

BASES DE DATOS 2

Proyecto #1 – Sistema de Emisión de Cotizaciones para ensamblaje de equipos electrónicos

Abstract

The purpose of this research is to increase our knowledge about data modelling using the Oracle 12c tools and taking advantage of its powerful and fully customizable platform. In this document we will describe the solution we created. After that we included some conclusions.

Profesor

Msc. José A. Stradi Granados

Estudiantes

Marcos Calderón Badilla - 200927239
Kevin Alonso Escobar Miranda - 201265615
Moisés Ernesto Viales Espinoza - 201238614

Fecha

Viernes 24 de Abril 2015

Contenido

Introducción.....	3
Descripción del problema	3
Diagrama de clases y de base de datos	3
Diagrama de base de datos	4
Diagrama de clases	5
Diccionario de datos	6
T_ARTICULOESTADOS	6
T_ARTICULOS	6
T_ARTICULOSCOTIZADOS	6
T_ARTICULOSCOTIZADOSHISTO	7
T_BITACORA.....	7
T_COMPONENTES.....	7
T_CONTACTO	7
T_COTIZACIONES	8
T_COTIZACIONESHISTO	8
T_COTIZACIONESESTADOS	8
T_CIUDADES.....	8
T_DETALLEDIRECCION	9
T_EMPRESA.....	9
T_ESTADOSTIPOCAMBIO	9
T_FACTURACOMPRAS.....	9
T_FAMILIAS	9
T_INDICESECONOMICOS.....	10
T_LINEASFACTURASCOMPRAS	10
T_LISTAPRECIOS.....	10
T_LISTAPRECIOXARTICULOS	10
T_MARCAS	10
T_MESANIO.....	10
T_MONEDAS	10

T_PAISES	11
T_PERMISOSUSUARIOS.....	11
T_PROVINCIAS	11
T_PROYECCIONESLISTAPRECIOS	11
T_TIPOCAMBIOS	11
T_TIPOCONTACTOS.....	11
T_TIPOEMPRESA	12
T_TIPOERRORES.....	12
T_TIPOUSUARIOS.....	12
T_UNIDADMEDIDAS.....	12
T_USUARIOS	12
Conclusiones	13
Puntos concluidos	13
Puntos pendientes	13
Experiencia adquirida	13
Recomendaciones.....	13
División de tareas.....	14
Referencias Bibliográficas	14
Anexo	15
Scripts	15
Creación de las tablas	15
Script de llenado	37
Paquetes con procedimientos almacenados (Stored procedures)	43
Procedimientos almacenados (Stored procedures)	51

Introducción

Actualmente, las bases de datos están teniendo un impacto decisivo sobre el creciente uso de las computadoras. Debido al avance de las generaciones de las computadoras, las bases de datos no se quedan atrás, tuvieron que avanzar junto con los lenguajes de programación y las necesidades de los programadores para el almacenaje de datos y actualmente las bases de datos se han convertido en un tipo de lenguaje multiusuario.

La evolución de las bases de datos está en incremento, actualmente nos encontramos en la tercera generación, muchos expertos en la materia coinciden en que el incremento de las bases de datos está en aumento día con día, en todas las áreas profesionales ya que todo es información, en las áreas como la investigación, tecnología, arte, educación, sistemas médicos, programas de ingeniería de software, programas de desarrollo, de diseño, sistemas de información geográfica, entre otros. El presente proyecto trata con uno de los motores de bases de datos más conocidos y exitosos, como lo es Oracle, y con varias herramientas asociadas que permiten un desarrollo más completo de aplicaciones, como lo son jdeveloper, weblogic y sqldeveloper.

Descripción del problema

Para el desarrollo de esta tarea programada se requiere el desarrollo de una base de datos de un sistema para la emisión de cotizaciones de una empresa dedicada al ensamblaje de equipos electrónicos. Se debe llevar el registro de clientes, proveedores, los artículos ofrecidos por o a ellos, así como también calcular los precios de los productos cotizados considerando el comportamiento de índices económicos.

Las tecnologías a utilizar para el desarrollo de este proyecto son las siguientes: Oracle 12c, Java y WebLogic

Además se debe llevar a cabo el cálculo de las proyecciones de los tipos de cambios según los tipos cambios de meses anteriores, esto con el fin de poder calcular los precios de productos cotizados como antes se mencionó.

Diagrama de clases y de base de datos

En esta sección se muestran las imágenes de los diagramas que se utilizaron para el desarrollo del proyecto.

Diagrama de base de datos

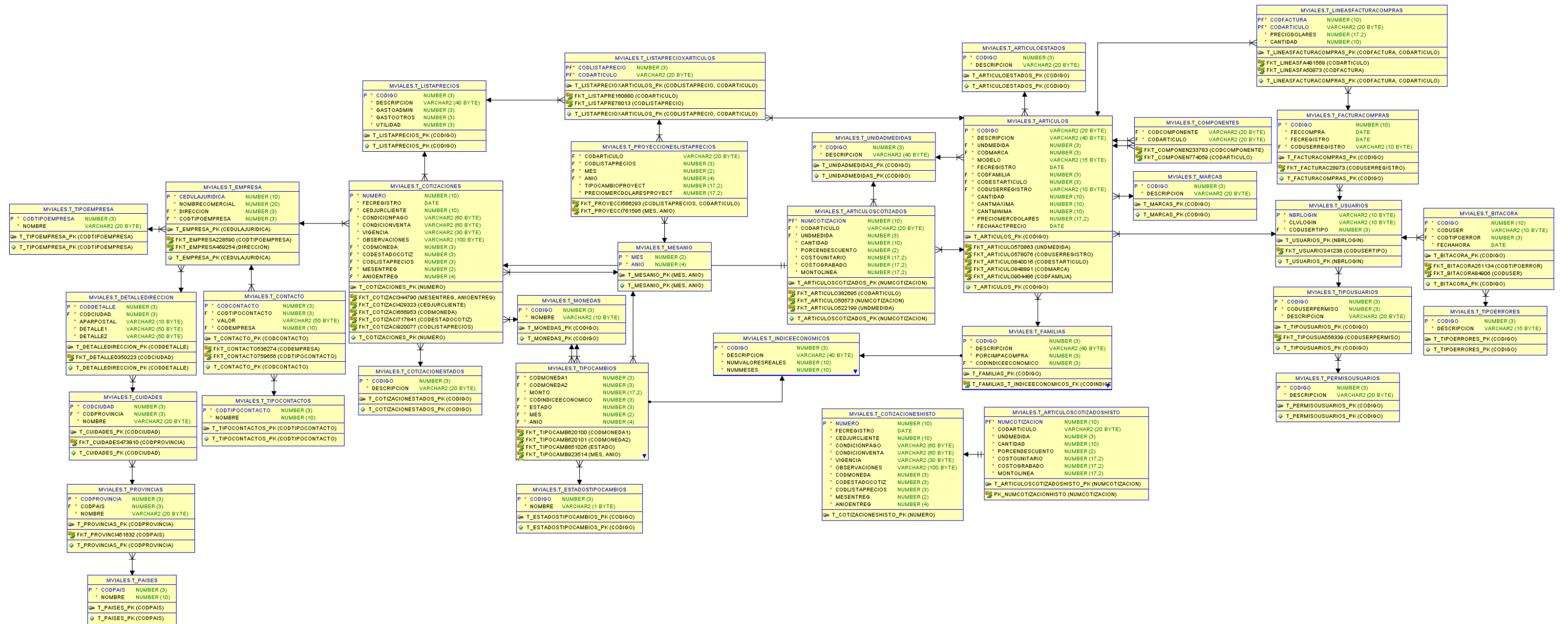
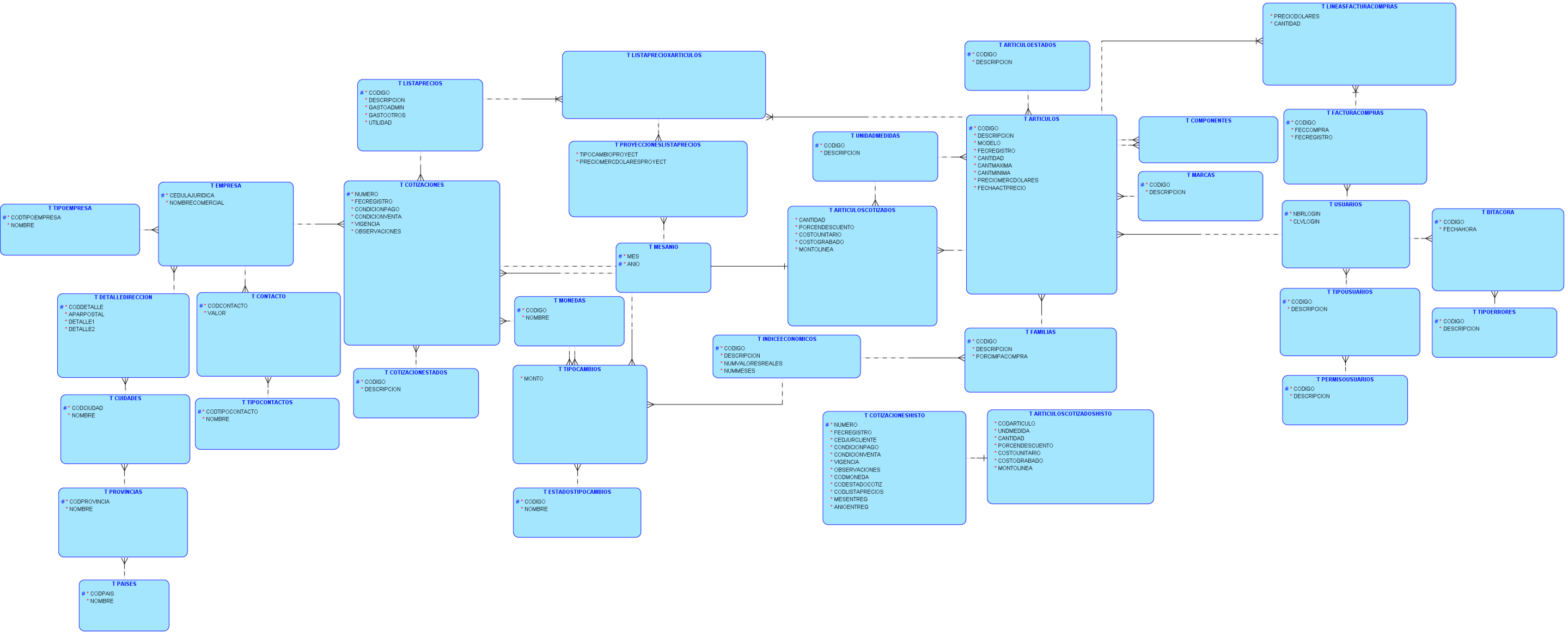


Diagrama de clases



Diccionario de datos

T_ARTICULOESTADOS

❖ COLUMN_NAME	❖ DATA_TYPE	❖ NULLABLE	DATA_DEFAULT	❖ COLUMN_ID
CODIGO	NUMBER(3,0)	No	(null)	1
DESCRIPCION	VARCHAR2(20 BYTE)	No	(null)	2

T_ARTICULOS

❖ COLUMN_NAME	❖ DATA_TYPE	❖ NULLABLE	DATA_DEFAULT	❖ COLUMN_ID
CODIGO	VARCHAR2(20 BYTE)	No	(null)	1
DESCRIPCION	VARCHAR2(40 BYTE)	No	(null)	2
UNDMEDIDA	NUMBER(3,0)	No	(null)	3
CODMARCA	NUMBER(3,0)	No	(null)	4
MODELO	VARCHAR2(15 BYTE)	No	(null)	5
FECREGISTRO	DATE	No	(null)	6
CODFAMILIA	NUMBER(3,0)	No	(null)	7
CODESTARTICULO	NUMBER(3,0)	No	(null)	8
CODUSERREGISTRO	VARCHAR2(10 BYTE)	No	(null)	9
CANTIDAD	NUMBER(10,0)	No	(null)	10
CANTMAXIMA	NUMBER(10,0)	No	(null)	11
CANTMINIMA	NUMBER(10,0)	No	(null)	12
PRECIOMERCDOLARES	NUMBER(17,2)	No	(null)	13
FECHAACTPRECIO	DATE	No	(null)	14

