

ADMINISTRACIÓN DE BASE DE DATOS

CURSO 2017/18 - JUNIO

PRÁCTICA ABD ONG

GRUPO ABD02

Juan Pedro Martínez Marín juanpedro.martinez7@um.es

Miguel Angel Carrasco Garcia miguelangel.carrasco1@um.es

1. CONSULTAS DEL DICCIONARIO DE LA BASE DE DATOS	2
2. TABLESPACES	6
3. TABLAS	10
4. INSERCIÓN DE DATOS	23
5. ÍNDICES	40
6. VISTAS MATERIALIZADAS	45
7. TABLESPACES: tipos y modos	53
8. OPTIMIZACIÓN	65
9. USUARIOS, PERFILES, PERMISOS Y ROLES	74

1. CONSULTAS DEL DICCIONARIO DE LA BASE DE DATOS

1.1. Mostrar con un SELECT el tamaño total en MB de la SGA (tamaño).

El tamaño total de la SGA se obtiene sumando todos los elementos de memoria que la componen y se divide el resultado para obtener el tamaño en MB.

```
SELECT sum(value)/1024/1024 "TOTAL SGA (MB)" FROM v$sga;
```

TOTAL SGA (MB)	
1	510,84765625

1.2. Mostrar el tamaño del bloque de la base de datos, el nombre de la base de datos, y el nombre de la instancia a la que se está conectado.

En la vista dinámica de parámetros del sistema tenemos toda esta información. Hemos buscado los nombres de las filas concretas.

```
show parameter db_block_size;
show parameter db_name;
show parameter instance_name;
```

NAME	TYPE	VALUE
db_block_size	integer	8192
NAME	TYPE	VALUE
db_name	string	orcl11
NAME	TYPE	VALUE
instance_name	string	orcl11
'		

Como alternativa podemos hacerlo en una sola consulta, usando la vista dinámica de parámetros del sistema, donde tenemos toda esta información buscando los nombres de las filas concretas.

```
SELECT name parametro, value valor
FROM v$system_parameter
WHERE name in ('db_block_size','db_name','instance_name');
```

PARAMETRO	VALOR
1 db_block_size	8192
2 instance_name	orcl11
3 db_name	orcl11

1.3. Mostrar con un SELECT tres parámetros de inicialización de la base de datos que estén relacionados con el tamaño de la SGA (nombre del parámetro, descripción y valor). Justificar en qué afecta cada uno de ellos al tamaño de la SGA.

```
select name, description, value
from v$parameter
where name='sga_max_size' or name='shared_pool_size' or
name='large_pool_size';
```

NAME	DESCRIPTION	VALUE
1 sga_max_size	max total SGA size	536870912
2 shared_pool_size	size in bytes of shared pool	0
3 large_pool_size	size in bytes of large pool	0

sga_max_size: Establece el máximo tamaño que puede alojar la SGA cuando se levanta la instancia de base de datos. Este parámetro permitirá aumentar el tamaño de la SGA sin necesidad de reiniciar la instancia, teniendo en cuenta que el total de la SGA no exceda este parámetro.

shared_pool_size: Como su propio nombre indica, indica el tamaño en bytes de la Shared Pool. Que contiene información relevante sobre sentencias SQL, código ejecutable PL/SQL, parámetros del sistema y referentes a las tablas del DD.

large_pool_size: Como ocurría con la anterior, su propio nombre indica, que indica el número de bytes para el área “Large Pool”. Para conocer su importancia, hay que ver el uso del área opcional mencionada. Este área permite, que se pueda disponer de más memoria para las sesiones, búferes de mensajes en ejecución paralela o para Recovery Manager I/O Slaves.

1.4. Mostrar con un SELECT las 3 últimas sentencias SQL cargadas en la shared-pool (texto de la sentencia, fecha y hora en que se cargó).

Mostramos las últimas tres sentencias ejecutadas en el SQL_AREA, el cual se encuentra dentro de la SHARED_POOL.

```
select *
```

```
from ( select sql_text, first_load_time from v$sqlarea order by
first_load_time desc )
where rownum <4;
```

SQL_TEXT	FIRST_LOAD_TIME
1 select * from (select sql_text, first_load_time from v\$sqlarea order by first_load_time desc) where rownum <4	2018-03-07/17:53:25
2 select sql_text, last_load_time from v\$sqlarea where rownum <=3 order by last_load_time	2018-03-07/17:51:57
3 select sql_text, last_load_time from v\$sqlarea order by last_load_time	2018-03-07/17:51:44

1.5. Mostrar con una sola sentencia SELECT

a) los ficheros de datos y los tablespaces a los que están asociados, b) los ficheros de control, y c) los ficheros de redo de la base de datos. La primera columna indicará el tipo de fichero (tipo de fichero –DATAFILE|CONTROLFILE|REDOFILE-, nombre del fichero, y nombre del tablespace si es un fichero de datos).

La información relativa a los ficheros de datos aparece en DBA_DATA_FILES, los ficheros de control se pueden observar en la vista V\$CONTROLFILE y los ficheros de log en V\$LOGFILE. Unificamos la consulta mediante uniones y añadimos una columna para indicar el tipo de fichero. Los ficheros de control y de log, al no tener tablespace, le asignamos “null” a esa consulta.

```
select 'CONTROLFILE' tipoFichero, name, null tablespace
from v$controlfile
UNION
select 'DATAFILE' tipoFichero, file_name, tablespace_name from
dba_data_files
UNION
select 'REDOFILE' tipoFichero, member, null tablespace from
v$logfile;
```

Salida de Script | Resultado de la Consulta | SQL | Todas las Filas Recuperadas: 21 en 0,062 segundos

TIPOFICHERO	NAME	TABLESPACE
1 CONTROLFILE	C:\ORACLE\FLASH_RECOVERY_AREA\ORCL11\CONTROL02.CTL	(null)
2 CONTROLFILE	C:\ORACLE\ORADATA\ORCL11\CONTROL01.CTL	(null)
3 DATAFILE	C:\ORACLE\ORADATA\ORCL11\ADBD_DATOS01.DBF	ADBD_DATOS
4 DATAFILE	C:\ORACLE\ORADATA\ORCL11\BDGRADO001	BDGRADO
5 DATAFILE	C:\ORACLE\ORADATA\ORCL11\BD_DATOS01	BD_DATOS
6 DATAFILE	C:\ORACLE\ORADATA\ORCL11\DG_DATOS01	DAG_DATOS
7 DATAFILE	C:\ORACLE\ORADATA\ORCL11\DSI_DATOS01	DSI_DATOS
8 DATAFILE	C:\ORACLE\ORADATA\ORCL11\SYSAUX01.DBF	SYSAUX
9 DATAFILE	C:\ORACLE\ORADATA\ORCL11\SYSTEM01.DBF	SYSTEM
10 DATAFILE	C:\ORACLE\ORADATA\ORCL11\UNDOTBS01.DBF	UNDOTBS1
11 DATAFILE	C:\ORACLE\ORADATA\ORCL11\USERS01.DBF	USERS
12 DATAFILE	C:\ORACLE\PRODUCT\11.2.0\DBHOME_1\DATABASE\ABD01_DATOS1.DBF	ABD01_DATOS
13 DATAFILE	C:\ORACLE\PRODUCT\11.2.0\DBHOME_1\DATABASE\ABD01_INDICES1.DBF	ABD01_INDICES
14 DATAFILE	C:\ORACLE\PRODUCT\11.2.0\DBHOME_1\DATABASE\ABD03_DATOS1.DBF	ABD03_DATOS
15 DATAFILE	C:\ORACLE\PRODUCT\11.2.0\DBHOME_1\DATABASE\ABD03_INDICES1.DBF	ABD03_INDICES
16 DATAFILE	C:\ORACLE\PRODUCT\11.2.0\DBHOME_1\DATABASE\ABDXX_DATOS1.DBF	ABDXX_DATOS
17 DATAFILE	C:\ORACLE\PRODUCT\11.2.0\DBHOME_1\DATABASE\ABDXX_INDICES1.DBF	ABDXX_INDICES
18 DATAFILE	C:\ORACLE\PRODUCT\11.2.0\DBHOME_1\DATABASE\ABDXX_INDICES2.DBF	ABDXX_INDICES
19 REDOFILE	C:\ORACLE\ORADATA\ORCL11\REDO01.LOG	(null)
20 REDOFILE	C:\ORACLE\ORADATA\ORCL11\REDO02.LOG	(null)
21 REDOFILE	C:\ORACLE\ORADATA\ORCL11\REDO03.LOG	(null)

1.6.Mostrar con un SELECT los tablespaces de rollback (undo) que están definidos en la base de datos (nombre del tablespace, tipo de contenido).

Se selecciona de dba_tablespaces el nombre del tablespace cuyo contenido sea UNDO.

```
select tablespace_name, contents
from dba_tablespaces
where contents='UNDO';
```

Salida de Script | Resultado de la Co... | SQL | Todas las Filas Recuperadas: 1 en 0,000 segundos

TABLESPACE_NAME	CONTENTS
1 UNDOTBS1	UNDO

2. TABLESPACES

2.1.Crear un tablespace permanente llamado ABDXX_DATOS, usando el fichero ‘ABDXX_DATOS1.DBF’, con un tamaño inicial de 256 KB, que pueda crecer automáticamente hasta un tamaño máximo de 1MB, con gestión manual del espacio de los segmentos, y con un valor por defecto de 32KB para todas las extensiones de todos los objetos que se creen en él. Importante: Si durante la realización de los ejercicios posteriores de esta práctica el tablespace se llenará, justo en ese momento (no ahora), se deberá añadir al tablespace un segundo fichero ‘ABDXX_DATOS2.DBF’ con las mismas características de espacio que el primero.

