

Usuarios, permisos y tablespaces.

Administración de Sistemas Gestores de Bases de Datos.

Paulino Esteban Bermúdez R.

IES. ALONSO DE AVELLANEDA Alcalá de Henares, Madrid.

Contenido

Ejercicio 1: Creación de un tablespace.....	2
Ejercicio 2: Crear usuarios para tus compañeros.....	3
Ejercicio 3: PROFESOR con permisos especiales.	4
Ejercicio 4: Creación de las tabla.....	5
Ejercicio 5: Dar acceso a otros usuarios a un campo de tus tablas.....	5
Ejercicio 6: Llena el tablespace CARRERAS.....	7

Ejercicio 1: Creación de un tablespace

Crea un tablespace

- De nombre PARAOTROS.
- Con dos ficheros que se guardarán en /datos/paraotros.
- El tamaño de cada fichero será como máximo de 100 MBytes.

Para crear el tablespace en el sistema, es necesario crear en el sistema operativo, la carpeta 'datos' con ruta en la raíz. '/datos'. Seguidamente nos conectamos con privilegios de sys para crear el tablespace con los dos ficheros .dbf. Para ello ejecutamos:

```
create tablespace PARAOTROS
datafile '/datos/paraotros1.dbf', size 800k
autoextend on
next 200k
maxsize 100M,
'/datos/paraotros2.dbf'
size 800k
autoextend on
next 200k
maxsize 100M;
```

```
#!/bin/sh
#####
# ----- EJ1. Creacion de un tablespace ----- #
#Crea tablespace                                #
# De nombre PARAOTROS.                          #
# Con dos ficheros que se guardaran en ~/datos/paraotros. #
# El tamaño de cada fichero sera como maximo de 100 MByte. #
#####

# Accedemos al directorio donde crear el directorio
cd /
# Creamos el directorio del tablespace
mkdir datos/
# Creamos el tablespace en sqlplus como administrador.
# |
# + Asignamos un nombre al tablespace en este caso 'PARAOTROS'
# + Escribimos la ruta donde se almacenará el datafile '/datos/paraotros1.dbf',
# | paraotros1.dbf es el tipo de archivo donde guarda mis datos del tablespace.
# + Asignamos el tamaño INICIAL del tablespace, auto extenderse 'ON' con el siguiente tamaño '200k'
# + Con tamaño maximo de 100M.
#
# => Como nos pide que sea un tablespace pero con DOS ficheros DBF, creamos de la misma forma 'paraotros2
# NOTA: IMPORTANTE terminar el tablespace con el punto y coma (;).

sqlplus / as sysdba<<EOF
create tablespace paraotros
datafile '/datos/paraotros1.dbf',
size 800k
autoextend on
next 200k
maxsize 100M,
'/datos/paraotros2.dbf'
size 800k
autoextend on
next 200k
maxsize 100M;
exit
EOF
```

Ilustración 1 Creación de tablespace

Ejercicio 2: Crear usuarios para tus compañeros.

Creas un usuario para ti, uno para cada uno de tus compañeros, y uno para el profesor. La contraseña inicial será la misma que el nombre, excepto en tu usuario que debería ser una contraseña secreta:

ABABEI	BERMUDEZ	BLANCO	CASTELLANO
CIOBANU	COBERTERA	CONDORI	CONTONENTE
CRUZ	FERNANDEZ	GONZALEZ	LOPEZ
MARTINEZ	MBA	RAMOS	RANERA
ROLLAN	SIMON	UTRERO	VILLARTA
YUBERO	PROFESOR		

Para crear el script SQL que genere el resto de usuarios lanzo desde el sistema un bucle que lea el archivo 'lista' que contiene los nombres de los usuarios para la base de datos, guarda el usuario y contraseña con un tamaño de 10 Mb de cuota y como tablespace por defecto 'paraotros', además de ello, se le darán los privilegios necesarios para conectarse y realizar las tareas básicas de consultas en la base de datos. Al igual lo realizo con mi usuario pero con tablespace 'users', como su tablespace por defecto.

