <!--ATTLIST Servicio nombre CDATA #REQUIRED-->
  <!--ATTLIST Servicio precioVenta CDATA #IMPLIED-->
  <!--ATTLIST Servicio manoObra CDATA #IMPLIED-->
  <!--ATTLIST Insumo producto IDREF #REQUIRED-->
  <!--ATTLIST Insumo unidades CDATA #IMPLIED-->
  <!--ATTLIST Proveedor nit CDATA #REQUIRED-->
  <!--ATTLIST Proveedor precio CDATA #IMPLIED-->
  <!--ATTLIST Alquiler codigo ID #REQUIRED-->
  <!--ATTLIST Alquiler nombre CDATA #REQUIRED-->
  <!--ATTLIST Alquiler precioVenta CDATA #REQUIRED-->
  <!--ATTLIST Alquiler fecha CDATA #IMPLIED-->
--]
1>
```

XML

```
<Bienes>
  <Producto codigo="C25369" nombre="Líquido para vidrios" precioVenta="20000" existencias="10" precioCompra="18000">
  <Producto codigo="C493256" nombre="Jabón en polvo" precioVenta="2000" existencias="100" precioCompra="18000">
  <Producto codigo="C32014" nombre="Limpiador con caucho y esponja" precioVenta="50000">
  <Servicio codigo="LVEDA" nombre="Lavado vidrios edificios" precioVenta="500000" manoObra="100000">
  <Servicio codigo="LPISO" nombre="Limpieza pisos acrilicos" precioVenta="40000" manoObra="5000">
  <Alquiler codigo="ALQ001" nombre="Alquiler 1" precioVenta="100" fecha="10/10/2019"/>
</Bienes>
```

- <!DOCTYPE Bienes [ 1 2
- <!ELEMENT Bienes (Producto | Servicio )\*>
- <!ELEMENT Producto (Descripcion?, Componentes?, Recomendaciones?, Proveedores)>
- <!ELEMENT Descripcion (#PCDATA)>
- <!ELEMENT Componentes (Componente+)>
- <!ELEMENT Componente (#PCDATA)>
- <!ELEMENT Proveedores (Proveedor+)>
- <!ELEMENT Proveedor EMPTY>
- <!ATTLIST Producto codigo ID #REQUIRED>
- <!ATTLIST Producto nombre CDATA #REQUIRED>
- <!ATTLIST Producto precioVenta CDATA #IMPLIED>
- <!ATTLIST Producto existencias CDATA #REQUIRED>
- <!ATTLIST Producto precioCompra CDATA #IMPLIED>

## ELEMENTOS

## ATRIBUTOS

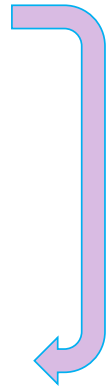
- <!DOCTYPE Bienes [
- <!ELEMENT Bienes (Producto | Servicio )\*>
- <!ELEMENT Producto (Descripcion?, Componentes?, Recomendaciones?, Proveedores) ...
- <!ATTLIST Producto codigo ID #REQUIRED>
- <!ATTLIST Producto nombre CDATA #REQUIRED>
- <!ATTLIST Producto precioVenta CDATA #IMPLIED>
- <!ATTLIST Producto existencias CDATA #REQUIRED>
- ]>

XML

```

<Bienes>
  <Producto codigo ="" nombre="" precioVenta="" existencias="">
    <Descripcion>
    </Descripcion>
    <Componentes>
    </Componentes>
  </Producto>
  ...
</Bienes>

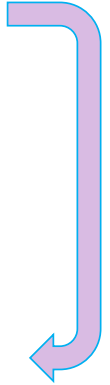
```



- <!DOCTYPE Bienes [
- <!ELEMENT Bienes (Producto | Servicio )\*>
- <!ELEMENT Producto (Descripcion?, Componentes?, Recomendaciones?, Proveedores)>
- ...

• XML  
 • ]>

```
<Bienes>
  <Producto>
    <Descripcion>
    </Descripcion>
    <Componentes>
    </Componentes>
  </Producto>
  <Servicio>
  </Servicio>
</Bienes>
```



- 1. Toda la información

XPATH

/Bienes/Producto[@codigo="C32014"]

```
Element='<Producto codigo="C32014"
        nombre="Limpiador con caucho y esponja"
        precioVenta="50000">
  <Componentes>
    <Componente> AGUA 930 c.c. </Componente>
    <Componente> COLOR - ANILINA (VEGETAL) 1 Gramo </Componente>
    <Componente> GENAPOL LRO 1 c.c. EDTA 1 c.c. </Componente>
    <Componente> MERGAL 1 c.c. </Componente>
    <Componente> BUTIL CELLOSOLVE 50 c.c. </Componente>
    <Componente> ALCOHOL ISOPROPILICO 20 c.c </Componente>
  </Componentes>
  <Proveedores>
    <Proveedor nit="123457" precio="100"/>
  </Proveedores>
</Producto>'
```

- 2. Todas las etiquetas

/Bienes/Producto[@codigo="C32014"]/\*

```
Element=<Componentes>
  <Componente> AGUA 930 c.c. </Componente>
  <Componente> COLOR - ANILINA (VEGETAL) 1 Gramo </Componente>
  <Componente> GENAPOL LRO 1 c.c. EDTA 1 c.c. </Componente>
  <Componente> MERGAL 1 c.c. </Componente>
  <Componente> BUTIL CELLOSOLVE 50 c.c. </Componente>
  <Componente> ALCOHOL ISOPROPILICO 20 c.c </Componente>
</Componentes>

Element=<Proveedores>
  <Proveedor nit="123457" precio="100"/>
</Proveedores>
```

- 3. Todas los atributos

/Bienes/Producto[@codigo="C32014"]/@\*

```
Attribute='codigo=C32014'
Attribute='nombre=Limpiador con caucho y esponja'
Attribute='precioVenta=50000'
```



- 4. Ejemplos adicionales

/Bienes/Producto/Descripcion

/Bienes/Producto/Componentes

/Bienes/Servicio/Insumos/Insumo/@producto

/Bienes/Servicio/Insumos/Insumo/@\*

/Bienes/Servicio/Insumos/Insumo/\*

/Bienes/(Servicio/Insumos/\*)| (Producto/Componentes/\*)

/Bienes/(Servicio/Insumos)| (Producto/Componentes)/\*

/Bienes/(Servicio | Producto) /@nombre

- 5. XML en Oracle

- Tipo de dato: XMLType

```
CREATE TABLE po_xml_tab(  
  poId number,  
  poDoc XMLTYPE); ←
```

```
INSERT INTO warehouses VALUES  
  (  
    100, XMLType(  
      '<Warehouse whNo="100">  
        <Building>Owned</Building>  
      </Warehouse>'), 'Tower Records', 1003);
```

- 5. XML en Oracle

- Tipo de dato: XMLType

```
UPDATE warehouses SET warehouse_spec = XMLType
    ('<Warehouse whono="200">
      <Building>Leased</Building>
    </Warehouse>');
```

```
SELECT
    w.warehouse_spec.extract('/Warehouse/Building/text()').getStringVal()
    "Building"
FROM warehouses w;
```