T_ARTICULOSCOTIZADOS

❖ COLUMN_NAME	❖ DATA_TYPE	❖ NULLABLE	DATA_DEFAULT	❖ COLUMN_ID
NUMCOTIZACION	NUMBER(10,0)	No	(null)	1
CODARTICULO	VARCHAR2(20 BYTE)	No	(null)	2
UNDMEDIDA	NUMBER(3,0)	No	(null)	3
CANTIDAD	NUMBER(10,0)	No	(null)	4
PORCENDESCUENTO	NUMBER(2,0)	No	(null)	5
COSTOUNITARIO	NUMBER(17,2)	No	(null)	6
COSTOGRABADO	NUMBER(17,2)	No	(null)	7
MONTOLINEA	NUMBER(17,2)	No	(null)	8

T_ARTICULOSCOTIZADOSHISTO

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
NUMCOTIZACION	NUMBER(10,0)	No	(null)	1
CODARTICULO	VARCHAR2(20 BYTE)	No	(null)	2
UNDMEDIDA	NUMBER(3,0)	No	(null)	3
CANTIDAD	NUMBER(10,0)	No	(null)	4
PORCENDESCUENTO	NUMBER(2,0)	No	(null)	5
COSTOUNITARIO	NUMBER(17,2)	No	(null)	6
COSTOGRABADO	NUMBER(17,2)	No	(null)	7
MONTOLINEA	NUMBER(17,2)	No	(null)	8

T_BITACORA

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
CODIGO	NUMBER(10,0)	No	(null)	1
CODUSER	VARCHAR2(10 BYTE)	No	(null)	2
CODTIPOERROR	NUMBER(3,0)	No	(null)	3
FECHAHORA	DATE	No	(null)	4

T_COMPONENTES

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
CODCOMPONENTE	VARCHAR2(20 BYTE)	No	(null)	1
CODARTICULO	VARCHAR2(20 BYTE)	No	(null)	2

T_CONTACTO

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
CODCONTACTO	NUMBER(3,0)	No	(null)	1
CODTIPOCONTACTO	NUMBER(3,0)	No	(null)	2
VALOR	VARCHAR2(50 BYTE)	No	(null)	3
CODEMPRESA	NUMBER(10,0)	No	(null)	4

T_COTIZACIONES

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
NUMERO	NUMBER(10,0)	No	(null)	1
FECREGISTRO	DATE	No	(null)	2
CEDJURCLIENTE	NUMBER(10,0)	No	(null)	3
CONDICIONPAGO	VARCHAR2(60 BYTE)	No	(null)	4
CONDICIONVENTA	VARCHAR2(60 BYTE)	No	(null)	5
VIGENCIA	VARCHAR2(30 BYTE)	No	(null)	6
OBSERVACIONES	VARCHAR2(100 BYTE)	No	(null)	7
CODMONEDA	NUMBER(3,0)	No	(null)	8
CODESTADOCOTIZ	NUMBER(3,0)	No	(null)	9
CODLISTAPRECIOS	NUMBER(3,0)	No	(null)	10
MESENTREG	NUMBER(2,0)	No	(null)	11
ANIOENTREG	NUMBER(4,0)	No	(null)	12

T_COTIZACIONESHISTO

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
NUMERO	NUMBER(10,0)	No	(null)	1
FECREGISTRO	DATE	No	(null)	2
CEDJURCLIENTE	NUMBER(10,0)	No	(null)	3
CONDICIONPAGO	VARCHAR2(60 BYTE)	No	(null)	4
CONDICIONVENTA	VARCHAR2(60 BYTE)	No	(null)	5
VIGENCIA	VARCHAR2(30 BYTE)	No	(null)	6
OBSERVACIONES	VARCHAR2(100 BYTE)	No	(null)	7
CODMONEDA	NUMBER(3,0)	No	(null)	8
CODESTADOCOTIZ	NUMBER(3,0)	No	(null)	9
CODLISTAPRECIOS	NUMBER(3,0)	No	(null)	10
MESENTREG	NUMBER(2,0)	No	(null)	11
ANIOENTREG	NUMBER(4,0)	No	(null)	12

T_COTIZACIONESESTADOS

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
CODIGO	NUMBER(3,0)	No	(null)	1
DESCRIPCION	VARCHAR2(20 BYTE)	No	(null)	2

T_CIUDADES

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
CODCIUDAD	NUMBER(3,0)	No	(null)	1
CODPROVINCIA	NUMBER(3,0)	No	(null)	2
NOMBRE	VARCHAR2(20 BYTE)	No	(null)	3

T_DETALLE DIRECCION

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
CODDETALLE	NUMBER(3,0)	No	(null)	1
CODCIUDAD	NUMBER(3,0)	No	(null)	2
APARPOSTAL	VARCHAR2(10 BYTE)	No	(null)	3
DETALLE1	VARCHAR2(50 BYTE)	No	(null)	4
DETALLE2	VARCHAR2(50 BYTE)	No	(null)	5

T_EMPRESA

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
CEDULAJURIDICA	NUMBER(10,0)	No	(null)	1
NOMBRECOMERCIAL	NUMBER(20,0)	No	(null)	2
DIRECCION	NUMBER(3,0)	No	(null)	3
CODTIPOEMPRESA	NUMBER(3,0)	No	(null)	4

T_ESTADOSTIPOCAMBIO

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
CODIGO	NUMBER(3,0)	No	(null)	1
NOMBRE	VARCHAR2(1 BYTE)	No	(null)	2

T_FACTURACOMPRAS

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
CODIGO	NUMBER(10,0)	No	(null)	1
FECCOMPRA	DATE	No	(null)	2
FECREGISTRO	DATE	No	(null)	3
CODUSERREGISTRO	VARCHAR2(10 BYTE)	No	(null)	4

T_FAMILIAS

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
CODIGO	NUMBER(3,0)	No	(null)	1
DESCRIPCION	VARCHAR2(40 BYTE)	No	(null)	2
PORCIMPACOMPRA	NUMBER(3,0)	No	(null)	3
CODINDICEECONOMICO	NUMBER(3,0)	No	(null)	4

T_INDICESECONOMICOS

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
CODIGO	NUMBER(3,0)	No	(null)	1
DESCRIPCION	VARCHAR2(40 BYTE)	No	(null)	2
NUMVALORESREALES	NUMBER(10,0)	No	(null)	3
NUMMESES	NUMBER(10,0)	No	(null)	4

T_LINEASFACTURASCOMPRAS

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
CODFACTURA	NUMBER(10,0)	No	(null)	1
CODARTICULO	VARCHAR2(20 BYTE)	No	(null)	2
PRECIODOLARES	NUMBER(17,2)	No	(null)	3
CANTIDAD	NUMBER(10,0)	No	(null)	4

T_LISTAPRECIOS

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
CODIGO	NUMBER(3,0)	No	(null)	1
DESCRIPCION	VARCHAR2(40 BYTE)	No	(null)	2
GASTOADMIN	NUMBER(3,0)	No	(null)	3
GASTOOTROS	NUMBER(3,0)	No	(null)	4
UTILIDAD	NUMBER(3,0)	No	(null)	5

T_LISTAPRECIOXARTICULOS

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
CODLISTAPRECIO	NUMBER(3,0)	No	(null)	1
CODARTICULO	VARCHAR2(20 BYTE)	No	(null)	2

T_MARCAS

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
CODIGO	NUMBER(3,0)	No	(null)	1
DESCRIPCION	VARCHAR2(20 BYTE)	No	(null)	2

T_MESANIO

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
MES	NUMBER(2,0)	No	(null)	1
ANIO	NUMBER(4,0)	No	(null)	2

T_MONEDAS

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
CODIGO	NUMBER(3,0)	No	(null)	1
NOMBRE	VARCHAR2(10 BYTE)	No	(null)	2

T_PAISES

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
CODPAIS	NUMBER(3,0)	No	(null)	1
NOMBRE	NUMBER(10,0)	No	(null)	2

T_PERMISOSUSUARIOS

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
CODIGO	NUMBER(3,0)	No	(null)	1
DESCRIPCION	VARCHAR2(20 BYTE)	No	(null)	2

T_PROVINCIAS

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
CODPROVINCIA	NUMBER(3,0)	No	(null)	1
CODPAIS	NUMBER(3,0)	No	(null)	2
NOMBRE	VARCHAR2(20 BYTE)	No	(null)	3

T_PROYECCIONESLISTAPRECIOS

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
CODARTICULO	VARCHAR2(20 BYTE)	No	(null)	1
CODLISTAPRECIOS	NUMBER(3,0)	No	(null)	2
MES	NUMBER(2,0)	No	(null)	3
ANIO	NUMBER(4,0)	No	(null)	4
TIPOCAMBIOPROYECT	NUMBER(17,2)	No	(null)	5
PRECIOMERCDOLARESPROYECT	NUMBER(17,2)	No	(null)	6

T_TIPOCAMBIOS

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
CODMONEDA1	NUMBER(3,0)	No	(null)	1
CODMONEDA2	NUMBER(3,0)	No	(null)	2
MONTO	NUMBER(17,2)	No	(null)	3
CODINDICEECONOMICO	NUMBER(3,0)	No	(null)	4
ESTADO	NUMBER(3,0)	No	(null)	5
MES	NUMBER(2,0)	No	(null)	6
ANIO	NUMBER(4,0)	No	(null)	7

T_TIPOCONTACTOS

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
CODTIPOCONTACTO	NUMBER(3,0)	No	(null)	1
NOMBRE	NUMBER(10,0)	No	(null)	2

T_TIPOEMPRESA

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
CODTIPOEMPRESA	NUMBER(3,0)	No	(null)	1
NOMBRE	VARCHAR2(20 BYTE)	No	(null)	2

T_TIPOERRORES

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
CODIGO	NUMBER(3,0)	No	(null)	1
DESCRIPCION	VARCHAR2(15 BYTE)	No	(null)	2

T_TIPOUSUARIOS

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
CODIGO	NUMBER(3,0)	No	(null)	1
CODUSERPERMISO	NUMBER(3,0)	No	(null)	2
DESCRIPCION	VARCHAR2(20 BYTE)	No	(null)	3

T_UNIDADMEDIDAS

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
CODIGO	NUMBER(3,0)	No	(null)	1
DESCRIPCION	VARCHAR2(40 BYTE)	No	(null)	2

T_USUARIOS

↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID
NBRLOGIN	VARCHAR2(10 BYTE)	No	(null)	1
CLVLOGIN	VARCHAR2(10 BYTE)	No	(null)	2
CODUSERTIPO	NUMBER(3,0)	No	(null)	3

Conclusiones

Se logró completar la mayoría del proyecto cumpliendo con los objetivos inmersos dentro de este.

Puntos concluidos

- Se efectúa el diseño de la base de datos bajo el motor de Oracle 12c
- Se crea y se pone en funcionamiento un Web service con servicios SOAP
- Se desarrolla un cliente en java el cual es unido utilizando la herramienta jdeveloper
- Se crean los Stored Procedures solicitados bajo instrucciones PL/SQL y estos son almacenados dentro de un Package
- Se desarrollan los Triggers solicitados de manera correcta y bajo el mismo esquema de instrucciones PL/SQL
- Se crean procedimientos y funciones para satisfacer el requerimiento de obtención de resultados para las consultas solicitadas

Puntos pendientes

- Prueba del sistema: No se han probado todos los casos de uso. Por lo tanto, no se puede asegurar el correcto funcionamiento de los procedimientos almacenados y triggers.