Por último, lanzo el script con sys para que cree los usuarios en la BD.

```
for i in `cat lista`;
do
echo "create user $i identified by $i quota 10M on paraotros default tablespace paraotros;" >> crear.sql
echo "grant connect, resource to $i;">> crear.sql

echo "create user BERMUDEZ identified by BERMUDEZ51 default tablespace USERS quota unlimited on users;" >> crear.sql

echo "grant connect, resource to BERMUDEZ;" >> crear.sql
```

```
#!/bin/bash

#####
# ----- Ejercicio 2 : Crear usuarios para tus compañeros. ----- #
# #
# Crea un usuario para ti, uno para cada uno de tus compañeros, #
# y uno para el profesor. La contraseña inicial será la misma #
# que el nombre, excepto en tu usuario que debería ser una #
# contraseña secreta. #
# ----- #
# Tu propio usuario tendrá como tablespace por defecto USERS, #
# y los demás PARAOTROS. Los usuarios necesitan poder conectarse #
# a la base de datos y crear tablas en su tablespace por defecto. #
# Los usuarios de los otros alumnos tendrán una cuota de 10 Mb #
# en PARAOTROS, y no podrán escribir nada en USERS. #
# | #
# + El resto de usuarios podrán conectarse con sqlplus #
# | USUARIO/USUARIO@HOST:1521/SID #
# + Ide a algunos compañeros que comprueben su usuario. #
# + Ide al profesor que compruebe su usuario. #
#####

# ==> Creación del script SQL para los usuarios. #
#####

# El resto de usuarios.
rm crear.sql
for i in `cat lista`;
do
echo "create user $i identified by $i quota 10M on paraotros default tablespace paraotros;">>crear.sql
echo "user and password === OK!"
echo "grant connect, resource to $i;">>crear.sql
echo "grant connect, resource to BERMUDEZ;">>crear.sql
echo "Privileges === OK!"
done
# mi usuario BERMUDEZ
echo "create user BERMUDEZ identified by BERMUDEZ51 default tablespace USERS quota unlimited on users;">>crear.sql <-- CREACION DE USUARIO BERMUDEZ CON TABLESPACE USERS POR DEFECTO.
echo "user and password === OK!"
echo "grant connect, resource to BERMUDEZ;">>crear.sql
echo "Privileges === OK!"
# creo los usuarios nuevos lanzando el script de SQL
sqlplus / as sysdba @crear.sql <-- CREACION DE LOS USUARIOS CREADOS
```

Ilustración 2 Creación de usuarios.

Haz que el usuario PROFESOR tenga permisos de lectura en las vistas de sistema DBA_TABLES, DBA_TABLESPACES, DBA_TSQUOTAS y DBA_DATAFILES.

Seguidamente, añadimos al role las vistas de sistema que se nos solicita, escribimos:

```
create role permisosProfe;
```

Seguidamente, añadimos al role las vistas de sistema que se nos solicita, escribimos:

```
Grant select on DBA_TABLES to permisosProfe;
```

```
Grant select on DBA_TABLESPACE to permisosProfe;
```

```
Grant select on DBA_DATA_FILES to permisosProfe;
```

```
Grant select on DBA_TS_QUOTAS to permisosProfe;
```

Por último, asignamos el role creado al usuario profesor.

Grant permisosProfe to PROFESOR;

```
#!/bin/bash
#####
# Ejercicio 3 :PROFESOR con permisos especiales. #
# Haz que el usuario PROFESOR tenga permisos de lectura en las vistas de sistema #
# DBTABLES y DBTABLESPACES. #
#####
# PROFESOR con permisos de lectura en las V.sistema DBA_TABLES & DBA_TABLESPACES. #
# Conectamos con sys. para dar los privilegios. #
#####

# PROFESOR con permisos de lectura en las V.sistema DBA_TABLES & DBA_TABLESPACES &
# DBA_DATA_FILES & DBA_TS_QUOTAS.
# Conectamos con sys. pero damos los privilegios sobre la tabla de BERMUDEZ.

sqlplus / as sysdba<<EOF
drop role permisosProfe;          <-- BORRAMOS EL ROLE CON LOS PERMISOS DE PROFESOR (SI EXISTE).
create role permisosProfe;        <-- CREAMOS EL ROLE CON LOS PERMISOS DE PROFESOR.

GRANT SELECT ON DBA_TABLES TO permisosProfe;          / <-- DBA_TABLES
GRANT SELECT ON DBA_TABLESPACES TO permisosProfe;    | <-- DBA_TABLESPACE
| | | | | | | | | | | | | | | | | | | | | | | | | | Privilegios del role <
GRANT SELECT ON DBA_DATA_FILES TO permisosProfe;    | <-- DBA_DATA_FILE
GRANT SELECT ON DBA_TS_QUOTAS TO permisosProfe;     \ <-- DBA_TS_QUOTAS

GRANT permisosProfe TO PROFESOR;          <-- ASIGNACION DEL ROLE AL USUARIO PROFESOR.
exit
EOF
```

