

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 qué 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.

Si no se ha hecho ya, crea un usuario denominado *PEVAU* y asígnale por defecto un tablespace denominado *TS_PEAU*. 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. Asigna cuota al usuario para ese Tablespace.

Comprobar consultando el diccionario de datos que existen los tablespace *TS_PEAU* y *TS_INDICES*. Comprobar consultando el diccionario de datos que el tablespace por defecto del usuario *PEVAU* es *TS_PEAU*. Comprobar consultando el diccionario de datos los datafiles que tienen asociado *TS_PEAU* y *TS_INDICES*.

2. Creación del Esquema

Si ya tienes tu script ejecutado en la BD, vamos a cambiar el tablespace sobre el que se han creado todos los índices. Puedes usar una vista del diccionario de datos para saber cuáles son los índices que existen en el esquema de *PEVAU*. Una vez los sepas, para cada uno de ellos ejecuta:

```
ALTER INDEX <INDEX_NAME> REBUILD TABLESPACE TS_INDICES;
```

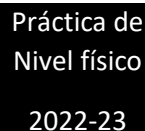
En otro caso, si aún no has ejecutado el script, o prefieres empezar de cero, desde el modelo E/R, te detallamos a continuación como hacerlo. **Se recomienda encarecidamente** repasar el modelo de acuerdo con los datos que posteriormente se importarán desde los distintos archivos excel y csv (archivos que se encuentran en la pestaña del trabajo en grupo). Es muy importante dedicarle tiempo a esta opción de forma que una vez tengamos el modelo relacional en la BD, se realicen el menor 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 *PEVAU*.



Una vez descargado, buscamos la tabla Vocaes y pulsamos con el botón derecho y elegimos Importar Datos:



DOCUMENTO	NOMBRE	APELLIDOS	TIPO	MATERIA	CODIGO	MATERIA
1	DI	VOCAL				
2	NOMBRE					
3	APELLIDOS					
4	TIPO					
5	MATERIA	CODIGO	MATERIA			
6	CARGO					

Seleccionar el archivo de origen (Vocales.xlsx)