Cuando se crea un tablespace se especifica el nombre que se le quiera dar y el fichero al que pertenece. Se indica el tamaño inicial (256K) además del tamaño con el que se va incrementando el mismo (32K), por último se establece el tamaño máximo (1M).

Los valores descritos anteriormente reflejan los tamaños de nuestro caso. No son tamaños recomendados ni obligatorios.

Con las últimas dos líneas de la sentencia se establece que las extensiones deben ser manejadas localmente y la gestión del espacio de los segmentos será manual. La cláusula REUSE indica que si el archivo ya existe Oracle lo reutiliza y lo borra.

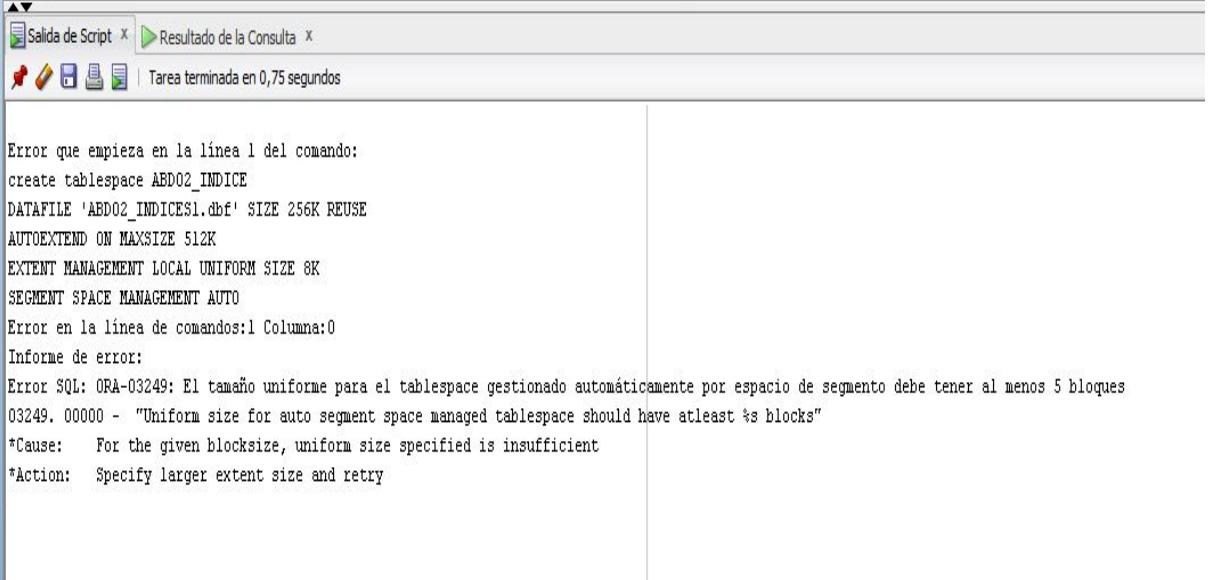
```
CREATE TABLESPACE ABD02_DATOS
DATAFILE 'ABD02_DATOS1.dbf' SIZE 256K REUSE
AUTOEXTEND ON MAXSIZE 1M
EXTENT MANAGEMENT LOCAL UNIFORM SIZE 32K
SEGMENT SPACE MANAGEMENT MANUAL;
```

tablespace ABD02_DATOS creado.

2.2.Crear un tablespace permanente para los índices llamado ABDXX_INDICES, usando el fichero ‘ABDXX_INDICES1.DBF’, con un tamaño inicial de 256KB que pueda crecer automáticamente hasta un tamaño máximo de 512KB, con extensiones del menor tamaño posible (justificar dicho tamaño) manejadas localmente, y con gestión automática del espacio de los segmentos. Importante: Si durante la realización de la práctica el tablespace se llenará, justo en ese momento, se deberá añadir al tablespace un segundo fichero ‘ABDXX_INDICES2.DBF’ con iguales características de espacio que el primero.

```
CREATE TABLESPACE ABD02_INDICES
```

```
DATAFILE 'ABD02_INDICES1.dbf' SIZE 256K
AUTOEXTEND ON MAXSIZE 512K
EXTENT MANAGEMENT LOCAL UNIFORM SIZE 8K
SEGMENT SPACE MANAGEMENT AUTO;
```



The screenshot shows a SQL developer interface with two tabs: 'Salida de Script' and 'Resultado de la Consulta'. The 'Resultado de la Consulta' tab is active and displays the following error message:

```
Error que empieza en la linea 1 del comando:
create tablespace ABD02_INDICE
DATAFILE 'ABD02_INDICES1.dbf' SIZE 256K REUSE
AUTOEXTEND ON MAXSIZE 512K
EXTENT MANAGEMENT LOCAL UNIFORM SIZE 8K
SEGMENT SPACE MANAGEMENT AUTO
Error en la linea de comandos:1 Columna:0
Informe de error:
Error SQL: ORA-03249: El tamaño uniforme para el tablespace gestionado automáticamente por espacio de segmento debe tener al menos 5 bloques
03249. 00000 -  "Uniform size for auto segment space managed tablespace should have atleast %s blocks"
*Cause: For the given blocksize, uniform size specified is insufficient
>Action: Specify larger extent size and retry
```

Al intentarlo la primera vez, se mostró un error como el que se aprecia en la imagen anterior.

Como se conoce que el tamaño de un bloque es de 8K, pues al necesitar 5 bloques, el tamaño que se necesita es de 40K. (Se sabe que el tamaño es de 8k, por los primeros ejercicios realizados).

```
CREATE TABLESPACE ABD02_INDICES
DATAFILE 'ABD02_INDICES1.dbf' SIZE 256K REUSE
AUTOEXTEND ON MAXSIZE 512K
EXTENT MANAGEMENT LOCAL UNIFORM SIZE 40K
SEGMENT SPACE MANAGEMENT AUTO;
```

```

create tablespace ABD02_INDICE
DATAFILE 'ABD02_INDICES1.dbf' size 256K reuse
autoextend on maxsize 512K
extent management local uniform size 40K
segment space management auto;

```

Resultado de la Consulta x Salida de Script x
Tarea terminada en 0,109 segundos
ONLINE, OFFLINE, FORCE, RETENTION.
tablespace ABD02_INDICE creado.

2.3. Mostrar con un SELECT los dos tablespaces creados (nombre del tablespace, tipo de contenido, tamaño de bloque, tamaño de la extensión inicial, tamaño de la extensión siguiente, porcentaje de incremento de las siguientes extensiones, forma de manejo de las extensiones, y forma de manejo del espacio de los segmentos).

```

Select TABLESPACE_NAME, CONTENTS, BLOCK_SIZE, INITIAL_EXTENT,
NEXT_EXTENT,PCT_INCREASE, EXTENT_MANAGEMENT, ALLOCATION_TYPE
from dba_tablespaces
where ( TABLESPACE_NAME='ABD02_DATOS' OR
TABLESPACE_NAME='ABD02_INDICES' );

```

	TABLESPACE_NAME	CONTENTS	BLOCK_SIZE	INITIAL_EXTENT	NEXT_EXTENT	PCT_INCREASE	EXTENT_MANAGEMENT	ALLOCATION_TYPE
1	ABD02_DATOS	PERMANENT	8192	32768	32768	0 LOCAL	UNIFORM	
2	ABD02_INDICES	PERMANENT	8192	40960	40960	0 LOCAL	UNIFORM	

2.4.Mostrar con un SELECT los ficheros de los dos tablespaces creados (nombre del tablespace, nombre del fichero, tamaño inicial, si/no puede crecer automáticamente, y tamaño máximo).

Para listar los parámetros solicitados usamos “dba_data_files” que contiene información relativa a los ficheros de la base de datos. Seleccionamos los atributos que se solicitan y por último filtramos los resultados para los ficheros de los tablespaces creados en los apartados anteriores.

```
Select TABLESPACE_NAME, FILE_NAME, BYTES, AUTOEXTENSIBLE, MAXBYTES  
from dba_data_files  
where ( TABLESPACE_NAME='ABD02_DATOS' OR  
TABLESPACE_NAME='ABD02_INDICES' );
```

TABLESPACE_NAME	FILE_NAME	BYTES	AUTOEXTENSIBLE	MAXBYTES
1 ABD02_DATOS	C:\ORACLE\PRODUCT\11.2.0\DBHOME_1\DATABASE\ABD02_DATOS1.DBF	262144	YES	1048576
2 ABD02_INDICES	C:\ORACLE\PRODUCT\11.2.0\DBHOME_1\DATABASE\ABD02_INDICES1.DBF	262144	YES	524288

2.5.Mostrar con un SELECT el espacio libre de los dos tablespaces creados (nombre del tablespace, espacio libre). Justificar el resultado.