Ilustración 3 Permisos especiales para PROFESOR.

Ejercicio 4: Creación de las tabla.

Cambia la contraseña de tu usuario, si no lo has hecho ya, para que ningún compañero pueda utilizarlo (con ALTER USER).

1. Crea un tablespace de nombre CARRERAS, con un datafile en el directorio/datos/carreras.
2. Con tu propio usuario, crea las tablas en ese tablespace. Utiliza el script carreras-coches.sql para la creación de las tablas Tendrás que modificar el script para que tenga en cuenta el nuevo tablespace.

Para crear el tablespace carreras, hacemos el mismo proceso que en el ejercicio 1, pero con la ruta que nos indica el enunciado '/datos/carreras'.

Para lanzar mediante sqlplus el script del enunciado dos, escribimos, sqlplus usuario/contraseña@ip/SID @script_a_ejecutar, entonces:

sqlplus bermudez/BERMUDEZ51@10.1.35.51/asir

```
#####
# Ejercicio 4 : Creacion de las tablas.                                     #
#                                                                           #
# Cambia la contraseña de tu usuario, si no lo has hecho ya, para que ningún compañero pueda utilizarlo #
# (conALTER USER).                                                         #
#                                                                           #
# 1. Crea un tablespacede nombre CARRERAS, con un datafile en el directorio /datos/carreras.             #
# 2. Con tu propio usuario, crea las tablas en ese tablespace.             #
# 2.1 Utiliza el script carreras-coches.sql para la creación de las tablas. #
# 2.2 Tendrás que modificar elscript para que tenga en cuenta el nuevo tablespace #
#####

# Accedemos al directorio donde crear el directorio
cd /
# Creamos el directorio del tablespace
mkdir datos/carreras
# Creamos el tablespace en sqlplus con SYS.
sqlplus / as sysdba<<EOF
drop tablespace carreras;                                <-- Borramos (si existe) el tablespace CARRERAS
create tablespace carreras                                <-- Creamos el tablespace CARRERAS
datafile '/datos/carreras/carreras.dbf'                    <-- Ruta de almacenamiento del tablespace en el SO Centos 7
                                                            (archivo dbf)
size 800k                                                  <-- Tamaño inicial del tablespace
autoextend on
next 200k
maxsize 100M;                                             <-- Tamaño máximo
alter user BERMUDEZ default tablespace carreras quota on unlimited; <-- asignación de cuota infinita del tablespace
exit                                                       'CARRERAS' a ususario BERMUDEZ
EOF
# Lanzamos el script de carreras de SQL con BERMUDEZ Modificado para que almacene las tablas en el TABLESPACE CARRERAS.
sqlplus BERMUDEZ/BERMUDEZ51@10.1.35.51/asir @carreras-coches.sql <-- Creo las tablas del script carreras-coches.sql para BERMUDEZ
exit
```

Ilustración 4 Creación de tablas CARRERAS-COCHES.SQL

Ejercicio 5: Dar acceso a otros usuarios a un campo de tus tablas.

Tras la importación, haz que las tablas puedan ser leídas por el resto de los usuarios:

Haz que el resto de los usuarios pueda realizar SELECT sobre tus tablas.