Asistente de Importación de Datos: Paso 1 de 5

Vista Previa de Datos

Archivo de Datos de Importación: C:\Users\Usuario_UMA\Downloads\Vocales.xlsx

Formato de Archivo

☒ Cabecera Omitir Filas: 0

Formato: excel 95-2003 (.xls) ☒ Límite de Vista Previa de Filas: 100

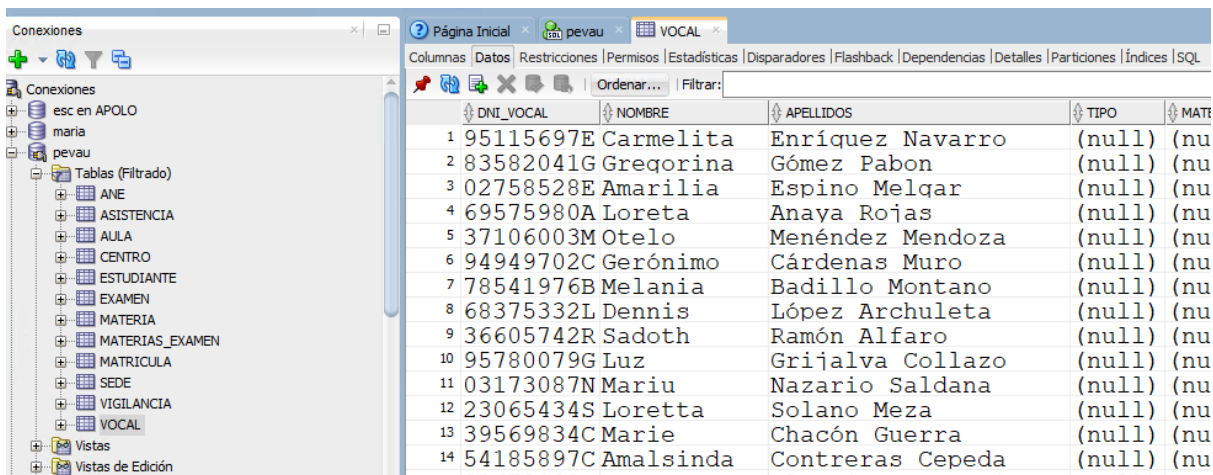
Contenido del Archivo

DOCUMENTO	NOMBRE	APELLIDOS
95115697E	Carmelita	Enríquez Na...
83582041G	Gregorina	Gómez Pabon
02758528E	Amarilla	Espino Melgar
69575980A	Loreta	Anaya Rojas
37106003M	Otelo	Menéndez M...
94949702C	Gerónimo	Cárdenas M...
78541976B	Melania	Badillo Mont...
68375332L	Dennis	López Archu...
36605742R	Sadoth	Ramón Alfaro
95780079G	Luz	Grijalva Coll...
03173087N	Mariu	Nazario Sald...
23065434S	Loretta	Solano Meza
39569834C	Marie	Chacón Guerra
54185897C	Amalsinda	Contreras C...
02670198A	Rocón	Altamirano ...

Y aceptar todas las opciones por defecto.

Si el nombre de la columna del fichero Excel y el de la tabla no coinciden, hay que indicarle cual es el nombre que se corresponde. Para ello asignar el nombre de la columna de la tabla con el nombre de la columna de la hoja Excel (cuando no se llaman igual o tiene una tilde, un espacio, etc). En este caso no debería hacer falta.

Una vez importados los datos de los vocales debería quedar una cosa así:

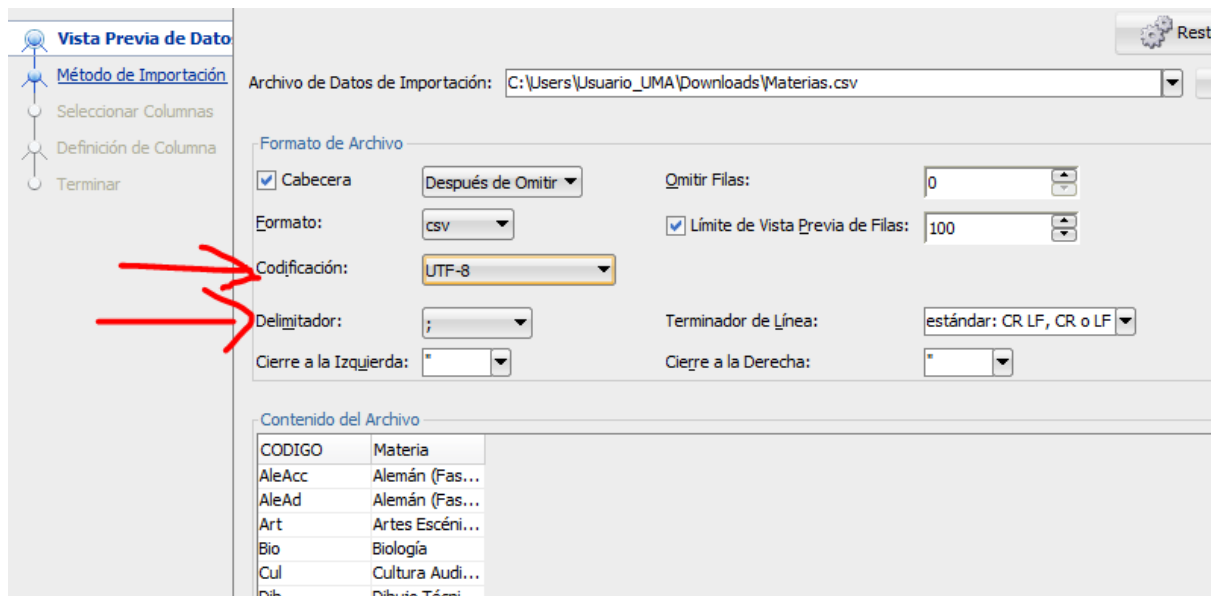


	DNI_VOCAL	NOMBRE	APELLIDOS	TIPO	MATE
1	95115697E	Carmelita	Enríquez Navarro	(null)	(nu
2	83582041G	Gregorina	Gómez Pabon	(null)	(nu
3	02758528E	Amarilia	Espino Melgar	(null)	(nu
4	69575980A	Loreta	Anaya Rojas	(null)	(nu
5	37106003M	Otelo	Menéndez Mendoza	(null)	(nu
6	94949702C	Gerónimo	Cárdenas Muro	(null)	(nu
7	78541976B	Melania	Badillo Montano	(null)	(nu
8	68375332L	Dennis	López Archuleta	(null)	(nu
9	36605742R	Sadoth	Ramón Alfaro	(null)	(nu
10	95780079G	Luz	Grijalva Collazo	(null)	(nu
11	03173087N	Mariu	Nazario Saldana	(null)	(nu
12	23065434S	Loretta	Solano Meza	(null)	(nu
13	39569834C	Marie	Chacón Guerra	(null)	(nu
14	54185897C	Amalsinda	Contreras Cepeda	(null)	(nu

NOTA: Si te ha dado error al importar, **lee la descripción del error antes** de preguntar al profesor. Quizás no hayas dado quota en algún tablespace necesario.

Materias

Se trata de un fichero CSV pero que usa como delimitador “;” en vez de la coma. Modificar también la codificación para el tratamiento correcto de acentos:



Vista Previa de Datos

Método de Importación

Archivo de Datos de Importación: C:\Users\Usuario_UMA\Downloads\Materias.csv

Formato de Archivo

☒ Cabecera Después de Omitir Omitir Filas: 0

Formato: csv ☒ Límite de Vista Previa de Filas: 100

Codificación: UTF-8

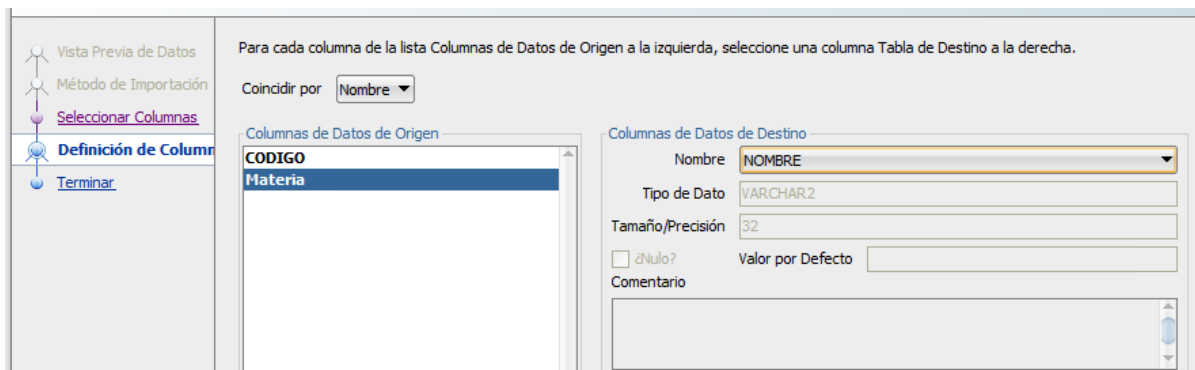
Delimitador: ; Terminador de Línea: estándar: CR LF, CR o LF

Cierre a la Izquierda: " Cierre a la Derecha: "

Contenido del Archivo

CODIGO	Materia
AleAcc	Alemán (Fas...
AleAd	Alemán (Fas...
Art	Artes Escéni...
Bio	Biología
Cul	Cultura Audi...
Dib	Dibujo Téni...

Como ya se ha indicado, si el nombre de la columna del fichero Excel y el de la tabla no coinciden, hay que indicarle cual es el nombre que se corresponde. En la siguiente figura, el nombre del atributo en la tabla de la base de datos es **NOMBRE** mientras que en el archivo se llama **Materia**:



Vista Previa de Datos
Método de Importación
Seleccionar Columnas
Definición de Columnas
Terminar

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

Columna
CODIGO
Materia

Columnas de Datos de Destino

Nombre: NOMBRE

Tipo de Dato: VARCHAR2

Tamaño/Precisión: 32

☐ ¿Nulo?