Estudiando el espacio libre de la base de datos, nos muestra el espacio libre de cada tablespace. Ha sido cuestión de definir en la consulta que tablespaces son los que queremos mostrar, en nuestro caso, los dos que hemos creado.

```
Select TABLESPACE_NAME, bytes/1024 "Espacio Libre (MB)"  
from dba_free_space  
where ( TABLESPACE_NAME='ABD02_DATOS' OR  
TABLESPACE_NAME='ABD02_INDICES' );
```

TABLESPACE_NAME	Espacio Libre (MB)
1 ABD02_DATOS	192
2 ABD02_INDICES	200

2.6. Asignar por defecto el tablespace ABDXX_DATOS a su usuario ABDXX

Vamos a mirar antes de modificar que tablespace tenemos asignado por defecto, para ello usamos la siguiente consulta:

```
SELECT *
FROM dba_users
WHERE username='ABD02';
```

USERNAME	USER_ID	PASSWORD	ACCOUNT_STATUS	LOCK_DATE	EXPIRY_DATE	DEFAULT_TABLESPACE	TEMPORARY_TABLESPACE
1 ABD02	1690 (null)	OPEN	(null)	(null)	(null)	ABD02_DATOS	ABD02_TEMP

```
ALTER USER ABD02 DEFAULT TABLESPACE ABD02_DATOS;
user ABD02 alterado.
```

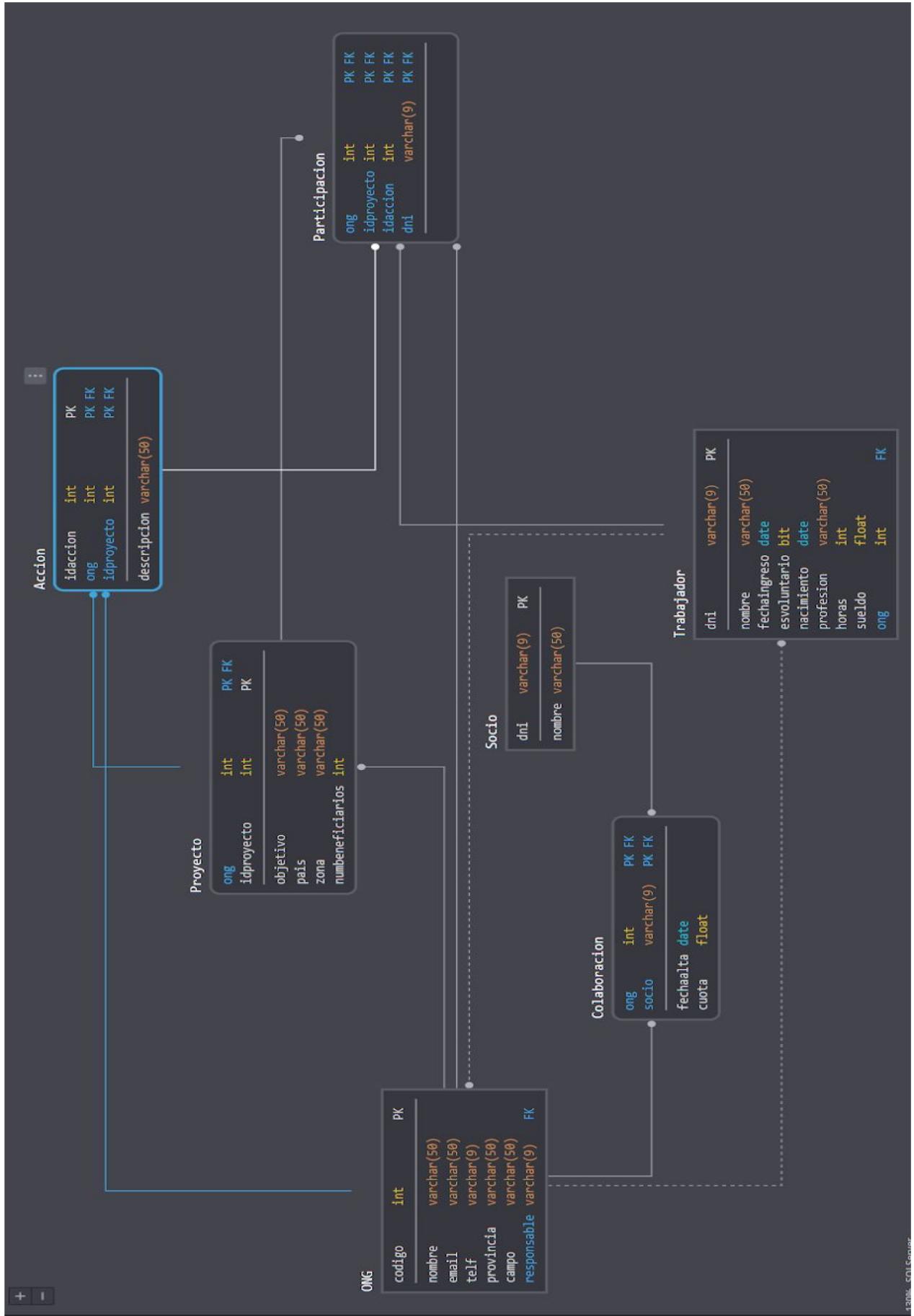
Con este mensaje se confirma que la operación ha sido satisfactoria pero podemos comprobar con la siguiente sentencia que para el usuario 'ABD02' su tablespace por defecto es 'ABD02_DATOS':

```
SELECT *
FROM dba_users
WHERE username='ABD02';
```

USERNAME	USER_ID	PASSWORD	ACCOUNT_STATUS	LOCK_DATE	EXPIRY_DATE	DEFAULT_TABLESPACE	TEMPORARY_TABLESPACE
1 ABD02	1690 (null)	OPEN	(null)	(null)	(null)	ABD02_DATOS	ABD02_TEMP

3. TABLAS

3.1. Dibujar1 y mostrar el esquema de BD del enunciado. Se recomienda dibujar este esquema antes de intentar crear las tablas en la BD, con el fin de tener una visión clara de las relaciones existentes entre ellas y por tanto del orden de creación más adecuado.



3.2. Crear las tablas en el tablespace de datos teniendo en cuenta las siguientes indicaciones:

- ABDXX_SOCIO ha de estar organizada por índice (tabla IOT).
- ABDXX_ONG debe contener 3 particiones por hash según la columna código.
- ABDXX_TRABAJADOR ha de organizarse en 2 particiones según el tipo de trabajador: una llamada CONTRATADO con los trabajadores contratados, y otra llamada VOLUNTARIO con los trabajadores voluntarios.
- ABDXX_PROYECTO, ABDXX_ACCION y ABDXX_PARTICIPACION deben ser almacenadas en un cluster llamado ABDXX_PAP.
- ABDXX_COLABORACION ha de estar particionada por rango de fechas de alta, con 2 particiones, una llamada COLAB_ANTIGUA con los socios dados de alta antes del 2010, y otra llamada COLAB_RECIENTE con los socios dados de alta a partir de 2010. A su vez, cada una de ellas debe contener 2 subparticiones por hash de la columna cuota.

Para la realización de este apartado un problema que tuvimos para la creación del cluster fue que los usuarios del tablespace no tenían permisos para operar sobre el mismo, por lo el usuario propietario (abd02) debe proporcionar permisos y además también de darse quota a sí mismo.

```
ALTER USER abd02
QUOTA UNLIMITED
ON ABD02_DATOS;
```

user ABD02 alterado.

La sentencia ejecutada simplemente modifica los permisos para el tablespace indicado (ABD02_DATOS en este caso) pudiendo utilizar un tamaño ilimitado (QUOTA UNLIMITED). Ahora ya cualquiera de los componentes del grupo de prácticas puede manipular el contenido del tablespace.

Creación de tablas:

--Crear tabla Socio

```
CREATE TABLE ABD02_SOCIO(
    dni      VARCHAR(9)      NOT NULL,
    nombre   VARCHAR(50)      NOT NULL,
    CONSTRAINT socio_pk PRIMARY KEY(dni)
)
ORGANIZATION INDEX
TABLESPACE ABD02_DATOS
```

;

The screenshot shows the Oracle SQL Developer interface. The top window is titled 'Generador de Consultas' and contains the SQL code for creating the 'ABD02_SOCIO' table. The bottom window is titled 'Salida de Script' and displays the message 'Tarea terminada en 0,078 segundos' (Task completed in 0,078 seconds) and the confirmation 'table ABD02_SOCIO creado.' (Table ABD02_SOCIO created.).

```
--Crear tabla Socio
CREATE TABLE ABD02_SOCIO(
    dni      VARCHAR(9)      NOT NULL,
    nombre   VARCHAR(50)      NOT NULL,
    CONSTRAINT socio_pk PRIMARY KEY(dni)
)
ORGANIZATION INDEX
TABLESPACE ABD02_DATOS
```

table ABD02_SOCIO creado.

--Crear tabla ONG

```
CREATE TABLE ABD02_ONG(
    codigo          NUMBER(5)  NOT NULL,
    nombre          VARCHAR(50) NOT NULL,
    email           VARCHAR(50)  NULL,
    telf             VARCHAR(9)  NOT NULL,
    provincia       VARCHAR(50) NOT NULL,
    campo            VARCHAR(50) NOT NULL,
    responsable     VARCHAR(9)  NOT NULL,
    CONSTRAINT ong_pk PRIMARY KEY (codigo),
    CONSTRAINT clave_unique UNIQUE(responsable)
)
PARTITION BY HASH (codigo)
PARTITIONS 3
STORE IN (ABD02_DATOS);
```

The screenshot shows the Oracle SQL Developer interface. In the top menu, 'Herramientas' is selected. The main area displays the following SQL code:

```

ORGANIZATION INDEX
TABLESPACE ABD02_DATOS
;

---Crear tabla ONG
CREATE TABLE ABD02_ONG(
    codigo      NUMBER(5)  NOT NULL,
    nombre      VARCHAR(50) NOT NULL,
    email       VARCHAR(50)  NULL,
    telf        VARCHAR(9)  NOT NULL,
    provincia   VARCHAR(50) NOT NULL,
    campo       VARCHAR(50) NOT NULL,
    responsable  VARCHAR(9) NOT NULL,
    CONSTRAINT ong_pk PRIMARY KEY (codigo),
    CONSTRAINT clave_unique UNIQUE(responsable)
)
PARTITION BY HASH (codigo)
PARTITIONS 3
STORE IN (ABD02_DATOS);