Experiencia adquirida

- Reforzamiento de los conocimientos de diseño relacional
- Reforzamiento de los conocimientos de web services
- Reforzamiento de los conocimientos en el lenguaje de programación orientada a objetos en Java
- Aprendizaje de Oracle 12c
- Aprendizaje de WebLogic
- Aprendizaje de PL/SQL a mayor profundidad

Recomendaciones

Para futuras implementaciones de proyectos utilizando herramientas como las utilizadas en este proyecto se recomienda lo siguiente:

- Realizar un buen diseño de la base de datos de manera que este sea completo y abarque todas las necesidades requeridas de manera que se faciliten las consultas y operaciones sobre la base de datos.

- Configurar de manera minuciosa el origen de los servicios dentro del weblogic de manera que los nombres y opciones de configuración sean la misma dentro del servidor y dentro del web service para facilitar que éstos se sincronicen correctamente.

División de tareas

Estudiante	Documentación	Diseño de base de datos	Script SQL	Codificación del cliente
Marcos Calderón Badilla	33.3%	20%	40%	33.3%
Kevin Alonso Escobar Miranda	33.3%	40%	20%	33.3%
Moisés Ernesto Viales Espinoza	33.3%	40%	40%	33.3%
Total:	100%	100%	100%	100%

Referencias Bibliográficas

- A Data Type Conversion. (n.d.). Recuperado April 25, 2015, de http://docs.oracle.com/cd/B19306_01/gateways.102/b14270/apa.htm
- Solorio González Yadira. (2008, junio 4). Tendencias en sistemas de información y bases de datos. Recuperado de <http://www.gestiopolis.com/tendencias-sistemas-informacion-bases-datos/>

Anexo

En esta sección se adjuntan todos los scripts utilizados para la creación de la base de datos, para introducir los datos básicos, procedimientos almacenados y disparadores necesarios para el funcionamiento de la aplicación

Scripts

Creación de las tablas

```
CREATE TABLE MVIALES.T_LISTAPRECIOXARTICULOS

(
    CODLISTAPRECIO NUMBER (3) NOT NULL ,
    CODARTICULO          VARCHAR2 (20 BYTE) NOT NULL
)

PCTFREE 10 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE

(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
BUFFER_POOL DEFAULT
) ;

CREATE UNIQUE INDEX MVIALES.T_LISTAPRECIOXARTICULOS_PK ON MVIALES.T_LISTAPRECIOXARTICULOS

(
    CODLISTAPRECIO ASC , CODARTICULO ASC
)

TABLESPACE PROYECTO PCTFREE 10 STORAGE

(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
BUFFER_POOL DEFAULT
)

LOGGING ;

ALTER TABLE MVIALES.T_LISTAPRECIOXARTICULOS ADD CONSTRAINT T_LISTAPRECIOXARTICULOS_PK
PRIMARY KEY ( CODLISTAPRECIO, CODARTICULO ) USING INDEX MVIALES.T_LISTAPRECIOXARTICULOS_PK
;

CREATE TABLE MVIALES.T_LISTAPRECIOS

(
    CODIGO          NUMBER (3) NOT NULL ,
    DESCRIPCION VARCHAR2 (40 BYTE) NOT NULL ,
    GASTOADMIN      NUMBER (3) NOT NULL ,
```



```

    GASTOOTROS    NUMBER (3) NOT NULL ,
    UTILIDAD      NUMBER (3) NOT NULL
)
PCTFREE 10 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE
(
    INITIAL 1048576 NEXT 1048576 PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS 2147483645 FREELISTS
1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT
) ;
CREATE UNIQUE INDEX MVIALES.T_LISTAPRECIOS_PK ON MVIALES.T_LISTAPRECIOS
(
    CODIGO ASC
)
TABLESPACE PROYECTO PCTFREE 10 STORAGE
(
    INITIAL 1048576 NEXT 1048576 PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS 2147483645 FREELISTS
1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT
)
LOGGING ;
ALTER TABLE MVIALES.T_LISTAPRECIOS ADD CONSTRAINT T_LISTAPRECIOS_PK PRIMARY KEY ( CODIGO
) USING INDEX MVIALES.T_LISTAPRECIOS_PK ;

CREATE TABLE MVIALES.T_PROYECCIONESLISTAPRECIOS
(
    CODARTICULO          VARCHAR2 (20 BYTE) NOT NULL ,
    CODLISTAPRECIOS      NUMBER (3) NOT NULL ,
    MES                  NUMBER (2) NOT NULL ,
    ANIO                 NUMBER (4) NOT NULL ,
    TIPOCAMBIOPROYECT    NUMBER (17,2) NOT NULL ,
    PRECIOMERCDOLARESPROYECT NUMBER(17,2) NOT NULL
)
PCTFREE 10 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE
(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
    BUFFER_POOL DEFAULT
) ;

CREATE TABLE MVIALES.T_ARTICULOESTADOS
(
    CODIGO              NUMBER (3) NOT NULL ,

```

```

        DESCRIPCION VARCHAR2 (20 BYTE) NOT NULL
    )

    PCTFREE 10 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE

    (
        INITIAL 1048576 NEXT 1048576 PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS 2147483645 FREELISTS
        1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT
    ) ;

CREATE UNIQUE INDEX MVIALES.T_ARTICULOESTADOS_PK ON MVIALES.T_ARTICULOESTADOS

    (
        CODIGO ASC
    )

    TABLESPACE PROYECTO PCTFREE 10 STORAGE

    (
        INITIAL 1048576 NEXT 1048576 PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS 2147483645 FREELISTS
        1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT
    )

    LOGGING ;

ALTER TABLE MVIALES.T_ARTICULOESTADOS ADD CONSTRAINT T_ARTICULOESTADOS_PK PRIMARY KEY (
    CODIGO ) USING INDEX MVIALES.T_ARTICULOESTADOS_PK ;

CREATE TABLE MVIALES.T_ARTICULOSCOTIZADOS

    (
        NUMCOTIZACION    NUMBER (10) NOT NULL ,
        CODARTICULO       VARCHAR2 (20 BYTE) NOT NULL ,
        UNDMEDIDA         NUMBER (3) NOT NULL ,
        CANTIDAD          NUMBER (10) NOT NULL ,
        PORCENDESCUENTO  NUMBER (2) NOT NULL ,
        COSTOUNITARIO     NUMBER (17,2) NOT NULL ,
        COSTOGRABADO      NUMBER (17,2) NOT NULL ,
        MONTOLINEA        NUMBER (17,2) NOT NULL
    )

    PCTFREE 10 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE

    (
        PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
        BUFFER_POOL DEFAULT
    ) ;

CREATE UNIQUE INDEX MVIALES.T_ARTICULOSCOTIZADOS_PK ON MVIALES.T_ARTICULOSCOTIZADOS

    (
        NUMCOTIZACION ASC

```

```

)

TABLESPACE PROYECTO PCTFREE 10 STORAGE

(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
    BUFFER_POOL DEFAULT
)

LOGGING ;

ALTER TABLE MVIALES.T_ARTICULOSCOTIZADOS ADD CONSTRAINT T_ARTICULOSCOTIZADOS_PK PRIMARY
KEY ( NUMCOTIZACION ) USING INDEX MVIALES.T_ARTICULOSCOTIZADOS_PK ;

CREATE TABLE MVIALES.T_ARTICULOSCOTIZADOSHISTO

(
    NUMCOTIZACION    NUMBER (10) NOT NULL ,
    CODARTICULO       VARCHAR2 (20 BYTE) NOT NULL ,
    UNDMEDIDA         NUMBER (3) NOT NULL ,
    CANTIDAD          NUMBER (10) NOT NULL ,
    PORCENDESCUENTO  NUMBER (2) NOT NULL ,
    COSTOUNITARIO     NUMBER (17,2) NOT NULL ,
    COSTOGRABADO      NUMBER (17,2) NOT NULL ,
    MONTOLINEA        NUMBER (17,2) NOT NULL
)

PCTFREE 28 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE

(
    INITIAL 245760 NEXT 122880 PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS 2147483645 FREELISTS 1
    FREELIST GROUPS 1 BUFFER_POOL DEFAULT
) ;

ALTER TABLE MVIALES.T_ARTICULOSCOTIZADOSHISTO ADD CONSTRAINT T_ARTICULOSCOTIZADOSHISTO_PK
PRIMARY KEY ( NUMCOTIZACION ) USING INDEX PCTFREE 10 INITRANS 2 TABLESPACE PROYECTO LOGGING
STORAGE ( PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
    BUFFER_POOL DEFAULT ) ;

CREATE TABLE MVIALES.T_BITACORA

(
    CODIGO           NUMBER (10) NOT NULL ,
    CODUSER          VARCHAR2 (10 BYTE) NOT NULL ,
    CODTIPOERROR     NUMBER (3) NOT NULL ,
    FECHAHORA        DATE NOT NULL
)

PCTFREE 10 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE

(

```

```

        PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
BUFFER_POOL DEFAULT

    ) ;

CREATE UNIQUE INDEX MVIALES.T_BITACORA_PK ON MVIALES.T_BITACORA

    (

        CODIGO ASC

    )

    TABLESPACE PROYECTO PCTFREE 10 STORAGE

    (

        PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
BUFFER_POOL DEFAULT

    )

    LOGGING ;

ALTER TABLE MVIALES.T_BITACORA ADD CONSTRAINT T_BITACORA_PK PRIMARY KEY ( CODIGO ) USING
INDEX MVIALES.T_BITACORA_PK ;


CREATE TABLE MVIALES.T_CONTACTO

    (

        CODCONTACTO          NUMBER (3) NOT NULL ,

        CODTIPOCONTACTO NUMBER (3) NOT NULL ,

        VALOR                 VARCHAR2 (50 BYTE) NOT NULL ,

        CODEMPRESA           NUMBER (10) NOT NULL

    )

    PCTFREE 10 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE

    (

        PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
BUFFER_POOL DEFAULT

    ) ;

CREATE UNIQUE INDEX MVIALES.T_CONTACTO_PK ON MVIALES.T_CONTACTO

    (

        CODCONTACTO ASC

    )

    TABLESPACE PROYECTO PCTFREE 10 STORAGE

    (

        PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
BUFFER_POOL DEFAULT

    )

    LOGGING ;

ALTER TABLE MVIALES.T_CONTACTO ADD CONSTRAINT T_CONTACTO_PK PRIMARY KEY ( CODCONTACTO )
USING INDEX MVIALES.T_CONTACTO_PK ;

```

```

CREATE TABLE MVIALES.T_COTIZACIONES
(
    NUMERO                NUMBER (10) NOT NULL ,
    FECREGISTRO           DATE NOT NULL ,
    CEDJURCLIENTE        NUMBER (10) NOT NULL ,
    CONDICIONPAGO          VARCHAR2 (60 BYTE) NOT NULL ,
    CONDICIONVENTA         VARCHAR2 (60 BYTE) NOT NULL ,
    VIGENCIA              VARCHAR2 (30 BYTE) NOT NULL ,
    OBSERVACIONES          VARCHAR2 (100 BYTE) NOT NULL ,
    CODMONEDA             NUMBER (3) NOT NULL ,
    CODESTADOCOTIZ        NUMBER (3) NOT NULL ,
    CODLISTAPRECIOS        NUMBER (3) NOT NULL ,
    MESENTREG             NUMBER (2) NOT NULL ,
    ANIOENTREG            NUMBER (4) NOT NULL
)
PCTFREE 10 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE
(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
    BUFFER_POOL DEFAULT
) ;

CREATE UNIQUE INDEX MVIALES.T_COTIZACIONES_PK ON MVIALES.T_COTIZACIONES
(
    NUMERO ASC
)
TABLESPACE PROYECTO PCTFREE 10 STORAGE
(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
    BUFFER_POOL DEFAULT
)
LOGGING ;

ALTER TABLE MVIALES.T_COTIZACIONES ADD CONSTRAINT T_COTIZACIONES_PK PRIMARY KEY ( NUMERO
) USING INDEX MVIALES.T_COTIZACIONES_PK ;

CREATE TABLE MVIALES.T_COTIZACIONESHISTO
(
    NUMERO                NUMBER (10) NOT NULL ,
    FECREGISTRO           DATE NOT NULL ,