Comentario

Valor por Defecto

Sedes

Realizar la misma operación con las sedes, teniendo en cuenta que los atributos **RESPONSABLE** y **SECRETARIO** pueden no coincidir.

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 del software de Oracle tenga acceso**. Por ejemplo, podemos usar el directorio: `C:\Users\app\alumnos\admin\orcl\dpdump`.

1. Descargamos del campus virtual y ponemos en el directorio el archivo de estudiantes (datos-estudiantes-pevau.csv)

datos-estudiantes-pevau.csv: Bloc de notas

Archivo Edición Formato Ver Ayuda

```

CENTRO;Nombre;Apellido1;Apellido2;DNI/NIF;DETALLE MATERIAS
C.C. JUAN XXIII;Paula;Gámez;Baños;88126719U;Historia de España, Lengua Castellana y Literatura, Francés
C.C. JUAN XXIII;Marcos;Tejada;Dueñas;40733372M;Historia de España, Lengua Castellana y Literatura, Ingl
C.C. JUAN XXIII;Marta;Obellón;Galeno;79642372V;Economía de la Empresa
C.C. JUAN XXIII;Sofia;Carrillo;Rivera;67183718X;Biología, Química
C.C. JUAN XXIII;Nahia;De la Cruz;Carvajal;36424818E;Diseño, Matemáticas Aplicadas a las CCSS
C.C. JUAN XXIII;Nahia;Rubio;Munera;56549023P;Historia de España, Lengua Castellana y Literatura, Inglés
C.C. JUAN XXIII;Martina;Tablada;Gillén;19071747L;Historia de España, Lengua Castellana y Literatura, In
C.C. JUAN XXIII;Alba;Ospino;Gamboa;57571072G;Historia de España, Lengua Castellana y Literatura, Inglés
C.C. JUAN XXIII;Maria;Salas;Parado;L2162539S;Historia de España, Lengua Castellana y Literatura, Inglés
C.C. JUAN XXIII;Oliver;Bazán;Paniagua;17992067F;Biología, Química
C.C. JUAN XXIII;Paula;Maurer;Paz;G6395402I;Historia de España, Lengua Castellana y Literatura, Inglés (
C.C. JUAN XXIII;Jorge;Pastrana;Mijares;64086121Z;Geografía, Historia del Arte
C.C. JUAN XXIII;Yasmin;Borges;Olguín;30981745P;Economía de la Empresa, Matemáticas Aplicadas a las CCSS
C.C. JUAN XXIII;Elisa;Munera;Rosales;16533715U;Historia de España, Lengua Castellana y Literatura, Ingl
C.C. JUAN XXIII;Fernando;Uribe;Dorado;86978112R;Historia de España, Lengua Castellana y Literatura, Ing
C.C. JUAN XXIII;Carmen;Arjona;Maldonado;97646710H;Historia de España, Lengua Castellana y Literatura, I
C.C. JUAN XXIII;Erik;Iglesias;Olivar;11735789G;Historia de España, Lengua Castellana y Literatura, Ingl
C.C. JUAN XXIII;Alvaro;Jara;Sotomayor;43417641R;Economía de la Empresa, Matemáticas Aplicadas a las CCS
C.C. JUAN XXIII;Max;Loredo;Huerta;33535088U;Historia de España, Lengua Castellana y Literatura, Inglés
C.C. JUAN XXIII;Julia;Ramírez;Iglesias;90043357S;Historia de España, Lengua Castellana y Literatura, In
C.C. JUAN XXIII;Jon;Venegas;Pineda;93259086T;Historia de España, Lengua Castellana y Literatura, Inglés
C.C. JUAN XXIII;Lucas;Calles;Ferreira;85295559F;Historia de España, Lengua Castellana y Literatura, Ing
  