```

--Crear tabla trabajador

```

CREATE TABLE ABD02_TRABAJADOR(
    dni          VARCHAR(9)      NOT NULL,
    nombre       VARCHAR(50)     NOT NULL,
    ong          NUMBER(5)       NOT NULL,
    fechaingreso date           NOT NULL,
    esvoluntario char(1)        NOT NULL,
    nacimiento   date           NOT NULL,
    profesion    VARCHAR(50)     NOT NULL,
    horas        number          NOT NULL,
    sueldo       number          DEFAULT 0 NOT NULL,
    Constraint trabajador_pk PRIMARY KEY (dni),
    Constraint trabajador_esvoluntario CHECK (esvoluntario
IN('S', 'N')),
    Constraint trabajador_sueldo CHECK (sueldo>200),
    Constraint trabajador_horas    CHECK (horas>0 and
horas<=50),
    Constraint trabajador_fk_ong FOREIGN KEY (ong) REFERENCES
ABD02_ONG(codigo)
)
PARTITION BY LIST (esvoluntario)
(
    PARTITION CONTRATADO VALUES ('N')

```

```

        TABLESPACE ABD02_DATOS,
PARTITION VOLUNTARIO VALUES ('S')
        TABLESPACE ABD02_DATOS,
PARTITION DESCONOCIDO VALUES(DEFAULT)

);

```

Hoja de Trabajo Generador de Consultas

```

31  CREATE TABLE ABD02_TRABAJADOR(
32      dni      VARCHAR(9) NOT NULL,
33      nombre   VARCHAR(50) NOT NULL,
34      ong      NUMBER(5) NOT NULL,
35      fechaingreso date    NOT NULL,
36      esvoluntario char(1) NOT NULL,
37      nacimiento date    NOT NULL,
38      profesion VARCHAR(50) NOT NULL,
39      horas     number   NOT NULL,
40      sueldo    number   DEFAULT 0 NOT NULL,
41      Constraint trabajador_pk PRIMARY KEY (dni),
42      Constraint trabajador_esvoluntario CHECK (esvoluntario IN('S', 'N')),
43      Constraint trabajador_sueldo CHECK (sueldo>200),
44      Constraint trabajador_horas CHECK (horas>0 and horas<=50),
45      Constraint trabajador_fk_ong FOREIGN KEY (ong) REFERENCES ABD02_ONG(codigo)
46  )
47  PARTITION BY LIST (esvoluntario)
48  (
49      PARTITION CONTRATADO VALUES ('N')
50          TABLESPACE ABD02_DATOS,
51      PARTITION VOLUNTARIO VALUES ('S')
52          TABLESPACE ABD02_DATOS,
53      PARTITION DESCONOCIDO VALUES(DEFAULT)
54

```

La restricción que puede verse a continuación, es referente a la tabla ong, pero en su momento no podía insertarse porque no existía la tabla trabajador, a la cual referencia dicha restricción y por lo tanto, no hubiera sido posible. Por eso una vez creada la tabla trabajador se inserta la restricción debidamente.

```

ALTER TABLE ABD02_ONG
ADD Constraint ong_fk_trabajador FOREIGN KEY (responsable)
REFERENCES ABD02_TRABAJADOR(dni);

```

The screenshot shows the Oracle SQL Developer interface. On the left, there is a tree view of database objects including Logos de Vistas Materializadas, Símbolos, Símbolos Públicos, Enlaces de Base de Datos, Directorios, Ediciones, Application Express, Java, Esquemas XML, Repositorio de Base de Datos XML, Papelera de Reciclaje, and Trabajador de Firma. The main area contains the following code:

```

56
57  ALTER TABLE ABD02_ONG
58  ADD Constraint ong_fk_trabajador FOREIGN KEY (responsable)
59      REFERENCES ABD02_TRABAJADOR(dni);
60
61
62

```

Below the code, a status bar indicates "Tarea terminada en 0,047 segundos". The bottom pane shows the output of the script execution:

```

Salida de Script x
table ABD02_TRABAJADOR creado.
table ABD02_ONG alterado.

```

--Crear Cluster e índice.

```
create cluster ABD02_PAP (ong number(5), idproyecto number(5))
TABLESPACE ABD02_DATOS;
```

```
CREATE INDEX id02_pap ON CLUSTER ABD02_PAP;
```

--Crear proyecto

```
CREATE TABLE ABD02_PROYECTO(
    ong          NUMBER(5)  NOT NULL,
    idproyecto   NUMBER(5)  NOT NULL,
    objetivo      VARCHAR(50) NOT NULL,
    pais          VARCHAR(50) NOT NULL,
    zona          VARCHAR(50) NOT NULL,
    numbeneficiarios NUMBER,
    Constraint proyecto_pk primary key (ong, idproyecto),
    Constraint proyecto_fk_ong FOREIGN KEY (ong) REFERENCES
ABD02_ONG(codigo)
)
CLUSTER ABD02_PAP(ong, idproyecto);
```

The screenshot shows the Oracle SQL Developer interface with a script editor window. The code in the editor is identical to the one above. Below the editor, a message indicates the task was completed in 0,094 seconds. The status bar at the bottom shows 'table ABD02_PROYECTO creado.'

```
68 --Crear proyecto
69
70 CREATE TABLE ABD02_PROYECTO(
71     ong          NUMBER(5)  NOT NULL,
72     idproyecto   NUMBER(5)  NOT NULL,
73     objetivo      VARCHAR(50) NOT NULL,
74     pais          VARCHAR(50) NOT NULL,
75     zona          VARCHAR(50) NOT NULL,
76     numbeneficiarios NUMBER,
77     Constraint proyecto_pk primary key (ong, idproyecto),
78     Constraint proyecto_fk_ong FOREIGN KEY (ong) REFERENCES ABD02_ONG(codigo)
79 )
80 CLUSTER ABD02_PAP(ong, idproyecto);
81
```

--Crear acción.

```
CREATE TABLE ABD02_ACCION(
    ong          NUMBER(5)  NOT NULL,
    idproyecto   NUMBER(5)  NOT NULL,
```

```

        idaccion      NUMBER(5) NOT NULL,
        descripcion   VARCHAR(50),
        Constraint accion_pk primary key (ong, id proyecto,
idaccion),
        Constraint accion_fk_proyecto FOREIGN KEY (ong,
id proyecto) REFERENCES ABD02_PROYECTO(ong, id proyecto)
)
CLUSTER ABD02_PAP (ong, id proyecto);

```

```

78 |     Constraint proyecto_fk_ong FOREIGN KEY (ong) REFERENCES ABD02_ONG(codigo)
79 | )
80 |     CLUSTER ABD02_PAP(ong, idproyecto);
81 |
82 |
83 --Crear acciÃ³n.
84
85 ┌ CREATE TABLE ABD02_ACCION(
86 |     ong      NUMBER(5) NOT NULL,
87 |     idproyecto NUMBER(5) NOT NULL,
88 |     idaccion NUMBER(5) NOT NULL,
89 |     descripcion VARCHAR(50),
90 |     Constraint accion_pk primary key (ong, idproyecto, idaccion),
91 |     Constraint accion_fk_proyecto FOREIGN KEY (ong, idproyecto) REFERENCES ABD02_PROYECTO(ong, idproyecto)
92 | )
93 |     CLUSTER ABD02_PAP (ong, idproyecto);
94

```

Salida de Script

Tarea terminada en 0,047 segundos

--Crear participaciÃ³n.

```

CREATE TABLE ABD02_PARTICIPACION(
    ong      NUMBER(5) NOT NULL,
    idproyecto NUMBER(5) NOT NULL,
    idaccion NUMBER(5) NOT NULL,
    trabajador VARCHAR(9)      NOT NULL,
    Constraint participacion_pk primary key (ong, idproyecto,
idaccion, trabajador),
    Constraint participacion_fk_accion FOREIGN KEY (ong,
idproyecto, idaccion) REFERENCES ABD02_ACCION(ong, idproyecto,
idaccion),
    Constraint trabajador_fk FOREIGN KEY (trabajador)
REFERENCES ABD02_TRABAJADOR(dni)

)
CLUSTER ABD02_PAP (ong, idproyecto);

```

```

de Trabajo Generador de Consultas
CLUSTER ABD02_PAP (ong, idproyecto);

--Crear participaciÃ³n.

CREATE TABLE ABD02_PARTICIPACION(
    ong      NUMBER(5)  NOT NULL,
    idproyecto  NUMBER(5)  NOT NULL,
    idaccion  NUMBER(5)  NOT NULL,
    trabajador  VARCHAR(9)  NOT NULL,
    Constraint participacion_pk primary key (ong, idproyecto, idaccion, trabajador),
    Constraint participacion_fk_accion FOREIGN KEY (ong, idproyecto, idaccion) REFERENCES ABD02_ACCION(ong, idproyecto, idaccion)
    Constraint trabajador_fk FOREIGN KEY (trabajador) REFERENCES ABD02_TRABAJADOR(dni)

CLUSTER ABD02_PAP (ong, idproyecto);

Alta de Script X
Tarea terminada en 0,047 segundos
La ABD02_PARTICIPACION creada.