```

```

CEDJURCLIENTE    NUMBER (10) NOT NULL ,
CONDICIONPAGO    VARCHAR2 (60 BYTE) NOT NULL ,
CONDICIONVENTA   VARCHAR2 (60 BYTE) NOT NULL ,
VIGENCIA         VARCHAR2 (30 BYTE) NOT NULL ,
OBSERVACIONES    VARCHAR2 (100 BYTE) NOT NULL ,
CODMONEDA        NUMBER (3) NOT NULL ,
CODESTADOCOTIZ   NUMBER (3) NOT NULL ,
CODLISTAPRECIOS  NUMBER (3) NOT NULL ,
MESENTREG        NUMBER (2) NOT NULL ,
ANIOENTREG       NUMBER (4) NOT NULL
)
PCTFREE 28 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE
(
    INITIAL 245760 NEXT 122880 PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS 2147483645 FREELISTS 1
    FREELIST GROUPS 1 BUFFER_POOL DEFAULT
) ;

ALTER TABLE MVIALES.T_COTIZACIONESHISTO ADD CONSTRAINT T_COTIZACIONESHISTO_PK PRIMARY KEY (
    NUMERO ) USING INDEX PCTFREE 10 INITRANS 2 TABLESPACE PROYECTO LOGGING STORAGE (
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL
    DEFAULT ) ;

CREATE TABLE MVIALES.T_COTIZACIONESTADOS
(
    CODIGO        NUMBER (3) NOT NULL ,
    DESCRIPCION   VARCHAR2 (20 BYTE) NOT NULL
)
PCTFREE 10 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE
(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
    BUFFER_POOL DEFAULT
) ;

CREATE UNIQUE INDEX MVIALES.T_COTIZACIONESTADOS_PK ON MVIALES.T_COTIZACIONESTADOS
(
    CODIGO ASC
)
TABLESPACE PROYECTO PCTFREE 10 STORAGE
(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
    BUFFER_POOL DEFAULT
)

```

```

LOGGING ;

ALTER TABLE MVIALES.T_COTIZACIONESTADOS ADD CONSTRAINT T_COTIZACIONESTADOS_PK PRIMARY KEY
( CODIGO ) USING INDEX MVIALES.T_COTIZACIONESTADOS_PK ;

CREATE TABLE MVIALES.T_CUIDADES

(
    CODCIUDAD  NUMBER (3) NOT NULL ,
    CODPROVINCIA NUMBER (3) NOT NULL ,
    NOMBRE      VARCHAR2 (20 BYTE) NOT NULL
)

PCTFREE 10 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE

(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
    BUFFER_POOL DEFAULT
) ;

CREATE UNIQUE INDEX MVIALES.T_CUIDADES_PK ON MVIALES.T_CUIDADES

(
    CODCIUDAD ASC
)

TABLESPACE PROYECTO PCTFREE 10 STORAGE

(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
    BUFFER_POOL DEFAULT
)

LOGGING ;

ALTER TABLE MVIALES.T_CUIDADES ADD CONSTRAINT T_CUIDADES_PK PRIMARY KEY ( CODCIUDAD )
USING INDEX MVIALES.T_CUIDADES_PK ;

CREATE TABLE MVIALES.T_DETALLELDIRECCION

(
    CODDETALLE NUMBER (3) NOT NULL ,
    CODCIUDAD  NUMBER (3) NOT NULL ,
    APARPOSTAL VARCHAR2 (10 BYTE) NOT NULL ,
    DETALLE1    VARCHAR2 (50 BYTE) NOT NULL ,
    DETALLE2    VARCHAR2 (50 BYTE) NOT NULL
)

PCTFREE 10 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE

(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1

```

```

BUFFER_POOL DEFAULT

) ;

CREATE UNIQUE INDEX MVIALES.T_DETALLEDDIRECCION_PK ON MVIALES.T_DETALLEDDIRECCION

(
    CODDETALLE ASC
)

TABLESPACE PROYECTO PCTFREE 10 STORAGE

(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
BUFFER_POOL DEFAULT

)

LOGGING ;

ALTER TABLE MVIALES.T_DETALLEDDIRECCION ADD CONSTRAINT T_DETALLEDDIRECCION_PK PRIMARY KEY (
CODDETALLE ) USING INDEX MVIALES.T_DETALLEDDIRECCION_PK ;

CREATE TABLE MVIALES.T_EMPRESA

(
    CEDULAJURIDICA NUMBER (10) NOT NULL ,
    NOMBRECOMERCIAL NUMBER (20) NOT NULL ,
    DIRECCION          NUMBER (3) NOT NULL ,
    CODTIPOEMPRESA NUMBER (3) NOT NULL
)

PCTFREE 10 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE

(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
BUFFER_POOL DEFAULT

) ;

CREATE UNIQUE INDEX MVIALES.T_EMPRESA_PK ON MVIALES.T_EMPRESA

(
    CEDULAJURIDICA ASC
)

TABLESPACE PROYECTO PCTFREE 10 STORAGE

(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
BUFFER_POOL DEFAULT

)

LOGGING ;

ALTER TABLE MVIALES.T_EMPRESA ADD CONSTRAINT T_EMPRESA_PK PRIMARY KEY ( CEDULAJURIDICA )
USING INDEX MVIALES.T_EMPRESA_PK ;

```



```

CREATE TABLE MVIALES.T_ESTADOSTIPOCAMBIOS
(
    CODIGO NUMBER (3) NOT NULL ,
    NOMBRE VARCHAR2 (1 BYTE) NOT NULL
)
PCTFREE 10 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE
(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
    BUFFER_POOL DEFAULT
) ;

CREATE UNIQUE INDEX MVIALES.T_ESTADOSTIPOCAMBIOS_PK ON MVIALES.T_ESTADOSTIPOCAMBIOS
(
    CODIGO ASC
)
TABLESPACE PROYECTO PCTFREE 10 STORAGE
(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
    BUFFER_POOL DEFAULT
)
LOGGING ;

ALTER TABLE MVIALES.T_ESTADOSTIPOCAMBIOS ADD CONSTRAINT T_ESTADOSTIPOCAMBIOS_PK PRIMARY
KEY ( CODIGO ) USING INDEX MVIALES.T_ESTADOSTIPOCAMBIOS_PK ;

CREATE TABLE MVIALES.T_FACTURACOMPRAS
(
    CODIGO          NUMBER (10) NOT NULL ,
    FECCOMPRA       DATE NOT NULL ,
    FECREGISTRO     DATE NOT NULL ,
    CODUSERREGISTRO VARCHAR2 (10 BYTE) NOT NULL
)
PCTFREE 10 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE
(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
    BUFFER_POOL DEFAULT
) ;

CREATE UNIQUE INDEX MVIALES.T_FACTURACOMPRAS_PK ON MVIALES.T_FACTURACOMPRAS
(
    CODIGO ASC

```

```

)

TABLESPACE PROYECTO PCTFREE 10 STORAGE

(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
    BUFFER_POOL DEFAULT
)

LOGGING ;

ALTER TABLE MVIALES.T_FACTURACOMPRAS ADD CONSTRAINT T_FACTURACOMPRAS_PK PRIMARY KEY (
    CODIGO ) USING INDEX MVIALES.T_FACTURACOMPRAS_PK ;

CREATE TABLE MVIALES.T_FAMILIAS

(
    CODIGO                NUMBER (3) NOT NULL ,
    DESCRIPCION           VARCHAR2 (40 BYTE) NOT NULL ,
    PORCIMPACOMPRA        NUMBER (3) NOT NULL ,
    CODINDICEECONOMICO    NUMBER (3) NOT NULL
)

PCTFREE 10 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE

(
    INITIAL 1048576 NEXT 1048576 PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS 2147483645 FREELISTS
    1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT
) ;

CREATE UNIQUE INDEX MVIALES.T_FAMILIAS_PK ON MVIALES.T_FAMILIAS

(
    CODIGO ASC
)

TABLESPACE PROYECTO PCTFREE 10 STORAGE

(
    INITIAL 1048576 NEXT 1048576 PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS 2147483645 FREELISTS
    1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT
)

LOGGING ;

ALTER TABLE MVIALES.T_FAMILIAS ADD CONSTRAINT T_FAMILIAS_PK PRIMARY KEY ( CODIGO ) USING
INDEX MVIALES.T_FAMILIAS_PK ;

CREATE TABLE MVIALES.T_INDICEECONOMICOS

(
    CODIGO                NUMBER (3) NOT NULL ,
    DESCRIPCION           VARCHAR2 (40 BYTE) NOT NULL ,

```

```

    NUMVALORESREALES NUMBER (10) NOT NULL ,
    NUMMESES          NUMBER (10) NOT NULL
)
PCTFREE 10 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE
(
    INITIAL 1048576 NEXT 1048576 PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS 2147483645 FREELISTS
1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT
) ;
ALTER TABLE MVIALES.T_INDICEECONOMICOS ADD CONSTRAINT T_INDICEECONOMICOS_PK PRIMARY KEY (
CODIGO ) ;

CREATE TABLE MVIALES.T_LINEASFACTURACOMPRAS
(
    CODFACTURA NUMBER (10) NOT NULL ,
    CODARTICULO  VARCHAR2 (20 BYTE) NOT NULL ,
    PRECIODOLARES NUMBER (17,2) NOT NULL ,
    CANTIDAD     NUMBER (10) NOT NULL
)
PCTFREE 10 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE
(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
BUFFER_POOL DEFAULT
) ;
CREATE UNIQUE INDEX MVIALES.T_LINEASFACTURACOMPRAS_PK ON MVIALES.T_LINEASFACTURACOMPRAS
(
    CODFACTURA ASC , CODARTICULO ASC
)
TABLESPACE PROYECTO PCTFREE 10 STORAGE
(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
BUFFER_POOL DEFAULT
)
LOGGING ;
ALTER TABLE MVIALES.T_LINEASFACTURACOMPRAS ADD CONSTRAINT T_LINEASFACTURACOMPRAS_PK
PRIMARY KEY ( CODFACTURA, CODARTICULO ) USING INDEX MVIALES.T_LINEASFACTURACOMPRAS_PK ;

CREATE TABLE MVIALES.T_MARCAS
(
    CODIGO      NUMBER (3) NOT NULL ,

```

```

        DESCRIPCION VARCHAR2 (20 BYTE) NOT NULL
    )

    PCTFREE 10 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE

    (
        INITIAL 1048576 NEXT 1048576 PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS 2147483645 FREELISTS
        1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT
    ) ;

CREATE UNIQUE INDEX MVIALES.T_MARCAS_PK ON MVIALES.T_MARCAS

    (
        CODIGO ASC
    )

    TABLESPACE PROYECTO PCTFREE 10 STORAGE

    (
        INITIAL 1048576 NEXT 1048576 PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS 2147483645 FREELISTS
        1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT
    )

    LOGGING ;

ALTER TABLE MVIALES.T_MARCAS ADD CONSTRAINT T_MARCAS_PK PRIMARY KEY ( CODIGO ) USING
INDEX MVIALES.T_MARCAS_PK ;

CREATE TABLE MVIALES.T_MESANIO

    (
        MES  NUMBER (2) NOT NULL ,
        ANIO NUMBER (4) NOT NULL
    )

    PCTFREE 10 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE

    (
        PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
        BUFFER_POOL DEFAULT
    ) ;

CREATE UNIQUE INDEX MVIALES.T_MESANIO_PK ON MVIALES.T_MESANIO

    (
        MES ASC , ANIO ASC
    )

    TABLESPACE PROYECTO PCTFREE 10 STORAGE

    (
        PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
        BUFFER_POOL DEFAULT
    )

