

Al finalizar la práctica el alumno deberá entregar un script con todas las instrucciones. **Por ello se recomienda encarecidamente** que todas las instrucciones, scripts generados y comentarios explicando que se ha hecho (e.g. al importar los datos) se vayan guardando siempre en un fichero (script).

## 1. Creación del usuario y Tablespace

Las acciones de este apartado se pueden acometer desde Data Modeler (antes de generar el DDL necesario que haremos en un paso posterior), o directamente desde SQL Developer.

Crea un usuario denominado *secretaria* y asígnale por defecto un tablespace denominado TS\_SECRETARIA. Asígnale cuota suficiente en ese tablespace. Dale permisos para crear tablas, vistas y vistas materializadas. También para crear secuencias y procedimientos. Crear un Tablespace denominado TS\_INDICES con 50 M. Asignar cuota al usuario para ese Tablespace.

Comprobar consultando el diccionario de datos que existen los tablespace TS\_SECRETARIA y TS\_INDICES. Comprobar consultando el diccionario de datos que el tablespace por defecto del usuario SECRETARIA es TS\_SECRETARIA. Comprobar consultando el diccionario de datos los datafiles que tienen asociado TS\_SECRETARIA y TS\_INDICES.

## 2. Creación del Esquema

En este paso generaremos el script desde Data Modeler. Aquí, **se recomienda encarecidamente** repasar el modelo de acuerdo a los datos que posteriormente se importaran desde los distintos archivos excel y csv (**archivos que se encuentran en la pestaña del trabajo en grupo**). Es muy importante dedicarle un tiempo dilatado a esta opción de forma una vez tengamos el modelo relacional en la BD, se realicen el menos número de cambios posibles (que siempre serán más costosos debido a la existencia de dependencias y restricciones controladas por la BD).

Una vez generado, modifica el script para que todos los índices generados residan en el tablespace TS\_INDICES. Ten en cuenta que Oracle crea de forma automática algunos índices. Aunque son más, para esta práctica es suficiente con saber que genera índices de forma automática para las claves primarias, los campos únicos y los atributos equivalentes en las vistas materializadas.

Así, si el script está usando la sentencia CREATE INDEX, modifícala para añadir TS\_INDICES como tablespace. Si por el contrario el script está creando la constraint directamente en la sentencia CREATE o ALTER TABLE, entonces puede añadir la cláusula tras la definición la constraint. Véase el siguiente ejemplo:

```
CREATE TABLE EJEMPLO
( CODIGO VARCHAR2(50) ,
  TEXTO VARCHAR2(100) ,
  CONSTRAINT EJEMPLO_PK PRIMARY KEY (CODIGO) USING INDEX TABLESPACE TS_INDICES
) ;
```

Ejecuta las sentencias en la Máquina Virtual utilizando el usuario secretaria.

Crea un script que sea capaz de borrar todo el esquema creado. Para la creación de dicho script de forma automatizada hay distintas posibilidades:

- Ejecutar una sentencia que genere las instrucciones necesarias para el borrado de las tablas existentes.
- Utilizar las opciones de generación del script (de DDL) de Data Modeler para hacer un borrado previo de todos los elementos.

Ejecuta el script de borrado y tras comprobar que ha funcionado correctamente vuelve a ejecutar el script de creación nuevamente utilizando el usuario secretaria. Como se indicaba al comienzo de esta práctica es importante tener en un solo script la posibilidad de borrarlo todo y volverlo a generar porque, probablemente, es una acción que nos veremos obligados a realizar si observamos un error de diseño o un cambio importante sobre el diseño relacional final.

### 3. Importación de Datos

Importar los datos en la tabla TITULACION haciendo uso del fichero correspondiente. Con SqlDeveloper, pulsar en Titulacion con el botón derecho y elegir Importar. Seguir los pasos. Si es necesario, modificar el tamaño de la columna NOMBRE de TITULACION para que sea VARCHAR2(100). Ojo, Código y Créditos tienen asociada la misma columna del Excel, por lo que hay que cambiarlo.

Debemos también importar los datos de Asignaturas haciendo uso del archivo excel correspondiente en las tablas que sea conveniente. Téngase en cuenta que en este momento el alumno necesitará preprocesar los datos de la hoja excel antes de la importación y, posiblemente, retocar alguno de los campos de su modelo relacional de manera previa.

Asistente de Importación de Datos: Paso 4 de 5

#### Definición de Columna

Para cada columna de la lista Columnas de Datos de Origen a la izquierda, seleccione una columna Tabla de Destino a la derecha.

Coincidir por: Nombre

**Columnas de Datos de Origen**  
TITULAC.  
OFERTADA  
CÓD.  
REFERENCIA  
ASIGNATURA  
CURSO  
CRÉD.\_TEORÍA  
DURACIÓN  
OTRO\_IDIOMA

**Columnas de Datos de Destino**  
Nombre: IDIOMAS  
Tipo de Dato: VARCHAR2  
Tamaño/Precisión: 20  
☒ ¿Nulo? Valor por Defecto:  
Comentario:  
Datos:  
Solo un grupo