```

--Crear Colaboración.

```

CREATE TABLE ABD02_COLABORACION(
    ong          NUMBER(5)  NOT NULL,
    socio        VARCHAR(9)  NOT NULL,
    fechaalta    DATE      NOT NULL,
    cuota        NUMBER      DEFAULT 0 NULL,
    Constraint colaboracion_pk primary key (ong, socio),
    Constraint colaboracion_fk_ong  FOREIGN KEY (ong)
REFERENCES ABD02_ONG(codigo),
    Constraint colaboracion_fk_socio  FOREIGN KEY (socio)
REFERENCES ABD02_SOCIO(dni),
    Constraint colaboracion_cuota CHECK (cuota>0)
)
PARTITION BY RANGE(fechaalta)
    SUBPARTITION BY HASH (cuota)
    SUBPARTITIONS 2 store in (ABD02_DATOS)
(
    PARTITION COLAB_ANTIGUA
        VALUES LESS THAN (TO_DATE('01/01/2010'
,'dd/mm/yyyy'))
        TABLESPACE ABD02_DATOS,
    PARTITION COLAB_RECIENTE
        VALUES LESS THAN (MAXVALUE)
        TABLESPACE ABD02_DATOS
);

```

de Trabajo Generador de Consultas 0,18700001 segundos

```

CREATE TABLE ABD02_COLABORACION(
    ong      NUMBER(5)  NOT NULL,
    socio    VARCHAR(9) NOT NULL,
    fechalta DATE      NOT NULL,
    cuota    NUMBER     DEFAULT 0 NOT NULL,
    Constraint colaboracion_pk primary key (ong, socio),
    Constraint colaboracion_fk_ong FOREIGN KEY (ong) REFERENCES ABD02_ONG(codigo),
    Constraint colaboracion_fk_socio FOREIGN KEY (socio) REFERENCES ABD02_SOCIO(dni),
    Constraint colaboracion_cuota CHECK (cuota>0)
)
PARTITION BY RANGE(fechalta)
SUBPARTITION BY HASH (cuota)
    SUBPARTITIONS 2 store in (ABD02_DATOS)

    PARTITION COLAB_ANTIGUA
        VALUES LESS THAN (TO_DATE('01/01/2010', 'dd/mm/yyyy'))
        TABLESPACE ABD02_DATOS,
    PARTITION COLAB_RECIENTE
        VALUES LESS THAN (MAXVALUE)
        TABLESPACE ABD02_DATOS
);

```

Salida de Script Tarea terminada en 0,187 segundos

.e ABD02_PARTICIPACION creado.
.e ABD02_COLABORACION creado.

Oracle SQL Developer : orcl11

Archivo Editar Ver Navegar Ejecutar Control de Versiones Herramientas Ayuda

Conexiones X Informes X Generador de Consultas 0,029 segundos

```

CREATE TABLE ABD02_COLABORACION(
    ong      NUMBER(5)  NOT NULL,
    socio    VARCHAR(9) NOT NULL,
    fechalta DATE      NOT NULL,
    cuota    NUMBER     DEFAULT 0 NOT NULL,
    Constraint colaboracion_pk primary key (ong, socio),
    FOREIGN KEY (ong) REFERENCES ABD02_ONG(codigo),
    FOREIGN KEY (socio) REFERENCES ABD02_SOCIO(dni)
)
PARTITION BY RANGE(fechalta)
SUBPARTITION BY HASH (cuota)
    SUBPARTITIONS 2 store in (ABD02_DATOS)

    PARTITION COLAB_ANTIGUA
        VALUES LESS THAN (TO_DATE('01/01/2010', 'dd/mm/yyyy'))
        TABLESPACE ABD02_DATOS,
    PARTITION COLAB_RECIENTE
        VALUES LESS THAN (MAXVALUE)
        TABLESPACE ABD02_DATOS
);

```

Salida de Script Resultado de la Consulta Tarea terminada en 0,039 segundos

```

PARTITION BY RANGE(fechalta)
SUBPARTITION BY HASH (cuota)
    SUBPARTITIONS 2 store in (ABD02_DATOS)

    PARTITION COLAB_ANTIGUA
        VALUES LESS THAN (TO_DATE('01/01/2010', 'dd/mm/yyyy'))
        TABLESPACE ABD02_DATOS,
    PARTITION COLAB_RECIENTE
        VALUES LESS THAN (MAXVALUE)
        TABLESPACE ABD02_DATOS
);

Error en la linea 14 del comando:47 columnario
Indice de error:
Error SQL: ORA-00942: falta la opcion o no es valida
00942. 00000 -  "missing or invalid option"
*Cause:
*Action:
Tarea finalizada
table ABD02_COLABORACION creada.

```

Messages-Log X 2010-04-18 16:00:57 - Load Controllers
2010-04-18 16:01:04 - Init Recently opened Designs

Jueves 30 Columnas | Insertar | Modificado | Windows: C:\AF\ORACLE\ORACLE\ORCL11\BDG1109-p2-02.sql | 18:40 | 18/04/2010

```

CREATE TABLE ABD02_FATOS (
    PRIMER_ID NUMBER(10),
    FOREIGN KEY (PRIMER_ID) REFERENCES ABD02_OBQ(codigoo),
    FOREIGN KEY (secundo) REFERENCES ABD02_OCCID(dni)
)
PARTITION BY RANGE (codigoo)
SUBPARTITION BY HASH (coursa)
SUBPARTITIONS 2 store in (ABD02_DATOS)
PARTITION COAH_AMERICA
VALUES LESS THAN (TO_DATE('01/01/2010', 'dd/mm/yyyy'))
TABLESPACE ABD02_DATOS,
PARTITION COAH_RECIENTE
VALUES LESS THAN (MAXVALUE)
TABLESPACE ABD02_DATOS
);

ALTER TABLE ABD02_OBQ
ADD FOREIGN KEY (responsable)
REFERENCES ABD02_TRABAJADOR(dni);

```

Salida de Script > Resultado de la Consulta >

Subparticiones por HASH (coursa)

Error en la linea de comando:47 Columna:10

Indice de error:

Error SQL: ORA-00922: falta la opcion o no es valida

00922. 00000 - "missing or invalid option"

*Cause:

*Action:

table ABD02_COLEGACION creado.

table ABD02_OBQ alterada.

Mensajes-log A

2018-04-18 16:00:57 - Load Controllers

2018-04-18 16:01:06 - Init Recently opened Designs

3.3. Mostrar con un SELECT los tablespaces donde se ha creado todas las tablas, particiones y subparticiones (nombre de la tabla, nombre de la partición, nombre de la subpartición, y nombre del tablespace de la tabla o partición o subpartición).

La información relativa a las particiones y subparticiones, se encuentra en dba_tab_partitions y en dba_tab_subpartitions.

```

SELECT      table_name,      p.partition_name,      sp.subpartition_name,
sp.tablespace_name
FROM user_tables t
LEFT JOIN dba_tab_partitions p
LEFT JOIN dba_tab_subpartitions sp
      ON      ( (p.partition_name = sp.partition_name)      and
(sp.table_owner = 'ABD02' ) )
      ON t.table_name = p.table_name;

```

TABLE_NAME	PARTITION_NAME	SUBPARTITION_NAME	TABLESPACE_NAME
1 ABD02_PROYECTO	(null)	(null)	(null)
2 ABD02_ACCION	(null)	(null)	(null)
3 ABD02_PARTICIPACION	(null)	(null)	(null)
4 ABD02_SOCIO	(null)	(null)	(null)
5 ABD02_ONG	SYS_P2626	(null)	(null)
6 ABD02_ONG	SYS_P2625	(null)	(null)
7 ABD02_ONG	SYS_P2624	(null)	(null)
8 ABD02_TRABAJADOR	VOLUNTARIO	(null)	(null)
9 ABD02_TRABAJADOR	DESCONOCIDO	(null)	(null)
10 ABD02_TRABAJADOR	CONTRATADO	(null)	(null)
11 ABD02_COLABORACION	COLAB_RECIENTE	SYS_SUBP2646	ABD02_DATOS
12 ABD02_COLABORACION	COLAB_RECIENTE	SYS_SUBP2645	ABD02_DATOS
13 ABD02_COLABORACION	COLAB_ANTIGUA	SYS_SUBP2644	ABD02_DATOS
14 ABD02_COLABORACION	COLAB_ANTIGUA	SYS_SUBP2643	ABD02_DATOS

3.4. Mostrar con un SELECT las restricciones de integridad de clave primaria de las tablas creadas (nombre de la tabla, nombre de la clave primaria, columna que forma parte de la clave primaria).

```
SELECT cc.constraint_name , cc.table_name , cc.column_name
FROM user_cons_columns CC, user_constraints CO
WHERE constraint_type='P';
```

CONSTRAINT_NAME	TABLE_NAME	COLUMN_NAME
172 PARTICIPACION_PK	ABD02_PARTICIPACION	IDPROYECTO
173 PARTICIPACION_PK	ABD02_PARTICIPACION	ONG
174 PARTICIPACION_FK_ACCION	ABD02_PARTICIPACION	IDACCION
175 PARTICIPACION_FK_ACCION	ABD02_PARTICIPACION	IDPROYECTO
176 PARTICIPACION_FK_ACCION	ABD02_PARTICIPACION	ONG
177 ONG_PK	ABD02_ONG	CODIGO
178 ONG_FK_TRABAJADOR	ABD02_ONG	RESPONSABLE
179 COLABORACION_PK	ABD02_COLABORACION	SOCIO
180 COLABORACION_PK	ABD02_COLABORACION	ONG
181 COLABORACION_FK_SOCIO	ABD02_COLABORACION	SOCIO
182 COLABORACION_FK_ONG	ABD02_COLABORACION	ONG
183 COLABORACION CUOTA	ABD02_COLABORACION	CUOTA
184 CLAVE_UNIQUE	ABD02_ONG	RESPONSABLE
185 ACCION_PK	ABD02_ACCION	IDACCION
186 ACCION_PK	ABD02_ACCION	IDPROYECTO
187 ACCION_PK	ABD02_ACCION	ONG
188 ACCION_FK_PROYECTO	ABD02_ACCION	IDPROYECTO
189 ACCION_FK_PROYECTO	ABD02_ACCION	ONG
190 TRABAJADOR_SUELDO	ABD02_TRABAJADOR	SUELDO
191 TRABAJADOR_PK	ABD02_TRABAJADOR	DNI
192 TRABAJADOR_HORAS	ABD02_TRABAJADOR	HORAS
193 TRABAJADOR_FK_ONG	ABD02_TRABAJADOR	ONG
194 TRABAJADOR_FK	ABD02_PARTICIPACION	TRABAJADOR

3.5. Mostrar con un SELECT las restricciones de integridad de clave ajena (externa) de las tablas creadas (nombre de la tabla, nombre de la clave ajena, columna que forma parte de la clave ajena, y nombre de la clave primaria o clave única a la que apunta).

```
SELECT cc.constraint_name , cc.table_name , cc.column_name
FROM user_cons_columns CC, user_constraints CO
WHERE constraint_type='R'
```