```

```

LOGGING ;

ALTER TABLE MVIALES.T_MESANIO ADD CONSTRAINT T_MESANIO_PK PRIMARY KEY ( MES, ANIO ) USING
INDEX MVIALES.T_MESANIO_PK ;

CREATE TABLE MVIALES.T_MONEDAS
(
    CODIGO NUMBER (3) NOT NULL ,
    NOMBRE VARCHAR2 (10 BYTE) NOT NULL
)
PCTFREE 10 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE
(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
    BUFFER_POOL DEFAULT
) ;

CREATE UNIQUE INDEX MVIALES.T_MONEDAS_PK ON MVIALES.T_MONEDAS
(
    CODIGO ASC
)
TABLESPACE PROYECTO PCTFREE 10 STORAGE
(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
    BUFFER_POOL DEFAULT
)
LOGGING ;

ALTER TABLE MVIALES.T_MONEDAS ADD CONSTRAINT T_MONEDAS_PK PRIMARY KEY ( CODIGO ) USING
INDEX MVIALES.T_MONEDAS_PK ;

CREATE TABLE MVIALES.T_PAISES
(
    CODPAIS NUMBER (3) NOT NULL ,
    NOMBRE NUMBER (10) NOT NULL
)
PCTFREE 10 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE
(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
    BUFFER_POOL DEFAULT
) ;

CREATE UNIQUE INDEX MVIALES.T_PAISES_PK ON MVIALES.T_PAISES
(

```

```

        CODPAIS ASC

    )

    TABLESPACE PROYECTO PCTFREE 10 STORAGE

    (

        PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
        BUFFER_POOL DEFAULT

    )

    LOGGING ;

    ALTER TABLE MVIALES.T_PAISES ADD CONSTRAINT T_PAISES_PK PRIMARY KEY ( CODPAIS ) USING
INDEX MVIALES.T_PAISES_PK ;

CREATE TABLE MVIALES.T_PERMISOUSUARIOS

    (

        CODIGO          NUMBER (3) NOT NULL ,

        DESCRIPCION VARCHAR2 (20 BYTE) NOT NULL

    )

    PCTFREE 10 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE

    (

        INITIAL 1048576 NEXT 1048576 PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS 2147483645 FREELISTS
        1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT

    ) ;

CREATE UNIQUE INDEX MVIALES.T_PERMISOUSUARIOS_PK ON MVIALES.T_PERMISOUSUARIOS

    (

        CODIGO ASC

    )

    TABLESPACE PROYECTO PCTFREE 10 STORAGE

    (

        INITIAL 1048576 NEXT 1048576 PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS 2147483645 FREELISTS
        1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT

    )

    LOGGING ;

    ALTER TABLE MVIALES.T_PERMISOUSUARIOS ADD CONSTRAINT T_PERMISOUSUARIOS_PK PRIMARY KEY (
    CODIGO ) USING INDEX MVIALES.T_PERMISOUSUARIOS_PK ;

CREATE TABLE MVIALES.T_PROVINCIAS

    (

        CODPROVINCIA NUMBER (3) NOT NULL ,

        CODPAIS      NUMBER (3) NOT NULL ,

        NOMBRE       VARCHAR2 (20 BYTE) NOT NULL
    )

```

```

)

PCTFREE 10 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE

(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
    BUFFER_POOL DEFAULT
) ;

CREATE UNIQUE INDEX MVIALES.T_PROVINCIAS_PK ON MVIALES.T_PROVINCIAS

(
    CODPROVINCIA ASC
)

TABLESPACE PROYECTO PCTFREE 10 STORAGE

(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
    BUFFER_POOL DEFAULT
)

LOGGING ;

ALTER TABLE MVIALES.T_PROVINCIAS ADD CONSTRAINT T_PROVINCIAS_PK PRIMARY KEY (
    CODPROVINCIA ) USING INDEX MVIALES.T_PROVINCIAS_PK ;

CREATE TABLE MVIALES.T_TIPOCONTACTOS

(
    CODTIPOCONTACTO NUMBER (3) NOT NULL ,
    NOMBRE NUMBER (10) NOT NULL
)

PCTFREE 10 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE

(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
    BUFFER_POOL DEFAULT
) ;

CREATE UNIQUE INDEX MVIALES.T_TIPOCONTACTOS_PK ON MVIALES.T_TIPOCONTACTOS

(
    CODTIPOCONTACTO ASC
)

TABLESPACE PROYECTO PCTFREE 10 STORAGE

(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
    BUFFER_POOL DEFAULT
)

LOGGING ;

ALTER TABLE MVIALES.T_TIPOCONTACTOS ADD CONSTRAINT T_TIPOCONTACTOS_PK PRIMARY KEY (

```

```

CODTIPOCONTACTO ) USING INDEX MVIALES.T_TIPOCONTACTOS_PK ;

CREATE TABLE MVIALES.T_TIPOEMPRESA

(
    CODTIPOEMPRESA NUMBER (3) NOT NULL ,
    NOMBRE          VARCHAR2 (20 BYTE) NOT NULL
)

PCTFREE 10 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE

(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
    BUFFER_POOL DEFAULT
) ;

CREATE UNIQUE INDEX MVIALES.T_TIPOEMPRESA_PK ON MVIALES.T_TIPOEMPRESA

(
    CODTIPOEMPRESA ASC
)

TABLESPACE PROYECTO PCTFREE 10 STORAGE

(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
    BUFFER_POOL DEFAULT
)

LOGGING ;

ALTER TABLE MVIALES.T_TIPOEMPRESA ADD CONSTRAINT T_TIPOEMPRESA_PK PRIMARY KEY (
    CODTIPOEMPRESA ) USING INDEX MVIALES.T_TIPOEMPRESA_PK ;


CREATE TABLE MVIALES.T_TIPOERRORES

(
    CODIGO          NUMBER (3) NOT NULL ,
    DESCRIPCION VARCHAR2 (15 BYTE) NOT NULL
)

PCTFREE 10 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE

(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
    BUFFER_POOL DEFAULT
) ;

CREATE UNIQUE INDEX MVIALES.T_TIPOERRORES_PK ON MVIALES.T_TIPOERRORES

(
    CODIGO ASC
)

```



```

TABLESPACE PROYECTO PCTFREE 10 STORAGE

(
    PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS 1
    BUFFER_POOL DEFAULT
)

LOGGING ;

ALTER TABLE MVIALES.T_TIPOERRORES ADD CONSTRAINT T_TIPOERRORES_PK PRIMARY KEY ( CODIGO )
USING INDEX MVIALES.T_TIPOERRORES_PK ;

CREATE TABLE MVIALES.T_TIPOUSUARIOS

(
    CODIGO                NUMBER (3) NOT NULL ,
    CODUSERPERMISO NUMBER (3) NOT NULL ,
    DESCRIPCION           VARCHAR2 (20 BYTE) NOT NULL
)

PCTFREE 10 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE

(
    INITIAL 1048576 NEXT 1048576 PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS 2147483645 FREELISTS
    1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT
) ;

CREATE UNIQUE INDEX MVIALES.T_TIPOUSUARIOS_PK ON MVIALES.T_TIPOUSUARIOS

(
    CODIGO ASC
)

TABLESPACE PROYECTO PCTFREE 10 STORAGE

(
    INITIAL 1048576 NEXT 1048576 PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS 2147483645 FREELISTS
    1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT
)

LOGGING ;

ALTER TABLE MVIALES.T_TIPOUSUARIOS ADD CONSTRAINT T_TIPOUSUARIOS_PK PRIMARY KEY ( CODIGO
) USING INDEX MVIALES.T_TIPOUSUARIOS_PK ;

CREATE TABLE MVIALES.T_UNIDADMEDIDAS

(
    CODIGO                NUMBER (3) NOT NULL ,
    DESCRIPCION VARCHAR2 (40 BYTE) NOT NULL
)

PCTFREE 10 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE

```

```

(
    INITIAL 1048576 NEXT 1048576 PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS 2147483645 FREELISTS
    1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT
) ;

CREATE UNIQUE INDEX MVIALES.T_UNIDADMEDIDAS_PK ON MVIALES.T_UNIDADMEDIDAS

(
    CODIGO ASC
)

TABLESPACE PROYECTO PCTFREE 10 STORAGE

(
    INITIAL 1048576 NEXT 1048576 PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS 2147483645 FREELISTS
    1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT
)

LOGGING ;

ALTER TABLE MVIALES.T_UNIDADMEDIDAS ADD CONSTRAINT T_UNIDADMEDIDAS_PK PRIMARY KEY (
    CODIGO ) USING INDEX MVIALES.T_UNIDADMEDIDAS_PK ;

CREATE TABLE MVIALES.T_USUARIOS

(
    NBRLOGIN    VARCHAR2 (10 BYTE) NOT NULL ,
    CLVLOGIN    VARCHAR2 (10 BYTE) NOT NULL ,
    CODUSERTIPO NUMBER (3) NOT NULL
)

PCTFREE 10 PCTUSED 40 TABLESPACE PROYECTO LOGGING STORAGE

(
    INITIAL 1048576 NEXT 1048576 PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS 2147483645 FREELISTS
    1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT
) ;

CREATE UNIQUE INDEX MVIALES.T_USUARIOS_PK ON MVIALES.T_USUARIOS

(
    NBRLOGIN ASC
)

TABLESPACE PROYECTO PCTFREE 10 STORAGE

(
    INITIAL 1048576 NEXT 1048576 PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS 2147483645 FREELISTS
    1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT
)

LOGGING ;

ALTER TABLE MVIALES.T_USUARIOS ADD CONSTRAINT T_USUARIOS_PK PRIMARY KEY ( NBRLOGIN )
USING INDEX MVIALES.T_USUARIOS_PK ;