Ayuda < Atrás Siguiente > Terminar Cancelar

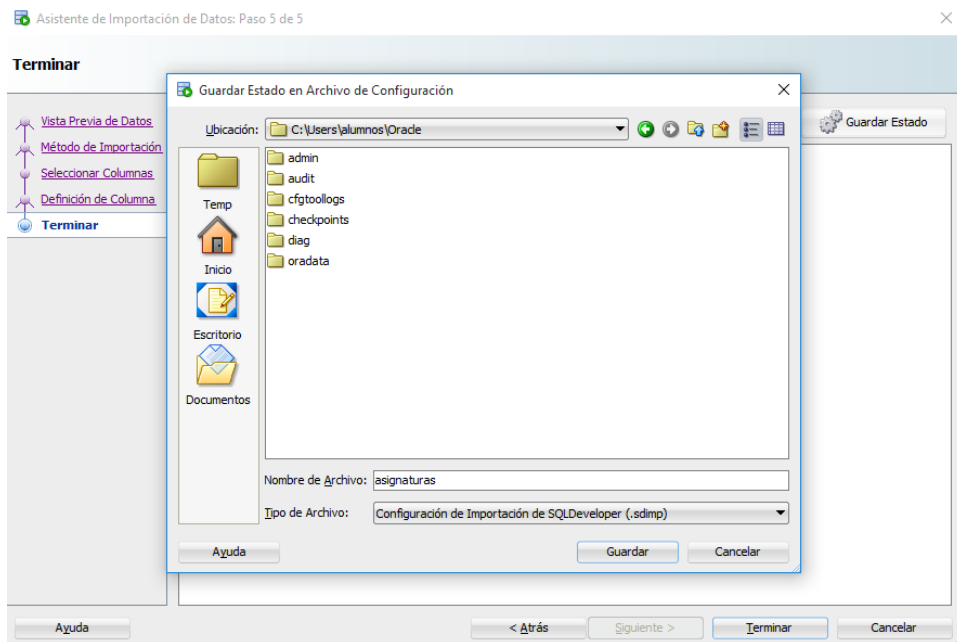
Asistente de Importación de Datos: Paso 5 de 5

#### Terminar

Guardar

**Resumen de Importación**  
Conexión de Destino: secretaria  
Archivo de Origen: C:\Users\alumnos\Oracle\Oferta asignaturas Modificado.xlsx  
Campos Seleccionados  
[X] Campo: TITULAC. --> TITULACION\_CODIGO  
[X] Campo: OFERTADA --> OFERTADA  
[X] Campo: CÓD. --> CODIGO  
[X] Campo: REFERENCIA --> REFERENCIA  
[X] Campo: ASIGNATURA --> NOMBRE  
[X] Campo: CURSO --> CURSO  
[X] Campo: CRÉD.\_TEORÍA --> CREDITOS  
[X] Campo: DURACIÓN --> DURACION  
[X] Campo: OTRO\_IDIOMA --> IDIOMAS  
Campos no Seleccionados  
Método de Importación: Insertar

Guardar el estado de importación:



## 4. Tablas Externas

Para crear una tabla externa, primero hay que dar de alta un directorio en Oracle. Para ello vamos a buscar un directorio donde el usuario de Oracle tenga acceso. Por ejemplo, podemos usar el directorio: C:\Users\app\alumnos\Oracle.

1. Creamos en ese directorio el fichero de alumnos.
2. Nos conectamos con el usuario system o sys.
3. Ejecutamos:

```
create or replace directory directorio_ext as 'C:\Users\app\alumnos\Oracle';
```

4. Damos permiso al usuario SECRETARIA para leer y escribir en el directorio:

```
grant read, write on directory directorio_ext to SECRETARIA;
```

5. **Conectarse como SECRETARIA. CUIDADO: No crear la tabla siguiente en SYS ni en SYSTEM!!!**

6. Crear la tabla:

```
create table alumnos_ext
(
    ...
)
organization external
(
    default directory directorio_ext
    access parameters
(
    records delimited by newline
    skip 4
    fields terminated by ';'
)
location ('alumnos.csv')
```

);

7. Desde el usuario SECRETARIA probar a ejecutar sentencias SQL para leer, modificar, insertar... Por ejemplo:  
`SELECT * FROM ALUMNOS_EXT`. Investigar que ocurre con cada una de ellas.

8. Añadir los datos a la tabla ALUMNOS de SECRETARIA desde la tabla externa. Utilice:

```
INSERT INTO ALUMNO SELECT campo1, campo2, campo3, ... FROM ALUMNOS_EXT;
```

Para el campo ID de la tabla ALUMNO es posible que necesites el uso de una secuencia si, durante el modelado, no tuviste en cuenta la autogeneración de este campo. Para ello, crea primero una secuencia con la instrucción:

```
CREATE SEQUENCE SECRETARIA.SEQUENCE_ALUMNO_ID MINVALUE 1 INCREMENT BY 1;
```

Después solo será necesario utilizarlo dentro del select, en el orden en el que está tu atributo ID de la siguiente manera: `SEQUENCE_ALUMNO_ID.NEXTVAL`.

No olvides confirmar la transacción.

## 5. Índices.

Asegúrese de que la tabla ALUMNOS tiene clave primaria. Además, hay que crear algunos índices sobre los atributos más comunes para realizar consultas (recuerda que estos nuevos índices deberán residir en TS\_INDICES). Al menos uno de los índices debe ser sobre una función. Compruebe ahora los índices con `USER_INDEXES`.

¿En qué tablespace reside la tabla ALUMNOS? ¿Y los índices? (compruébelo consultando el diccionario de datos)

Sobre la tabla ASIGNATURA crear tres índices de tipo BITMAP sobre los atributos curso, carácter y cuatrimestre respectivamente. Estos índices también deberán residir en TS\_INDICES.

Verificar en el diccionario de datos que estos últimos tres índices son de tipo BITMAP.