```

2. Copiamos en ese directorio el fichero de estudiantes.
3. Nos conectamos con el usuario system.
4. Ejecutamos:

```
create or replace directory directorio_ext as 'C:\Users\app\alumnos\admin\orcl\dpdump';
```

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

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

5. Conectarse como PEVAU. **CUIDADO: No crear la tabla siguiente en SYSTEM!!! → Cierra la sesión de System**

6. Crear la tabla de Estudiantes_EXT que vienen de un fichero.

```

create table estudiantes_ext
(
  ...
)
ORGANIZATION EXTERNAL (
  TYPE ORACLE_LOADER
  DEFAULT DIRECTORY directorio_ext
  ACCESS PARAMETERS (
    RECORDS DELIMITED BY NEWLINE
    CHARACTERSET UTF8
    FIELDS TERMINATED BY ';'
    OPTIONALLY ENCLOSED BY '"'
    MISSING FIELD VALUES ARE NULL
    (centro, nombre, apellido1, apellido2, dni, telefono, detalle)
  )
  LOCATION ('datos-estudiantes-pevau.csv')
);
  
```



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

8. Como no tenemos el correo electrónico de los estudiantes vamos a inventarnos uno para esta práctica. Así podemos hacer una vista preparada para cargar los datos en la tabla ESTUDIANTE. Creamos la vista así:

`create or replace view v_estudiantes as`

`SELECT dni, nombre, apellido1 || ' ' || apellido2 apellidos,`

`telefono,`

`substr(nombre,1,1) || apellido1 || substr(dni,6,3) || '@uncorreo.es' correo,`

`centro, detalle_materias`

`FROM estudiantes_ext`

`where dni is not null;`

Comprobar que los datos salen correctamente.

Ejecuta la instrucción: `select DISTINCT centro from v_estudiantes`

¿Cuántos centros salen?

5. Índices.

Asegúrate de que la tabla ESTUDIANTE 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, por ejemplo upper (apellidos). Comprueba ahora los índices con USER_INDEXES.

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

Aunque aún no hemos cargado los datos de los ESTUDIANTE, crea un índice de tipo BITMAP sobre el atributo que indica el código del centro. Este índice también deberá residir en TS_INDICES.

Verificar en el diccionario de datos que este último índice es de tipo BITMAP.

6. Vista Materializada.

Crea una Vista materializada VM_ESTUDIANTES con los datos de la última carga de estudiantes. La vista se debe refrescar cada día (refresco forzado) a las 00:00 horas.

7. Sinónimos

Crear un sinónimo público denominado S_ESTUDIANTES para el objeto creado en el apartado anterior

8. Centros

Asigna permisos al usuario PEVAU para crear secuencias;

Crear la secuencia SEQ_CENTROS.

Crea un trigger que modifique el código de centro si no se le suministra en el insert:

```
create or replace trigger tr_centros
before insert on centro for each row
begin
if :new.codigocentro is null then
:new.codigocentro := SEQ_CENTROS.NEXTVAL;
end if;
END tr_centros;
```

Como aún no sabemos la sede de cada centro modificamos el atributo que indica el código de sede de cada centro para permitir valores nulos

Prueba a insertar un centro cualquier con:

```
insert into centro (nombre) values ('Ejemplo');
select * from Centro;
rollback; -- para borrarlo
```

Insertamos los centros:

```
insert into centro (nombre) select distinct centro from
v_estudiantes;
select * from centro;
```

Si todo ha ido bien, confirmamos:

```
Commit;
```

9. Estudiante

Hay que obtener los datos de la tabla ESTUDIANTE haciendo el join de la vista V_ESTUDIANTE y CENTRO, obteniendo el código del centro en lugar de su nombre e insertar en la tabla ESTUDIANTE los datos:

```
insert into estudiante
SELECT
dni, v.nombre, apellidos, telefono, correo, codigocentro
-- Si hay campos adicionales en la tabla estudiantes, ponerlos
a null
FROM ...
```