```

ALTER TABLE MVIALES.T_ARTICULOSCOTIZADOS **ADD CONSTRAINT** FKT_ARTICULO382695 **FOREIGN KEY** (CODARTICULO) **REFERENCES** MVIALES.T_ARTICULOS (CODIGO) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_ARTICULOSCOTIZADOS **ADD CONSTRAINT** FKT_ARTICULO50573 **FOREIGN KEY** (NUMCOTIZACION) **REFERENCES** MVIALES.T_COTIZACIONES (NUMERO) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_ARTICULOSCOTIZADOS **ADD CONSTRAINT** FKT_ARTICULO522199 **FOREIGN KEY** (UNDMEDIDA) **REFERENCES** MVIALES.T_UNIDADMEDIDAS (CODIGO) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_ARTICULOS **ADD CONSTRAINT** FKT_ARTICULO570863 **FOREIGN KEY** (UNDMEDIDA) **REFERENCES** MVIALES.T_UNIDADMEDIDAS (CODIGO) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_ARTICULOS **ADD CONSTRAINT** FKT_ARTICULO578076 **FOREIGN KEY** (CODUSERREGISTRO) **REFERENCES** MVIALES.T_USUARIOS (NBRLOGIN) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_ARTICULOS **ADD CONSTRAINT** FKT_ARTICULO840016 **FOREIGN KEY** (CODESTARTICULO) **REFERENCES** MVIALES.T_ARTICULOESTADOS (CODIGO) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_ARTICULOS **ADD CONSTRAINT** FKT_ARTICULO848891 **FOREIGN KEY** (CODMARCA) **REFERENCES** MVIALES.T_MARCAS (CODIGO) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_ARTICULOS **ADD CONSTRAINT** FKT_ARTICULO904486 **FOREIGN KEY** (CODFAMILIA) **REFERENCES** MVIALES.T_FAMILIAS (CODIGO) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_BITACORA **ADD CONSTRAINT** FKT_BITACORA251134 **FOREIGN KEY** (CODTIPOERROR) **REFERENCES** MVIALES.T_TIPOERRORES (CODIGO) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_BITACORA **ADD CONSTRAINT** FKT_BITACORA84906 **FOREIGN KEY** (CODUSER) **REFERENCES** MVIALES.T_USUARIOS (NBRLOGIN) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_COMPONENTES **ADD CONSTRAINT** FKT_COMPONENTEN233783 **FOREIGN KEY** (CODCOMPONENTE) **REFERENCES** MVIALES.T_ARTICULOS (CODIGO) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_COMPONENTES **ADD CONSTRAINT** FKT_COMPONENTEN774059 **FOREIGN KEY** (CODARTICULO) **REFERENCES** MVIALES.T_ARTICULOS (CODIGO) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_CONTACTO **ADD CONSTRAINT** FKT_CONTACTO536274 **FOREIGN KEY** (CODEMPRESA) **REFERENCES** MVIALES.T_EMPRESA (CEDULAJURIDICA) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_CONTACTO **ADD CONSTRAINT** FKT_CONTACTO759656 **FOREIGN KEY** (CODTIPOCONTACTO) **REFERENCES** MVIALES.T_TIPOCONTACTOS (CODTIPOCONTACTO) NOT DEFERRABLE ;

```

ALTER TABLE MVIALES.T_COTIZACIONES ADD CONSTRAINT FKT_COTIZACI344790 FOREIGN KEY (
MESENTREG, ANIOENTREG ) REFERENCES MVIALES.T_MESANIO ( MES, ANIO ) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_COTIZACIONES ADD CONSTRAINT FKT_COTIZACI429323 FOREIGN KEY (
CEDJURCLIENTE ) REFERENCES MVIALES.T_EMPRESA ( CEDULAJURIDICA ) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_COTIZACIONES ADD CONSTRAINT FKT_COTIZACI656953 FOREIGN KEY (
CODMONEDA ) REFERENCES MVIALES.T_MONEDAS ( CODIGO ) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_COTIZACIONES ADD CONSTRAINT FKT_COTIZACI717841 FOREIGN KEY (
CODESTADOCOTIZ ) REFERENCES MVIALES.T_COTIZACIONESTADOS ( CODIGO ) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_COTIZACIONES ADD CONSTRAINT FKT_COTIZACI920077 FOREIGN KEY (
CODLISTAPRECIOS ) REFERENCES MVIALES.T_LISTAPRECIOS ( CODIGO ) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_CUIDADES ADD CONSTRAINT FKT_CUIDADES473810 FOREIGN KEY ( CODPROVINCIA
) REFERENCES MVIALES.T_PROVINCIAS ( CODPROVINCIA ) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_DETALLEDDIRECCION ADD CONSTRAINT FKT_DETALLEDD350223 FOREIGN KEY (
CODCIUDAD ) REFERENCES MVIALES.T_CUIDADES ( CODCIUDAD ) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_EMPRESA ADD CONSTRAINT FKT_EMPRESA228590 FOREIGN KEY ( CODTIPOEMPRESA
) REFERENCES MVIALES.T_TIPOEMPRESA ( CODTIPOEMPRESA ) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_EMPRESA ADD CONSTRAINT FKT_EMPRESA469254 FOREIGN KEY ( DIRECCION )
REFERENCES MVIALES.T_DETALLEDDIRECCION ( CODDETALLE ) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_FACTURACOMPRAS ADD CONSTRAINT FKT_FACTURAC28973 FOREIGN KEY (
CODUSERREGISTRO ) REFERENCES MVIALES.T_USUARIOS ( NBRLOGIN ) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_LINEASFACTURACOMPRAS ADD CONSTRAINT FKT_LINEASFA481568 FOREIGN KEY (
CODARTICULO ) REFERENCES MVIALES.T_ARTICULOS ( CODIGO ) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_LINEASFACTURACOMPRAS ADD CONSTRAINT FKT_LINEASFA50873 FOREIGN KEY (
CODFACTURA ) REFERENCES MVIALES.T_FACTURACOMPRAS ( CODIGO ) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_LISTAPRECIOXARTICULOS ADD CONSTRAINT FKT_LISTAPRE160880 FOREIGN KEY (
CODARTICULO ) REFERENCES MVIALES.T_ARTICULOS ( CODIGO ) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_LISTAPRECIOXARTICULOS ADD CONSTRAINT FKT_LISTAPRE78013 FOREIGN KEY (
CODLISTAPRECIO ) REFERENCES MVIALES.T_LISTAPRECIOS ( CODIGO ) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_PROVINCIAS ADD CONSTRAINT FKT_PROVINCIA451632 FOREIGN KEY ( CODPAIS )

```

```

REFERENCES MVIALES.T_PAISES ( CODPAIS ) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_PROYECCIONESLISTAPRECIOS ADD CONSTRAINT FKT_PROYECCI566293 FOREIGN
KEY ( CODLISTAPRECIOS, CODARTICULO ) REFERENCES MVIALES.T_LISTAPRECIOXARTICULOS (
CODLISTAPRECIO, CODARTICULO ) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_PROYECCIONESLISTAPRECIOS ADD CONSTRAINT FKT_PROYECCI761595 FOREIGN
KEY ( MES, ANIO ) REFERENCES MVIALES.T_MESANIO ( MES, ANIO ) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_TIPOCAMBIOS ADD CONSTRAINT FKT_TIPOCAMB620100 FOREIGN KEY (
CODMONEDA1 ) REFERENCES MVIALES.T_MONEDAS ( CODIGO ) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_TIPOCAMBIOS ADD CONSTRAINT FKT_TIPOCAMB620101 FOREIGN KEY (
CODMONEDA2 ) REFERENCES MVIALES.T_MONEDAS ( CODIGO ) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_TIPOCAMBIOS ADD CONSTRAINT FKT_TIPOCAMB651026 FOREIGN KEY ( ESTADO )
REFERENCES MVIALES.T_ESTADOSTIPOCAMBIOS ( CODIGO ) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_TIPOCAMBIOS ADD CONSTRAINT FKT_TIPOCAMB923514 FOREIGN KEY ( MES, ANIO
) REFERENCES MVIALES.T_MESANIO ( MES, ANIO ) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_TIPOUSUARIOS ADD CONSTRAINT FKT_TIPOUSUA556339 FOREIGN KEY (
CODUSERPERMISO ) REFERENCES MVIALES.T_PERMISOUSUARIOS ( CODIGO ) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_USUARIOS ADD CONSTRAINT FKT_USUARIOS41238 FOREIGN KEY ( CODUSERTIPO )
REFERENCES MVIALES.T_TIPOUSUARIOS ( CODIGO ) NOT DEFERRABLE ;

ALTER TABLE MVIALES.T_ARTICULOSCOTIZADOSHISTO ADD CONSTRAINT PK_NUMCOTIZACIONHISTO FOREIGN
KEY ( NUMCOTIZACION ) REFERENCES MVIALES.T_COTIZACIONESHISTO ( NUMERO ) NOT DEFERRABLE ;

-- ERROR: FK name length exceeds maximum allowed length(30)

ALTER TABLE MVIALES.T_FAMILIAS ADD CONSTRAINT T_FAMILIAS_T_INDICEECONOMICOS_FK FOREIGN KEY
( CODINDICEECONOMICO ) REFERENCES MVIALES.T_INDICEECONOMICOS ( CODIGO ) NOT DEFERRABLE ;

-- ERROR: FK name length exceeds maximum allowed length(30)

ALTER TABLE MVIALES.T_TIPOCAMBIOS ADD CONSTRAINT T_TIPOCAMBIOS_T_INDICEECONOMICOS_FK
FOREIGN KEY ( CODINDICEECONOMICO ) REFERENCES MVIALES.T_INDICEECONOMICOS ( CODIGO ) NOT
DEFERRABLE ;

```

Script de llenado

```
-----  
--insertar indices economicos  
-----
```

```
INSERT INTO t_indiceEconomicos(DESCRIPCION, NUMVALORESREALES, NUMMESES)  
VALUES('índice económico de ejemplo', 10, 2);
```

```
--para verificar
```

```
SELECT * FROM t_indiceEconomicos;  
commit;
```

```
-----  
--insertar familias  
-----
```

```
--CODINFICEECONOMICO por el momento es 1 para que sea el de ejemplo
```

```
--PORCIMPACOMPRA por el momento es 13 para que sea el de ejemplo
```

```
INSERT INTO t_familias(DESCRIPCION, PORCIMPACOMPRA, CODINDICEECONOMICO)  
VALUES('procesadores',13,1);
```

```
INSERT INTO t_familias(DESCRIPCION, PORCIMPACOMPRA, CODINDICEECONOMICO)  
VALUES('tarjetas de video',13,1);
```

```
INSERT INTO t_familias(DESCRIPCION, PORCIMPACOMPRA, CODINDICEECONOMICO)  
VALUES('discos duros',13,1);
```

```
INSERT INTO t_familias(DESCRIPCION, PORCIMPACOMPRA, CODINDICEECONOMICO)  
VALUES('memorias',13,1);
```

```
INSERT INTO t_familias(DESCRIPCION, PORCIMPACOMPRA, CODINDICEECONOMICO)  
VALUES('impresoras',13,1);
```

```
INSERT INTO t_familias(DESCRIPCION, PORCIMPACOMPRA, CODINDICEECONOMICO)  
VALUES('monitores',13,1);
```

```
INSERT INTO t_familias(DESCRIPCION, PORCIMPACOMPRA, CODINDICEECONOMICO)  
VALUES('escáneres',13,1);
```

```
INSERT INTO t_familias(DESCRIPCION, PORCIMPACOMPRA, CODINDICEECONOMICO)  
VALUES('unidades de CD-ROM',13,1);
```

```
INSERT INTO t_familias(DESCRIPCION, PORCIMPACOMPRA, CODINDICEECONOMICO)
```

```

VALUES ('CPU',13,1);

INSERT INTO t_familias (DESCRIPCION, PORCIMPACOMPRA, CODINDICEECONOMICO)
VALUES ('microcomputadores',13,1);

INSERT INTO t_familias (DESCRIPCION, PORCIMPACOMPRA, CODINDICEECONOMICO)
VALUES ('PLC',13,1);

INSERT INTO t_familias (DESCRIPCION, PORCIMPACOMPRA, CODINDICEECONOMICO)
VALUES ('soluciones integradas',13,1);

INSERT INTO t_familias (DESCRIPCION, PORCIMPACOMPRA, CODINDICEECONOMICO)
VALUES ('sistema operativo',13,1);

INSERT INTO t_familias (DESCRIPCION, PORCIMPACOMPRA, CODINDICEECONOMICO)
VALUES ('software de oficina',13,1);

INSERT INTO t_familias (DESCRIPCION, PORCIMPACOMPRA, CODINDICEECONOMICO)
VALUES ('ambiente de desarrollo',13,1);

INSERT INTO t_familias (DESCRIPCION, PORCIMPACOMPRA, CODINDICEECONOMICO)
VALUES ('software de base de datos',13,1);

--para verificar
SELECT * FROM t_familias;

commit;

-----

--insertar monedas
-----

INSERT INTO t_monedas (NOMBRE) VALUES ('dolares');
INSERT INTO t_monedas (NOMBRE) VALUES ('colones');

--para verificar
SELECT * FROM t_monedas;

commit;

-----

--insertar unidades de medidas
-----

INSERT INTO t_unidadMedidas (DESCRIPCION) VALUES ('unidad estandar');

```

```

--para verificar
SELECT * FROM t_unidadMedidas;
commit;

-----

--insertar marcas
-----

INSERT INTO t_marcas(DESCRIPCION) VALUES('Samsung');
INSERT INTO t_marcas(DESCRIPCION) VALUES('Toshiba');
INSERT INTO t_marcas(DESCRIPCION) VALUES('Dell');
INSERT INTO t_marcas(DESCRIPCION) VALUES('Apple');
INSERT INTO t_marcas(DESCRIPCION) VALUES('Hitachi');

--para verificar
SELECT * FROM t_marcas;
commit;

-----

--insertar estados de los articulos
-----

INSERT INTO t_articuloEstados(DESCRIPCION) VALUES('Activo');
INSERT INTO t_articuloEstados(DESCRIPCION) VALUES('Inactivo');

--para verificar
SELECT * FROM t_articuloEstados;
commit;

-----

--insertar permisos de usuarios
-----