Crea sinónimos en todos los usuarios para que puedan acceder a tus tablas sin problemas

- Por ejemplo, el usuario profesor debería poder ejecutar

```
SELECT * FROM CIRCUITOS
```

puesto que habrás creado un sinónimo del tipo

```
CREATE PUBLIC SYNONYM CIRCUITOS FOR MIUSUARIO.CIRCUITOS;
```

Después, haz que puedan escribir en un campo de la tabla RESULTADOS: Crea un nuevo campo en la tabla RESULTADOS: VALORACION, de tipo VARCHAR (20). Tendrá como únicos valores posibles MALO, ACEPTABLE, BUENO y MUY BUENO.

Da permisos al resto de usuarios para poder leer y actualizar este campo, con una orden GRANT.

•<http://stackoverflow.com/questions/14462353/grant-alter-on-only-one-column-in-table>

- Pide a algún otro compañero que compruebe que funciona
- Pide al profesor que compruebe que funciona

Para crear los sinónimos de las tablas ejecutamos:

Create or replace public synonym NOMBRE_TABLA for USUARIO.SU_TABLA;

Como las tablas son del usuario BERMUDEZ, en el comando anterior en USUARIO → BERMUDEZ, el siguiente paso es que en **nombre_tabla** y **su_tabla** escribir el nombre de la tabla a la que deseamos hacer su sinónimo. Ejemplo:

Create or replace public synonym carreras for BERMUDEZ.CARRERAS;

Después de crear los sinónimos, creamos un role para asignar al resto de usuarios de la BD, el privilegio de hacer 'select' sobre las tablas de BERMUDEZ. Del mismo modo, como se crearon los usuarios con sys en el ejercicio 2, hacemos lo mismo pero asignando el role, LECTURATABLAS.

```
#!/bin/bash

# Ejercicio 5: Dar acceso a otros usuarios a un campo de tus tablas
# Tras la importación, haz que las tablas puedan ser leídas por el resto de usuarios:
# Haz que el resto de usuarios pueda realizar SELECT sobre tus tablas.
# Crea sinónimos en todos los usuarios para que puedan acceder a tus tablas sin problemas
# * Por ejemplo, el usuario profesor debería poder ejecutar
#   SELECT * FROM CIRCUITOS;
# Puesto que habrás creado un sinónimo del tipo:
#   CREATE PUBLIC SYNONYM CIRCUITOS FOR USUARIO.CIRCUITOS;

sqlplus / as sysdba<EOF

create or replace public synonym escuderias for BERMUDEZ.escuderias;  <-- SINONIMO DE LA TABLA ESCUDERIAS
create or replace public synonym pilotos for BERMUDEZ.pilotos;        <-- SINONIMO DE LA TABLA PILOTOS
create or replace public synonym prototipos for BERMUDEZ.prototipos;   <-- SINONIMO DE LA TABLA PROTOTIPOS
create or replace public synonym carreras for BERMUDEZ.carreras;        <-- SINONIMO DE LA TABLA CARRERAS
create or replace public synonym resultados for BERMUDEZ.resultados;    <-- SINONIMO DE LA TABLA RESULTADOS
create or replace public synonym circuitos for BERMUDEZ.circuitos;      <-- SINONIMO DE LA TABLA CIRCUITOS

drop role LECTURATABLAS;  <-- BORRAMOS EL ROLE LECTURATABLAS (SI EXISTE)
create role LECTURATABLAS;  <-- CREAMOS EL ROLE LECTURATABLAS
-- ASIGNAMOS LOS PRIVILEGIOS DE 'SELECT' SOBRE LAS TABLAS DE BERMUDEZ
grant select on BERMUDEZ.escuderias to LECTURATABLAS;  <--> ESCUDERIAS
grant select on BERMUDEZ.pilotos to LECTURATABLAS;    <--> PILOTOS
grant select on BERMUDEZ.prototipos to LECTURATABLAS; <--> PROTOTIPOS
grant select on BERMUDEZ.carreras to LECTURATABLAS;   <--> CARRERAS
grant select on BERMUDEZ.resultados to LECTURATABLAS; <--> RESULTADOS
grant select on BERMUDEZ.circuitos to LECTURATABLAS;  <--> CIRCUITOS
exit
EOF
rm privilegiosOtroUsuarios.sql  <-- BORRAMOS EL ARCHIVO privilegiosOtroUsuarios.sql SI EXISTE EN EL SISTEMA
for i in `cat lista`;
do
  echo "grant LECTURATABLAS to $i;">privilegiosOtroUsuarios.sql  <-- ASIGNAMOS EL ROLE 'LECTURATABLAS' SOBRE LOS USUARIOS ALMACENADOS EN EL ARCHIVO 'LISTA'
  echo "Script terminado"
done

# Llamo al script 'privilegiosOtroUsuarios.sql' de SQL con sys en SQLPLUS
sqlplus / as sysdba @privilegiosOtroUsuarios.sql  <-- LANZAMOS EL SCRIPT SQL ANTERIORMENTE CREADO.
```