CONSTRAINT_NAME	TABLE_NAME	COLUMN_NAME
108 PARTICIPACION_PK	ABD02_PARTICIPACION	IDACCION
109 PARTICIPACION_PK	ABD02_PARTICIPACION	IDPROYECTO
110 PARTICIPACION_PK	ABD02_PARTICIPACION	ONG
111 PARTICIPACION_FK_ACCION	ABD02_PARTICIPACION	IDACCION
112 PARTICIPACION_FK_ACCION	ABD02_PARTICIPACION	IDPROYECTO
113 PARTICIPACION_FK_ACCION	ABD02_PARTICIPACION	ONG
114 ONG_PK	ABD02_ONG	CODIGO
115 ONG_FK_TRABAJADOR	ABD02_ONG	RESPONSABLE
116 COLABORACION_PK	ABD02_COLABORACION	SOCIO
117 COLABORACION_PK	ABD02_COLABORACION	ONG
118 COLABORACION_FK_SOCIO	ABD02_COLABORACION	SOCIO
119 COLABORACION_FK_ONG	ABD02_COLABORACION	ONG
120 COLABORACION CUOTA	ABD02_COLABORACION	CUOTA
121 CLAVE_UNIQUE	ABD02_ONG	RESPONSABLE
122 ACCION_PK	ABD02_ACCION	IDACCION
123 ACCION_PK	ABD02_ACCION	IDPROYECTO
124 ACCION_PK	ABD02_ACCION	ONG
125 ACCION_FK_PROYECTO	ABD02_ACCION	IDPROYECTO
126 ACCION_FK_PROYECTO	ABD02_ACCION	ONG
127 TRABAJADOR_SUELDO	ABD02_TRABAJADOR	SUELDO
128 TRABAJADOR_PK	ABD02_TRABAJADOR	DNI
129 TRABAJADOR_HORAS	ABD02_TRABAJADOR	HORAS
130 TRABAJADOR_FK_ONG	ABD02_TRABAJADOR	ONG

3.6. Mostrar con un SELECT las restricciones de integridad de tipo CHECK de las tablas creadas (nombre de la tabla, nombre de la restricción, y condición de la restricción).

```
SELECT constraint_name , table_name , search_condition
FROM user_constraints
WHERE constraint_type='C';
```

CONSTRAINT_NAME	TABLE_NAME	SEARCH_CONDITION
15 SYS_C00632094	ABD02_TRABAJADOR	"PROFESION" IS NOT NULL
16 SYS_C00632095	ABD02_TRABAJADOR	"HORAS" IS NOT NULL
17 SYS_C00632096	ABD02_TRABAJADOR	"SUELDO" IS NOT NULL
18 TRABAJADOR_ESVOLUNTARIO	ABD02_TRABAJADOR	esvoluntario IN('S', 'N')
19 TRABAJADOR_SUELDO	ABD02_TRABAJADOR	sueldo>200
20 TRABAJADOR_HORAS	ABD02_TRABAJADOR	horas>0 AND horas<=50
21 SYS_C00632088	ABD02_TRABAJADOR	"DNI" IS NOT NULL
22 SYS_C00632103	ABD02_PROYECTO	"ONG" IS NOT NULL
23 SYS_C00632104	ABD02_PROYECTO	"IDPROYECTO" IS NOT NULL
24 SYS_C00632105	ABD02_PROYECTO	"OBJETIVO" IS NOT NULL
25 SYS_C00632106	ABD02_PROYECTO	"PAIS" IS NOT NULL
26 SYS_C00632107	ABD02_PROYECTO	"ZONA" IS NOT NULL
27 SYS_C00632110	ABD02_ACCION	"ONG" IS NOT NULL
28 SYS_C00632111	ABD02_ACCION	"IDPROYECTO" IS NOT NULL
29 SYS_C00632112	ABD02_ACCION	"IDACCION" IS NOT NULL
30 SYS_C00632115	ABD02_PARTICIPACION	"ONG" IS NOT NULL
31 SYS_C00632116	ABD02_PARTICIPACION	"IDPROYECTO" IS NOT NULL
32 SYS_C00632117	ABD02_PARTICIPACION	"IDACCION" IS NOT NULL
33 SYS_C00632118	ABD02_PARTICIPACION	"TRABAJADOR" IS NOT NULL
34 SYS_C00632122	ABD02_COLABORACION	"ONG" IS NOT NULL
35 SYS_C00632123	ABD02_COLABORACION	"SOCIO" IS NOT NULL
36 SYS_C00632124	ABD02_COLABORACION	"FECHAALTA" IS NOT NULL
37 COLABORACION CUOTA	ABD02_COLABORACION	cuota>0

4. INSERCIÓN DE DATOS

4.1.Insertar varias filas en cada una de las tablas de forma que todas las tablas, particiones y subparticiones contengan alguna fila.

En la tabla ABDXX_ONG, la columna código ha de ser un número secuencial obtenido automáticamente a partir de una secuencia.

- Crear la secuencia llamada ABDXX_CO, tal que comenzando por el cero (0), genere identificadores de 2 en 2, hasta un valor máximo de 500 y sin ciclos.

```
CREATE SEQUENCE ABD02_CO
START WITH 0
INCREMENT BY 2
MINVALUE 0
MAXVALUE 500;
```

Cuando vayamos a meter los datos se escribe ABD02_CO.nextVal en el campo que corresponde

Cargar las tablas con datos utilizando sentencias INSERT. Para cada tabla, mostrar:

- Los comandos INSERT.
- Con un SELECT todas las filas insertadas.

4.1.Insertar varias filas en cada una de las tablas de forma que todas las tablas, particiones y subparticiones contengan alguna fila.

En la tabla ABDXX_ONG, la columna código ha de ser un número secuencial obtenido automáticamente a partir de una secuencia.

- Crear la secuencia llamada ABDXX_CO, tal que comenzando por el cero (0), genere identificadores de 2 en 2, hasta un valor máximo de 500 y sin ciclos.

Cargar las tablas con datos utilizando sentencias INSERT. Para cada tabla, mostrar:

- Los comandos INSERT.
- Con un SELECT todas las filas insertadas.

--Crear secuencia.

```
CREATE SEQUENCE ABD02_CO
START WITH 0
INCREMENT BY 2
MINVALUE 0
MAXVALUE 500;
```

--Para utilizar lo anterior ABD02_CO.nextVal

--Datos tabla ONG.

```
ALTER TABLE ABD02_ONG
DISABLE constraint ong_fk_trabajador;

INSERT INTO ABD02_ONG(codigo, nombre, email, telf, provincia, campo,
responsable)
VALUES(ABD02_CO.nextVal, 'Salvemos al
pato', 'pato@hotmail.es', '123456780', 'Albacete', 'medioambiente', '1111
1111A');

INSERT INTO ABD02_ONG(codigo, nombre, email, telf, provincia, campo,
responsable)
VALUES(ABD02_CO.nextVal, 'Salvemos al
panda', 'panda@hotmail.es', '123456781', 'Alicante', 'medioambiente', '22
222222A');

INSERT INTO ABD02_ONG(codigo, nombre, email, telf, provincia, campo,
responsable)
VALUES(ABD02_CO.nextVal, 'Naturaleza', 'natu@hotmail.es', '123456782', 'Cuenca',
'medioambiente', '33333333A');

INSERT INTO ABD02_ONG(codigo, nombre, email, telf, provincia, campo,
responsable)
VALUES(ABD02_CO.nextVal, 'Vivienda', 'vivienda@hotmail.es', '123456783',
'Lugo', 'inmigracion', '44444444A');

INSERT INTO ABD02_ONG(codigo, nombre, email, telf, provincia, campo,
responsable)
VALUES(ABD02_CO.nextVal, 'Lucha
Humana', 'luhu@hotmail.es', '123456784', 'Pontevedra', 'derechos
humanos', '55555555A');

INSERT INTO ABD02_ONG(codigo, nombre, email, telf, provincia, campo,
responsable)
VALUES(ABD02_CO.nextVal, 'SosHumano', 'Sos@hotmail.es', '123456785', 'Gr
anada', 'derechos humanos', '66666666A');
```

```
INSERT INTO ABD02_ONG(codigo, nombre, email, telf, provincia, campo, responsable)
```

```
VALUES(ABD02_CO.nextVal, 'Dehuma', 'Dehuma@hotmail.es', '123456786', 'Malaga', 'inmigracion', '77777777A');
```

```
INSERT INTO ABD02_ONG(codigo, nombre, email, telf, provincia, campo, responsable)
```

```
VALUES(ABD02_CO.nextVal, 'SegurAlimer', 'SegurAlimer@hotmail.es', '123456787', 'Cadiz', 'Nutricion y Seguridad Alimentaria', '88888888A');
```

```
COMMIT
```

CODIGO	NOMBRE	EMAIL	TELF	PROVINCIA	CAMPO	RESPONSABLE
1	6 Vivienda	vivienda@hotmail.es	123456783	Lugo	inmigracion	44444444A
2	0 Salvemos al pato	pato@hotmail.es	123456780	Albacete	medioambiente	11111111A
3	4 Naturaleza	natu@hotmail.es	123456782	Cuenca	medioambiente	33333333A
4	10 SosHumanos	Sos@hotmail.es	123456785	Granada	derechos humanos	66666666A
5	12 Dehuma	Dehuma@hotmail.es	123456786	Malaga	inmigracion	77777777A
6	14 SegurAlimer	SegurAlimer@hotmail.es	123456787	Cadiz	Nutricion y Seguridad Alimentaria	88888888A
7	2 Salvemos al panda	panda@hotmail.es	123456781	Alicante	medioambiente	22222222A
8	8 Lucha Humana	luhu@hotmail.es	968454545	Pontevedra	derechos humanos	55555555A
9	44 Copaelve	Copaelve@hotmail.es	125556787	Murcia	COOPERACIÓN PARA EL DESARROLLO	14444444A