```



```
INSERT INTO t_permisoUsuarios (DESCRIPCION) VALUES ('Todos');
```

```
--para verificar
```

```
SELECT * FROM t_permisoUsuarios;
```

```
commit;
```

```
-----  
--insertar tipos de usuarios  
-----
```

```
INSERT INTO t_tipoUsuarios (CODUSERPERMISO, DESCRIPCION)  
VALUES (1, 'Administrador');
```

```
--para verificar
```

```
SELECT * FROM t_tipoUsuarios;
```

```
commit;
```

```
-----  
--insertar usuarios  
-----
```

```
INSERT INTO t_usuarios (NBRLOGIN, CLVLOGIN, CODUSERTIPO)  
VALUES ('admin', 'admin', 1);
```

```
--para verificar
```

```
SELECT * FROM t_usuarios;
```

```
commit;
```

```
-----  
--insertar articulos  
-----
```

```
INSERT INTO t_articulos (CODIGO, DESCRIPCION, UNDMEDIDA, CODMARCA, MODELO,  
FECREGISTRO, CODFAMILIA, CODESTARTICULO, CODUSERREGISTRO, CANTIDAD,  
CANTMAXIMA, CANTMINIMA, PRECIOMERCDOLARES, FECHAACTPRECIO)
```

```
VALUES ('FTCO2175', 'Monitor pantalla plana de 21 pulgadas', 1, 1,
```

```

'Pantalla Plana', sysdate, 6, 1, 'admin', 10, 25, 20, 50, sysdate);

INSERT INTO t_articulos(CODIGO, DESCRIPCION, UNDMEDIDA, CODMARCA, MODELO,
FECREGISTRO, CODFAMILIA, CODESTARTICULO, CODUSERREGISTRO, CANTIDAD,
CANTMAXIMA, CANTMINIMA, PRECIOMERCDOLARES, FECHAACTPRECIO)

VALUES('SL02.8G08ME500SSD','carcaza slim 8 GB RAM, 500 GB disco duro', 1,
2, 'CPU Slim', sysdate, 9, 1, 'admin', 5, 15, 10, 100, sysdate);

INSERT INTO t_articulos(CODIGO, DESCRIPCION, UNDMEDIDA, CODMARCA, MODELO,
FECREGISTRO, CODFAMILIA, CODESTARTICULO, CODUSERREGISTRO, CANTIDAD,
CANTMAXIMA, CANTMINIMA, PRECIOMERCDOLARES, FECHAACTPRECIO)

VALUES('INCOHO','Impresora color inyeccion de tinta', 1, 1, 'Inyeccion',
sysdate, 5, 1, 'admin', 3, 15, 10, 40, sysdate);

INSERT INTO t_articulos(CODIGO, DESCRIPCION, UNDMEDIDA, CODMARCA, MODELO,
FECREGISTRO, CODFAMILIA, CODESTARTICULO, CODUSERREGISTRO, CANTIDAD,
CANTMAXIMA, CANTMINIMA, PRECIOMERCDOLARES, FECHAACTPRECIO)

VALUES('FP110','Fuente de poder 110V', 1, 5, '110V', sysdate, 9, 1,
'admin', 2, 15, 10, 25, sysdate);

--para verificar
SELECT * FROM t_articulos;

commit;

-----

--insertar componentes

-----

INSERT INTO t_componentes(CODCOMPONENTE, CODARTICULO)
VALUES('FP110','SL02.8G08ME500SSD');

--para verificar
SELECT * FROM t_componentes;

commit;

-----

--insertar listas de precios

-----

```

```
INSERT INTO t_listaPrecios(DESCRIPCION, GASTOADMIN, GASTOOTROS, UTILIDAD)
VALUES('Sector publico', 150, 100, 50);

INSERT INTO t_listaPrecios(DESCRIPCION, GASTOADMIN, GASTOOTROS, UTILIDAD)
VALUES('Sector privador', 120, 99, 60);

INSERT INTO t_listaPrecios(DESCRIPCION, GASTOADMIN, GASTOOTROS, UTILIDAD)
VALUES('Cliente contado', 100, 80, 30);

INSERT INTO t_listaPrecios(DESCRIPCION, GASTOADMIN, GASTOOTROS, UTILIDAD)
VALUES('Cliente frecuente-contado', 90, 80, 30);

INSERT INTO t_listaPrecios(DESCRIPCION, GASTOADMIN, GASTOOTROS, UTILIDAD)
VALUES('Cliente credito', 200, 150, 50);

INSERT INTO t_listaPrecios(DESCRIPCION, GASTOADMIN, GASTOOTROS, UTILIDAD)
VALUES('Importados', 150, 80, 35);

--para verificar
SELECT * FROM t_listaPrecios;

commit;
```

Paquetes con procedimientos almacenados (Stored procedures)

```
CREATE OR REPLACE PACKAGE MVIALES.PKG_ARTICULO AS

    PROCEDURE SP_InsertarArticulo (IN_CODIGO IN VARCHAR2, IN_DESCRIPCION
IN VARCHAR2, IN_UNDMEDIDA IN NUMBER, IN_CODMARCA IN NUMBER,
                                IN_MODELO IN VARCHAR2,
IN_CODFAMILIA IN NUMBER, IN_CODESTARTICULO IN NUMBER,
                                IN_CODUSERREGISTRO IN VARCHAR2,
IN_CANTIDAD IN NUMBER, IN_CANTMAXIMA IN NUMBER,
                                IN_CANTMINIMA IN NUMBER,
IN_PRECIOMERCDOLARES IN NUMBER);

    PROCEDURE SP_InsertarComponente (IN_CODIGOCOMPONENTE IN
VARCHAR2,IN_CODIGOARTICULO IN VARCHAR2 );

    PROCEDURE SP_ModificarArticulo (IN_CODIGO IN VARCHAR2,
IN_CODUSERREGISTRO IN VARCHAR2, IN_PRECIOMERCDOLARES IN NUMBER);

    PROCEDURE SP_ModificarComponente (IN_CODCOMPONENTE IN
VARCHAR2,IN_CODARTICULO IN VARCHAR2);

    PROCEDURE SP_BorrarComponente (IN_CODCOMPONENTE IN
VARCHAR2,IN_CODARTICULO IN VARCHAR2);

    Procedure SP_GetArticulos;

    Procedure SP_GetComponentes;

END PKG_Articulo;

CREATE OR REPLACE PACKAGE BODY MVIALES.PKG_ARTICULO
AS

    --*****Insercion
```

```
Articulos*****  
*/
```

```
PROCEDURE SP_InsertarArticulo(  
    IN_CODIGO          IN VARCHAR2,  
    IN_DESCRIPCION      IN VARCHAR2,  
    IN_UNDMEDIDA        IN NUMBER,  
    IN_CODMARCA         IN NUMBER,  
    IN_MODELO           IN VARCHAR2,  
    IN_CODFAMILIA       IN NUMBER,  
    IN_CODESTARTICULO   IN NUMBER,  
    IN_CODUSERREGISTRO  IN VARCHAR2,  
    IN_CANTIDAD         IN NUMBER,  
    IN_CANTMAXIMA       IN NUMBER,  
    IN_CANTMINIMA       IN NUMBER,  
    IN_PRECIOMERCDOLARES IN NUMBER)
```

AS

BEGIN

INSERT

INTO t_articulos

```
(  
    CODIGO,  
    DESCRIPCION,  
    UNDMEDIDA,  
    CODMARCA,  
    MODELO,  
    FECREGISTRO,  
    CODFAMILIA,  
    CODESTARTICULO,  
    CODUSERREGISTRO,  
    CANTIDAD,  
    CANTMAXIMA,  
    CANTMINIMA,  
    PRECIOMERCDOLARES,
```

```

        FECHAACTPRECIO
    )
    VALUES
    (
        IN_CODIGO,
        IN_DESCRIPCION,
        IN_UNDMEDIDA,
        IN_CODMARCA,
        IN_MODELO,
        sysdate,
        IN_CODFAMILIA,
        IN_CODESTARTICULO,
        IN_CODUSERREGISTRO,
        IN_CANTIDAD,
        IN_CANTMAXIMA,
        IN_CANTMINIMA,
        IN_PRECIOMERCDOLARES,
        sysdate
    );
END;

--*****Insercion
Componentes*****
***//

--Requiere el codigo de componente y el codigo del articulo al cual se
asociara, e.g execute SP_INSERTARCOMPONENTE('FTCO2175','INCOHO');

PROCEDURE SP_InsertarComponente
(
    IN_CODIGOCOMPONENTE IN VARCHAR2,
    IN_CODIGOARTICULO   IN VARCHAR2
)
AS
BEGIN
    INSERT
    INTO t_componentes

```

```

(
    CODCOMPONENTE,
    CODARTICULO
)
VALUES
(
    IN_CODIGOCOMPONENTE,
    IN_CODIGOARTICULO
);
END;

--*****Modificacion
Articulos*****
*/

--Requiere el codigo de articulo por modificar, el usuario e.g 'admin' y
el precio nuevo en dolares, la fecha se cambia sola, e.g execute
SP_ModificarArticulo('MOCO4075','admin',200);

PROCEDURE SP_ModificarArticulo
(
    IN_CODIGO          IN VARCHAR2,
    IN_CODUSERREGISTRO IN VARCHAR2,
    IN_PRECIOMERCDOLARES IN NUMBER
)
AS
BEGIN
    UPDATE t_articulos
    SET CODUSERREGISTRO = IN_CODUSERREGISTRO,
        PRECIOMERCDOLARES = IN_PRECIOMERCDOLARES,
        FECHAACTPRECIO    = sysdate
    WHERE CODIGO          = IN_CODIGO;
END;

--*****Modificacion
Componentes*****
***/

--Requiere el codigo de componente por modificar y el codigo nuevo del
articulo al cual se asociara, e.g execute
SP_MODIFICARCOMPONENTE('FP110','SL02.8G08ME500SSD');

```

```

PROCEDURE SP_ModificarComponente(
    IN_CODCOMPONENTE IN VARCHAR2,
    IN_CODARTICULO    IN VARCHAR2)
AS
BEGIN
    UPDATE t_componentes
    SET CODARTICULO      = IN_CODARTICULO
    WHERE CODCOMPONENTE = IN_CODCOMPONENTE;
END;

--*****Borrado
Articulo*****
*****//

--*****Borrado
Componente*****
*****//

--Si se desea borrar solo una linea en especifico (un componente esta
asociado a un articulo X) se pasa de parametro el codigo del componente
por eliminar y el codigo del articulo asociado, e.g execute
SP_BORRARCOMPONENTE('FTCO2175','SL02.8G08ME500SSD');

--Si se desea borrar todas las entradas donde se encuentr el componente
se pasa de parametro el codigo del componente por eliminar y un parametro
null ('') como segundo parametro, e.g execute
SP_BorrarComponente('FTCO2175','');

PROCEDURE SP_BorrarComponente(
    IN_CODCOMPONENTE IN VARCHAR2,
    IN_CODARTICULO    IN VARCHAR2)
AS
BEGIN
    IF(IN_CODARTICULO IS NULL) THEN
        DELETE FROM t_componentes WHERE CODCOMPONENTE = IN_CODCOMPONENTE;
    ELSE
        DELETE
        FROM t_componentes
        WHERE CODCOMPONENTE = IN_CODCOMPONENTE
        AND CODARTICULO      = IN_CODARTICULO;
    END IF;
END;

```


PROCEDURE SP_GetArticulos

IS

```
Var_codArticulo      VARCHAR2(20);
Var_descripcion      VARCHAR2(40);
Var_codUnidMed       VARCHAR2(40);
Var_codMarca         NUMBER(3);
Var_modelo           VARCHAR2(15);
Var_fechaReg         DATE;
Var_codFamilia       NUMBER(3);
Var_codEstArticulo   NUMBER(3);
Var_codUserRegistro  VARCHAR2(10);
Var_Cant             NUMBER(10);
Var_CantMax          NUMBER(10);
Var_CantMin          NUMBER(10);
Var_PrecioMerDolar   NUMBER(17,2);
Var_FechaActPrecio   DATE;
```