Ilustración 5 Dar permisos a otros usuarios a un campo de las tablas

Para realizar este ejercicio, es necesario alterar la tabla RESULTADOS con el usuario BERMUDEZ; con el mismo, añadimos la columna 'VALORACION' a la tabla y le indicamos los valores posibles que puede tener en dicho campo.

Seguidamente, como sys, creador del role LECTURATABLAS, le añadimos el 'update(valoración) de la tabla resultado del usuario BERMUDEZ. A partir de ahí, cualquiera que tenga el role LECTURATABLAS, puede cambiar el valor de VALIRACION.

```
#!/bin/bash

# Haz que puedan escribir en un campo de la tabla RESULTADOS:
# Crea un nuevo campo en la tabla RESULTADOS:VALORACION, de tipoVARCHAR(20).
# Tendrá como únicos valores posibles MALO,ACEPTABLE,BUENO y MUY BUENO.
# Da permisos al resto de usuarios para poder leer y actualizar este campo, con una orden GRANT
# * http://stackoverflow.com/questions/14462353/grant-alter-on-only-one-column-in-table
# - Pide a algún otro compañero que compruebe que funciona
# - Pide al profesor que compruebe que funciona.

# Modifico el tabla RESULTADOS para que puedan hacer cambios.
sqlplus BERMUDEZ/BERMUDEZ@10.1.35.51/ asir<<EOF
alter table RESULTADOS add VALORACION varchar(20) default 'BUENO' not null check (valoracion in ('MALO', 'ACEPTABLE', 'BUENO', 'MUY BUENO' ))
EOF

# LECTURATABLAS -> rol creado con anterioridad.
# Le añado el privilegio de UPDATE(VALORACION) para la tabla resultados.
# -- alter role LECTURATABLAS add update; => No creo que funcione pero por ahí lo ví.
sqlplus / as sysdba<<EOF
grant update(valoracion) on BERMUDEZ.RESULTADOS to LECTURATABLAS;
EOF
sleep 5
clear
echo "Cambiando campo VALORACION - Modo pruebas"
sleep 5
# PRUEBAS:
# - Usuario que realiza un cambio en valoracion=aceptable.
sqlplus PRUEBAS/PRUEBAS@10.1.35.51/ asir<<EOF
update RESULTADOS set valoracion='ACEPTABLE';
commit;
exit
EOF
sleep 5
clear
echo "Confirmando cambios - Modo pruebas"
sleep 5
# - Usuario que verifica el cambio del campo valoracion.
sqlplus UTRERO/UTRERO@10.1.35.51/ asir<<EOF
select * from resultados;
exit
EOF
echo "Confirmando cambios - Modo pruebas"
sleep 5
```

Ilustración 6 Escribir en el campo resultados(valoración).

Ejercicio 6: Llena el tablespace CARRERAS.

Llena de datos el tablespace CARRERAS.

Indica el método que utilizas para llenarlo de datos y qué mensaje de error aparece.

Amplía el tablespace con un nuevo datafile para se puedan insertar más datos.

Para llenar el tablespace CARRERAS, tenemos que realizar una consulta PLSQL que realice un bucle que vaya llenando el disco con datos en la tabla. Para ello:

```
begin
for i in 1 .. 100000 loop
insert into resultados(valoracion) values('MALO');
loop end;
end;
/

commit;
```

Una vez lleno el tablespace, son el usuario sys, amplio el tablespace CARRERAS o bien añadiendo otro fichero .dbf o redimensionando el espacio del tablespace.