--Datos tabla Trabajador.

```
INSERT INTO ABD02_TRABAJADOR(dni, nombre, ong, fechaingreso, esvoluntario, nacimiento, profesion, horas, sueldo)
```

```
VALUES('11111111A', 'Marcos', 0, '7/2/2014', 'N', '25/1/1980', 'Empresario', 37, 950);
```

```
INSERT INTO ABD02_TRABAJADOR(dni, nombre, ong, fechaingreso, esvoluntario, nacimiento, profesion, horas, sueldo)
```

```
VALUES('22222222A', 'Elena', 2, '7/3/2015', 'N', '27/2/1978', 'Vendedora', 40, 1000);
```

```
INSERT INTO ABD02_TRABAJADOR(dni, nombre, ong, fechaingreso, esvoluntario, nacimiento, profesion, horas, sueldo)
```

```
VALUES('33333333A','Paco',4,'8/2/2012','N','05/02/1977','Locutor',41,1230);

INSERT INTO ABD02_TRABAJADOR(dni, nombre, ong, fechaingreso, esvoluntario, nacimiento, profesion, horas, sueldo)

VALUES('44444444A','Antonio',6,'1/2/2014','N','07/1/1980','Periodista',39,1020);

INSERT INTO ABD02_TRABAJADOR(dni, nombre, ong, fechaingreso, esvoluntario, nacimiento, profesion, horas, sueldo)

VALUES('66666666A','Concha',8,'7/5/2011','N','06/1/1950','Zapatera',48,1000);

INSERT INTO ABD02_TRABAJADOR(dni, nombre, ong, fechaingreso, esvoluntario, nacimiento, profesion, horas, sueldo)

VALUES('55555555A','Ana',10,'7/4/2010','N','25/1/1985','Conductora',50,700);

INSERT INTO ABD02_TRABAJADOR(dni, nombre, ong, fechaingreso, esvoluntario, nacimiento, profesion, horas, sueldo)

VALUES('77777777A','Lourdes',12,'7/5/2014','N','20/09/1970','Limpadora',41,750);

INSERT INTO ABD02_TRABAJADOR(dni, nombre, ong, fechaingreso, esvoluntario, nacimiento, profesion, horas, sueldo)

VALUES('88888888A','Lucia',14,'7/2/2014','N','09/5/1980','Profesora',35,850);

INSERT INTO ABD02_TRABAJADOR(dni, nombre, ong, fechaingreso, esvoluntario, nacimiento, profesion, horas, sueldo)

VALUES('7777766A','AS',0,'7/2/2014','S','25/1/1980','Empresario',37,950);

INSERT INTO ABD02_TRABAJADOR(dni, nombre, ong, fechaingreso, esvoluntario, nacimiento, profesion, horas, sueldo)
```

```
VALUES('88888889A','EN',10,'7/2/2014','N','25/1/1980','Empresario',3  
7,950);  
  
INSERT INTO ABD02_TRABAJADOR(dni, nombre, ong, fechaingreso,  
esvoluntario, nacimiento, profesion, horas, sueldo)  
  
VALUES('58799999E','TS',10,'7/2/2014','S','25/1/1980','Empresario',3  
7,950);  
  
INSERT INTO ABD02_TRABAJADOR(dni, nombre, ong, fechaingreso,  
esvoluntario, nacimiento, profesion, horas, sueldo)  
  
VALUES('10101010A','Eusebio',0,'19/4/2012','N','10/4/1975','Escritor  
a',32,900);  
  
INSERT INTO ABD02_TRABAJADOR(dni, nombre, ong, fechaingreso,  
esvoluntario, nacimiento, profesion, horas, sueldo)  
  
VALUES('12222222A','Marta',2,'17/5/2012','N','10/4/1975','Doctora',4  
0,900);  
  
INSERT INTO ABD02_TRABAJADOR(dni, nombre, ong, fechaingreso,  
esvoluntario, nacimiento, profesion, horas, sueldo)  
  
VALUES('13333333A','Lazaro',2,'14/5/2015','N','10/4/1975','Pintor',3  
8,1200);  
  
INSERT INTO ABD02_TRABAJADOR(dni, nombre, ong, fechaingreso,  
esvoluntario, nacimiento, profesion, horas, sueldo)  
  
VALUES('14444444A','Luisa',2,'11/5/1965','N','10/4/1975','Fontanera'  
,35,1000);  
  
INSERT INTO ABD02_TRABAJADOR(dni, nombre, ong, fechaingreso,  
esvoluntario, nacimiento, profesion, horas, sueldo)  
  
VALUES('12121212A','Mariano',4,'11/5/1973','N','10/4/1975','Pintor',  
38,1050);
```

```
ALTER TABLE ABD02_ONG
```

```
ENABLE CONSTRAINT ong_fk_trabajador;
```

```
COMMIT
```

DNI	NOMBRE	ONG	FECHAINGRESO	ESVOLUNTARIO	NACIMIENTO	PROFESION	HORAS	SUELDO
1 11111111A	Marcos	007/02/14	N	25/01/80	Empresario	37	950	
2 22222222A	Elena	207/03/15	N	27/02/78	Vendedora	40	1000	
3 33333333A	Paco	408/02/12	N	05/02/77	Locutor	41	1230	
4 44444444A	Antonio	601/02/14	N	07/01/80	Periodista	39	1020	
5 66666666A	Concha	807/05/11	N	06/01/50	Zapatera	48	1000	
6 55555555A	Ana	1007/04/10	N	25/01/85	Conductora	50	700	
7 77777777A	Lourdes	1207/05/14	N	20/09/70	Limpiadora	41	750	
8 88888888A	Lucia	1407/02/14	N	09/05/80	Profesora	35	850	
9 88888889A	EN	1007/02/14	N	25/01/80	Empresario	37	950	
10 10101010A	Eusebio	019/04/12	N	10/04/75	Escritora	32	900	
11 12222222A	Marta	217/05/12	N	10/04/75	Doctora	40	900	
12 13333333A	Lazaro	214/05/15	N	10/04/75	Pintor	38	1200	
13 14444444A	Luisa	211/05/65	N	10/04/75	Fontanera	35	1000	
14 12121212A	Mariano	411/05/73	N	10/04/75	Pintor	38	1050	
15 7777766A	AS	007/02/14	S	25/01/80	Empresario	37	950	
16 58799999E	TS	1007/02/14	S	25/01/80	Empresario	37	950	

--Datos tabla Proyecto.

```
INSERT INTO ABD02_PROYECTO(ong, idproyecto, objetivo, pais, zona, numbeneficiarios)
VALUES(0,101,'Informar sobre el pato','ESPAÃ'A','Centro',25);
```

```
INSERT INTO ABD02_PROYECTO(ong, idproyecto, objetivo, pais, zona, numbeneficiarios)
VALUES(0,102,'Recuperar humedales','ESPAÃ'A','a',78);
```

```
INSERT INTO ABD02_PROYECTO(ong, idproyecto, objetivo, pais, zona, numbeneficiarios)
VALUES(2,201,'Informacion sobre el panda','ESPAÃ'A','OESTE',80);
```

```
INSERT INTO ABD02_PROYECTO(ong, idproyecto, objetivo, pais, zona, numbeneficiarios)
VALUES(2,202,'Cultivo Bambu','ESPAÃ'A','Sur',70);
```

```
INSERT INTO ABD02_PROYECTO(ong, idproyecto, objetivo, pais, zona,
numbeneficiarios)
VALUES(2,203,'Recuperacion
especies','ESPAÃ‘A','Central',65);

INSERT INTO ABD02_PROYECTO(ong, idproyecto, objetivo, pais, zona,
numbeneficiarios)
VALUES(4,301,'Salvar bosques','ESPAÃ‘A','Norte',85);

INSERT INTO ABD02_PROYECTO(ong, idproyecto, objetivo, pais, zona,
numbeneficiarios)
VALUES(6,401,'Ayuda a familias','ESPAÃ‘A','Sur',45);

INSERT INTO ABD02_PROYECTO(ong, idproyecto, objetivo, pais, zona,
numbeneficiarios)
VALUES(6,402,'Viviendas sociales','ESPAÃ‘A','b',50);

INSERT INTO ABD02_PROYECTO(ong, idproyecto, objetivo, pais, zona,
numbeneficiarios)
VALUES(8,501,'Ayuda personas
problemas','ESPAÃ‘A','extrarradio',78);

INSERT INTO ABD02_PROYECTO(ong, idproyecto, objetivo, pais, zona,
numbeneficiarios)
VALUES(10,601,'Reparto comida','ElSalvador','centro',98);

INSERT INTO ABD02_PROYECTO(ong, idproyecto, objetivo, pais, zona,
numbeneficiarios)
VALUES(12,701,'Ayuda busqueda hogar','Guinea','Sin
determinar',99);

INSERT INTO ABD02_PROYECTO(ong, idproyecto, objetivo, pais, zona,
numbeneficiarios)
```

```

VALUES(14,801,'Seguridad Alimentos infantiles','ESPAÑA','Interior',38);

```

```

INSERT INTO ABD02_PROYECTO(ong, idproyecto, objetivo, pais, zona,
numbeneficiarios)

```

```

VALUES(14,802,'Vigilancia Alimentos 1ª necesidad','Senegal','Centro',78);