CURSOR cursorArticulos

IS

```
SELECT codigo,
        descripcion,
        undMedida,
        codMarca,
        modelo,
        fecRegistro,
        codFamilia,
        codEstArticulo,
        codUserRegistro,
        cantidad,
        cantMaxima,
        cantMinima,
        precioMercDolares,
        fechaActPrecio
```

```

        FROM t_articulos;
BEGIN
    OPEN cursorArticulos;
    LOOP
        FETCH cursorArticulos
        INTO Var_codArticulo,
            Var_descripcion,
            Var_codUnidMed,
            Var_codMarca,
            Var_modelo,
            Var_fechaReg,
            Var_codFamilia,
            Var_codEstArticulo,
            Var_codUserRegistro,
            Var_Cant,
            Var_CantMax,
            Var_CantMin,
            Var_PrecioMerDolar,
            Var_FechaActPrecio;
        EXIT
    WHEN cursorArticulos %notfound;
        DBMS_OUTPUT.PUT_LINE(Var_descripcion); --solo para verificar
    END LOOP;
    CLOSE cursorArticulos;
    COMMIT;
END SP_GetArticulos;
PROCEDURE SP_GetComponentes
IS
    Var_codComponente VARCHAR2(20);
    Var_codArticulo   VARCHAR2(20);
    CURSOR cursorComponentes
    IS

```

```
        SELECT codComponente,codArticulo FROM t_componentes;
BEGIN
    OPEN cursorComponentes;
    LOOP
        FETCH cursorComponentes INTO Var_codComponente,Var_codArticulo;
        EXIT
    WHEN cursorComponentes %notfound;
        DBMS_OUTPUT.PUT_LINE(Var_codComponente); --solo para verificar
    END LOOP;
    CLOSE cursorComponentes;
    COMMIT;
END SP_GetComponentes;
END PKG_Articulo;
```

Procedimientos almacenados (Stored procedures)

```
CREATE OR REPLACE PROCEDURE MVIALES.SP_OBTENERTC (
    PDATE IN DATE,
    PRESULT OUT NUMBER
) IS
    MES NUMBER(2);
    ANIO NUMBER(4);
BEGIN
    -- OBTENER EL MES Y ANIO DE LA FECHA
    SELECT
        EXTRACT(MONTH FROM PDATE),EXTRACT(YEAR FROM PDATE) INTO MES,ANIO
    FROM DUAL;
    -- SELECCIONE EL TIPO DE CAMBIO DE LA TABLA DE TIPOS DE CAMBIO
    SELECT
        TC.MONTO INTO PRESULT
    FROM T_TIPOCAMBIOS TC
    WHERE TC.MES = MES AND TC.ANIO = ANIO;
END;

CREATE OR REPLACE procedure MVIALES.SP_PROYECCIONGENERAR (
    PCODLISTAPRECIO in number,
    PNUMMESES in number,
    PMSGRESPUESTA OUT VARCHAR2
) is
    -- Define variables utilizadas para los calculos
    PORC1 number(3);
    PORC2 number(3);
    PORC3 number(3);
    SUMPORC NUMBER(3);
```

```

SBTPORC NUMBER(17,2);
CODARTICULO VARCHAR2(20);
PRCARTICULO NUMBER(17,2);
MONTOTC NUMBER(17,2);
SBTPRECIOART NUMBER(17,2);
TPRECIOART NUMBER(17,2);
MES NUMBER(2);
ANIO NUMBER(3);
CONTA NUMBER(5);
NEWFECHA DATE;

-- OBTENER LOS PRECIOS DE LOS ARTICULOS ASOCIADOS A LA LISTA DE
PRECIOS

-- CURSOR DE PRECIO DE ARTICULOS

CURSOR CURPRECIOARTIC IS

SELECT

    ARTIC.CODIGO,

    ARTIC.PRECIOMERCDOLARES

FROM T_LISTAPRECIOXARTICULOS LPXA INNER JOIN T_ARTICULOS ARTIC ON
(LPXA.CODARTICULO = ARTIC.CODIGO)

WHERE LPXA.CODLISTAPRECIO = PCODLISTAPRECIO;

begin

-- OBTENER LOS PORCENTAJES DE LA LISTA DE PRECIO

SELECT

    lpr.gastoAdmin,lpr.gastoOtros,lpr.utilidad INTO PORC1, PORC2, PORC3

FROM T_ListaPrecios lpr

WHERE lpr.codigo = PCODLISTAPRECIO;

PMSGRESPUESTA := ' ' + 'Obtenido porcentajes\n';

-- CALCULAR TOTAL DE PORCENTAJES

SUMPORC := PORC1 + PORC2 + PORC3;

SBTPORC := SUMPORC / 100;

SBTPORC := SBTPORC + 1;

PMSGRESPUESTA := PMSGRESPUESTA + 'Calculado subtotal de
porcentajes\n';

```

```

-- PARA CADA ARTICULO EN EL CURSOR, CALCULAR SU PRECIO
OPEN CURPRECIOARTIC;

LOOP

    -- EXTRAER DATOS DEL ARTICULO

    FETCH CURPRECIOARTIC INTO CODARTICULO, PRCARTICULO;

    IF (CURPRECIOARTIC%FOUND) THEN

        PMSGRESPUESTA := PMSGRESPUESTA + 'Dato ok en cursor\n';

        -- INICIALIZA CALCULO DE PRECIO PARA LA CANTIDAD DE MESES
        ESPECIFICADA

        CONTA := 0;

        WHILE (CONTA < PNUMMESES)

            LOOP

                -- INCREMENTA UN MES

                SELECT

                    ADD_MONTHS(sysdate,CONTA) INTO NEWFECHA

                FROM DUAL;

                -- OBTENER EL MONTO DE TIPO DE CAMBIO PARA ESE MES

                SP_OBTENERTC(NEWFECHA, MONTOTC);

                -- CALCULAR SUBTOTAL DEL PRECIO DEL ARTICULO

                SBTPRECIOART := MONTOTC * PRCARTICULO;

                -- CALCULAR TOTAL DEL PRECIO DEL ARTICULO (APLICA PORCENTAJES)

                TPRECIOART := SBTPRECIOART * (1 + SBTPORC);

                -- EXTRAER EL MES Y EL AÑO DE LA FECHA

                SELECT

                    EXTRACT(MONTH FROM NEWFECHA), EXTRACT(YEAR FROM NEWFECHA) INTO
MES, ANIO

                FROM DUAL;

                -- INSERTA UNA FILA DE PROYECCIONES

                INSERT INTO T_PROYECCIONESLISTAPRECIOS VALUES (CODARTICULO,
PCODLISTAPRECIO, MES, ANIO, MONTOTC, TPRECIOART);

                -- SIGUE CON EL SIGUIENTE MES DEL MISMO ARTICULO

                CONTA := CONTA + 1;

            END LOOP;

```

```

ELSE

    PMSGRESPUESTA := PMSGRESPUESTA + 'No se encontro datos en
cursor\n';

    END IF;

    EXIT WHEN CURPRECIOARTIC %NOTFOUND;

    END LOOP;

End;

/

CREATE OR REPLACE PROCEDURE MVIALES.SP_PROYECCIONTC1MES(

    PNUMVALREALES IN NUMBER,

    PDATE IN DATE,

    -- PARA INSERT SOLAMENTE

    PCODMONEDA1 IN NUMBER,

    PCODMONEDA2 IN NUMBER,

    PCODIDECONO IN VARCHAR2,

    PNEXTTC OUT NUMBER

) IS

    MES NUMBER(2);

    ANIO NUMBER(4);

    CURTC NUMBER(17,2);    -- VALOR DEL TC DE LA FECHA

    TC0 NUMBER(17,2);      -- VARIABLE DE CONTROL DE MONTO DE TIPO DE CAMBIO
(MININUENDO)

    TC1 NUMBER(17,2);      -- VARIABLE DE CONTROL DE MONTO DE TIPO DE CAMBIO
(SUSTRAENDO)

    DIFF NUMBER(17,2);     -- VARIABLE DE CONTROL DE MONTO (DIFERENCIA ENTRE
MONTOS DE 2 MESES CONSECUTIVOS)

    NEWFECHA DATE;        -- NUEVA FECHA DE DONDE OBTENER EL MONTO DEL TIPO
DE CAMBIO

    CONTA NUMBER(10);      -- CONTROL DEL BUCLE

BEGIN

    -- INICIALIZA VARIABLES

    -- OBTENER EL TIPO DE CAMBIO DEL MES ACTUAL

```

```

SP_OBTENERTC (SYSDATE, CURTC);

TC0 := 0.0;

TC1 := 0.0;

CONTA := 0;

DIFF := 0.0;

-- BUCLE DE RECORRIDO EN LA TABLA TC SEGUN EL NUMERO DE VALORES REALES
WHILE (CONTA < PNUMVALREALES)

LOOP

    -- OBTIENE EL TIPO DE CAMBIO DEL MES ACTUAL

    SELECT

        ADD_MONTHS (sysdate, - CONTA) INTO NEWFECHA

    FROM DUAL;

    SP_OBTENERTC (NEWFECHA, TC0);

    -- OBTIENE EL TIPO DE CAMBIO DEL MES ANTERIOR

    SELECT

        ADD_MONTHS (sysdate, - (CONTA+1)) INTO NEWFECHA

    FROM DUAL;

    SP_OBTENERTC (NEWFECHA, TC1);

    -- CALCULA DIFERENCIAS

    DIFF := DIFF + (TC0 - TC1);

    CONTA := CONTA + 1;

END LOOP;

-- CALCULA EL PROMEDIO

DIFF := (DIFF / PNUMVALREALES);

-- CALCULA EL TIPO DE CAMBIO PARA EL SIGUIENTE MES

CURTC := (CURTC + DIFF);

PNEXTTC := CURTC;

-- INSERTAR EL MONTO COMO PROYECTADO EN LA TABLA DE MONTOS

SELECT

    ADD_MONTHS (sysdate, (CONTA+1)) INTO NEWFECHA

FROM DUAL;

```



```

SELECT

    EXTRACT(MONTH FROM NEWFECHA),EXTRACT(YEAR FROM NEWFECHA) INTO
MES,ANIO

FROM DUAL;

-- INSERTAR EN LA TABLA DE MESANIO :'( NOP, PUEDE DAR ERROR DE LLAVE
DUPLICADA

-- INSERT INTO T_MESANIO VALUES (MES,ANIO);

-- INSERTAR EN LA TABLA DE TIPOS DE CAMBIOS

INSERT INTO T_TIPOCAMBIOS VALUES (PCODMONEDA1, PCODMONEDA2, PNEXTTC,
PCODIDECONO, 2, MES, ANIO);

END;

/

CREATE OR REPLACE PROCEDURE MVIALES.SP_PROYECCIONTCNMESES (

    PNUMMESES IN NUMBER,

    PNUMVALREALES IN NUMBER,

    -- PARA INSERT SOLAMENTE

    PCODMONEDA1 IN NUMBER,

    PCODMONEDA2 IN NUMBER,

    PCODIDECONO IN VARCHAR2

) IS

    CONTA NUMBER(10);

    PMONTO NUMBER(17,2);

    NEWFECHA DATE;

BEGIN

    CONTA := 0;

    PMONTO := 0;

    WHILE (CONTA < PNUMMESES)

    LOOP

        -- INCREMENTA UN MES

        SELECT

            ADD_MONTHS(sysdate,CONTA) INTO NEWFECHA

```

```
FROM DUAL;

-- CALCULA EL TC PARA ESE MES

SP_PROYECCIONTC1MES(PNUMVALREALES, NEWFECHA, PCODMONEDA1,
PCODMONEDA2, PCODIDECONO, PMONTO);

CONTA := CONTA + 1;

END LOOP;

END;
```