- ➔ Con otro archivo DBF.
alter tablespace carreras
add datafile '/datos/carreras/MasCarreras.dbf'
- ➔ Redimensionando el tablespace.
alter tablespace carreras resize 1024M;

Para llenar la tabla de nuevo y ver el mensaje que nos indica que llenamos la 'quota', lanzamos de nuevo un bucle hasta cumplir con el cometido. En este segundo caso, uso una nueva tabla.

```
create table llenar (id number, texto varchar(255)) tablespace carreras;

begin
for i in 1 .. 50000 loop
insert into llenar (id,texto)
values(i,'lleno');
loop end;
end;
/

commit;
```



```

#!/bin/bash

#####
# Ejercicio 6 : Llena el tablespace CARRERAS.                                     #
# Llena de datos el tablespace CARRERAS                                         #
# - Indica el metodo que utilizas para llenarlo de datos y qué mensaje de error aparece #
# - Amplia el tablespace con un nuevo datafile para se puedan insertar mas datos.   #
#####

# Llenamos el tablespace CARRERAS con un bucle. Nos conectamos con el USER: BERMUDEZ.
sqlplus BERMUDEZ/BERMUDEZ51@10.1.35.51/asir<<EOF
begin
for i in 1 .. 100000 loop
insert into resultados(valoracion) values('MALO');
loop end;
end;
/
COMMIT;
EOF

# Ampliacion del datafile de CARRERAS.
sqlplus / as sysdba<<EOF
alter tablespace carreras
add datafile '/datos/carreras/MasCarreras.dbf'
size 2M
autoextend on
next 200k
maxsize 10M;
alter user BERMUDEZ quota 5M on carreras;
exit
EOF

# Llenamos de nuevo el tablespace con un bucle infinito.
sqlplus BERMUDEZ/BERMUDEZ51@10.1.35.51/asir<<EOF
declare
i NUMBER :=0;
begin
while i <= true
loop
insert into resultados(valoracion) values('MALO');
loop end;
end;
/
EOF

# Ampliación del tablespace carreras (sin crear otro dbf).
sqlplus / as sysdba<<EOF
alter tablespace carreras resize 1024;
exit
EOF

# Llenado de tablespace con otra tabla nueva.
sqlplus BERMUDEZ/BERMUDEZ51@10.1.35.51/asir<<EOF
drop table llenar;
create table llenar (id number, text varchar(255)) tablespace carreras;
begin
for i in 1 .. 50000 loop
insert into llenar (id,texto)
values(i,'lleno');
end loop;
end;
/
select * from llenar;

```

Ilustración 7 Llenar el tablespace carreras – Con tabla resultados.

```

declare
i NUMBER :=0;
begin
while i > -1
loop
insert into resultados(valoracion) values('MALO');
end loop;
end;
Informe de error -
ORA-01653: no se ha podido ampliar la tabla BERMUDEZ.RESULTADOS con 128 en el tablespace CARRERAS
ORA-06512: en línea 6
01653. 00000 - "unable to extend table %s.%s by %s in tablespace %s"
*Cause:      Failed to allocate an extent of the required number of blocks for
              a table segment in the tablespace indicated.
*Action:     Use ALTER TABLESPACE ADD DATAFILE statement to add one or more
              files to the tablespace indicated.

```

Ilustración 8 Error por falta de espacio en el tablespace CARRERAS.

```

Table LLENAR creado.

Error que empieza en la línea: 34 del comando :
begin
  for i in 1 .. 50000 loop
    insert into llenar (id,texto)
    values(i,'lleno');
  end loop;
end;
Informe de error -
ORA-01536: cuota de espacio excedida para tablespace 'CARRERAS'
ORA-06512: en línea 3
01536. 00000 - "space quota exceeded for tablespace '%s'"
*Cause:      The space quota for the segment owner in the tablespace has
              been exhausted and the operation attempted the creation of a
              new segment extent in the tablespace.
*Action:     Either drop unnecessary objects in the tablespace to reclaim
              space or have a privileged user increase the quota on this
              tablespace for the segment owner.

```

Ilustración 9 Error de espacio en la cuota, excedido para tablespace CARRERAS.