```

```

COMMIT

```

	ONG	IDPROYECTO	OBJETIVO	PAIS	ZONA	NUMBENEFICIARIOS
1	14	802	Vigilancia Alimentos 1ª necesidad	Senegal	Centro	78
2	0	101	Informar sobre el pato	ESPAÑA	Centro	25
3	0	102	Recuperar humedales	ESPAÑA	a	78
4	2	201	Informacion sobre el panda	ESPAÑA	OESTE	80
5	2	202	Cultivo Bambu	ESPAÑA	Sur	70
6	2	203	Recuperacion especies	ESPAÑA	Central	65
7	4	301	Salvar bosques	ESPAÑA	Norte	85
8	6	401	Ayuda a familias	ESPAÑA	Sur	45
9	6	402	Viviendas sociales	ESPAÑA	b	50
10	8	501	Ayuda personas problemas	ESPAÑA	extrarradio	78
11	10	601	Reparto comida	ElSalvador	centro	98
12	12	701	Ayuda busqueda hogar	Guinea	Sin determinar	99
13	14	801	Seguridad Alimentos infantiles	ESPAÑA	Interior	38
14	44	901	Escolarizacion infantil	ETIOPIA	Centro	100

--Datos tabla accion.

```

INSERT INTO ABD02_ACCION(ong, idproyecto, idaccion, descripcion)
VALUES(0,101,1101,'Informacion en colegios');

```

```

INSERT INTO ABD02_ACCION(ong, idproyecto, idaccion, descripcion)
VALUES(0,101,1102,'Informar en la calle');

```

```

INSERT INTO ABD02_ACCION(ong, idproyecto, idaccion, descripcion)
VALUES(0,102,1201,'Concienciacion importancia humedales');

```

```

INSERT INTO ABD02_ACCION(ong, idproyecto, idaccion, descripcion)
VALUES(0,102,1202,'Trabajo en humedales');

```

```

INSERT INTO ABD02_ACCION(ong, idproyecto, idaccion, descripcion)
VALUES(2,201,2101,'Informacion en calles');

```

```

INSERT INTO ABD02_ACCION(ong, idproyecto, idaccion, descripcion)
VALUES(2,202,2202,'Busqueda financiacion');

```

```
INSERT INTO ABD02_ACCION(ong, idproyecto, idaccion, descripcion)
VALUES(2,202,2203,'Accion sobre el terreno');
```

```
INSERT INTO ABD02_ACCION(ong, idproyecto, idaccion, descripcion)
VALUES(2,203,2301,'Comenzar por especies mas perjudicadas');
```

```
INSERT INTO ABD02_ACCION(ong, idproyecto, idaccion, descripcion)
VALUES(10,601,2601,'Reparto comida a personas en apuros');
```

```
INSERT INTO ABD02_ACCION(ong, idproyecto, idaccion, descripcion)
VALUES(4,301,3301,'Recaudacion dinero');
```

```
INSERT INTO ABD02_ACCION(ong, idproyecto, idaccion, descripcion)
VALUES(4,301,3302,'Busqueda personas');
```

```
INSERT INTO ABD02_ACCION(ong, idproyecto, idaccion, descripcion)
VALUES(4,301,3303,'Reforestacion');
```

```
INSERT INTO ABD02_ACCION(ong, idproyecto, idaccion, descripcion)
VALUES(6,401,4101,'Buscar familias');
```

```
INSERT INTO ABD02_ACCION(ong, idproyecto, idaccion, descripcion)
VALUES(6,401,4102,'Conocer familias');
```

```
INSERT INTO ABD02_ACCION(ong, idproyecto, idaccion, descripcion)
VALUES(6,402,4201,'Preparacion Viviendas');
```

```
INSERT INTO ABD02_ACCION(ong, idproyecto, idaccion, descripcion)
VALUES(6,402,4202,'Reparto viviendas');
```

```
INSERT INTO ABD02_ACCION(ong, id proyecto, idaccion, descripcion)
VALUES(8,501,5101,'Conocer personas');

INSERT INTO ABD02_ACCION(ong, id proyecto, idaccion, descripcion)
VALUES(8,501,5102,'Integracion');

INSERT INTO ABD02_ACCION(ong, id proyecto, idaccion, descripcion)
VALUES(12,701,7101,'Estudio recursos');

INSERT INTO ABD02_ACCION(ong, id proyecto, idaccion, descripcion)
VALUES(12,701,7102,'Busqueda casa adecuada');

INSERT INTO ABD02_ACCION(ong, id proyecto, idaccion, descripcion)
VALUES(14,801,8101,'Test supermercados');

INSERT INTO ABD02_ACCION(ong, id proyecto, idaccion, descripcion)
VALUES(14,801,8102,'Test empresas');

INSERT INTO ABD02_ACCION(ong, id proyecto, idaccion, descripcion)
VALUES(14,802,8201,'Reparto');

INSERT INTO ABD02_ACCION(ong, id proyecto, idaccion, descripcion)
VALUES(14,802,8202,'Vigilancia');

COMMIT
```

	ONG	IDPROYECTO	IDACCION	DESCRIPCION
1	14	802	8201	Reparto
2	14	802	8202	Vigilancia
3	0	101	1101	Informacion en colegios
4	0	101	1102	Informar en la calle
5	0	102	1201	Concienciacion importancia humedales
6	0	102	1202	Trabajo en humedales
7	2	201	2101	Informacion en calles
8	2	202	2202	Busqueda financiacion
9	2	202	2203	Accion sobre el terreno
10	2	203	2301	Comenzar por especies mas perjudicadas
11	4	301	3301	Recaudacion dinero
12	4	301	3302	Busqueda personas
13	4	301	3303	Reforestacion
14	6	401	4101	Buscar familias
15	6	401	4102	Conocer familias
16	6	402	4201	Preparacion Viviendas
17	6	402	4202	Reparto viviendas
18	8	501	5101	Conocer personas
19	8	501	5102	Integracion
20	10	601	2601	Reparto comida a personas en apuros
21	12	701	7101	Estudio recursos
22	12	701	7102	Busqueda casa adecuada
23	14	801	8101	Test supermercados
24	14	801	8102	Test empresas

--Datos tabla socio.

```
INSERT INTO ABD02_SOCIO(dni, nombre)
VALUES('10101010A', 'Eusebio');
```

```
INSERT INTO ABD02_SOCIO(dni, nombre)
VALUES('30000000A', 'Jose');
```

```
INSERT INTO ABD02_SOCIO(dni, nombre)
VALUES('30000001A', 'Paco');
```

```
INSERT INTO ABD02_SOCIO(dni, nombre)
VALUES('30000002A', 'Marisa');
```

```
INSERT INTO ABD02_SOCIO(dni, nombre)
VALUES('30000003A', 'Pepe');
```

```
INSERT INTO ABD02_SOCIO(dni, nombre)
VALUES('30000004A', 'Luis');
```

```
INSERT INTO ABD02_SOCIO(dni, nombre)
VALUES('30000005A', 'Luis');
```

```
INSERT INTO ABD02_SOCIO(dni, nombre)
VALUES('30000006A', 'Rodolfo');
```

```
INSERT INTO ABD02_SOCIO(dni, nombre)
VALUES('30000007A','Beatriz');
```

```
INSERT INTO ABD02_SOCIO(dni, nombre)
VALUES('30000008A','Mauricio');
```

```
INSERT INTO ABD02_SOCIO(dni, nombre)
VALUES('30000009A','Jorge');
```

```
INSERT INTO ABD02_SOCIO(dni, nombre)
VALUES('30000010A','Lucia');
```

```
INSERT INTO ABD02_SOCIO(dni, nombre)
VALUES('55555555A','Ana');
```

```
INSERT INTO ABD02_SOCIO(dni, nombre)
VALUES('88888888A','Lucia');
```

```
INSERT INTO ABD02_SOCIO(dni, nombre)
VALUES('77777777A','Lourdes');
```

```
COMMIT
```

	DNI	NOMBRE
1	55555555A	Ana
2	30000007A	Beatriz
3	10101010A	Eusebio
4	30000009A	Jorge
5	30000000A	Jose
6	77777777A	Lourdes
7	30000010A	Lucia
8	88888888A	Lucia
9	30000004A	Luis
10	30000005A	Luis
11	30000002A	Marisa
12	30000008A	Mauricio
13	30000001A	Paco
14	30000003A	Pepe
15	30000006A	Rodolfo

--Datos tabla Colaboración.

```
INSERT INTO ABD02_COLABORACION(ong, socio, fechaalta, cuota)
VALUES(8,'10101010A','04/06/2007',50);
```

```
INSERT INTO ABD02_COLABORACION(ong, socio, fechaalta, cuota)
VALUES(10, '10101010A', '04/06/2007', 50);

INSERT INTO ABD02_COLABORACION(ong, socio, fechaalta, cuota)
VALUES(12, '10101010A', '04/06/2007', 50);

INSERT INTO ABD02_COLABORACION(ong, socio, fechaalta, cuota)
VALUES(4, '30000000A', '04/06/2007', 50);

INSERT INTO ABD02_COLABORACION(ong, socio, fechaalta, cuota)
VALUES(4, '30000001A', '04/06/2007', 70);

INSERT INTO ABD02_COLABORACION(ong, socio, fechaalta, cuota)
VALUES(4, '30000002A', '04/06/2007', 50);

INSERT INTO ABD02_COLABORACION(ong, socio, fechaalta, cuota)
VALUES(4, '30000003A', '04/06/2007', 100);

INSERT INTO ABD02_COLABORACION(ong, socio, fechaalta, cuota)
VALUES(6, '30000004A', '04/06/2007', 40);

INSERT INTO ABD02_COLABORACION(ong, socio, fechaalta, cuota)
VALUES(6, '30000005A', '04/06/2007', 45);

INSERT INTO ABD02_COLABORACION(ong, socio, fechaalta, cuota)
VALUES(6, '30000006A', '04/06/2007', 25);

INSERT INTO ABD02_COLABORACION(ong, socio, fechaalta, cuota)
VALUES(2, '30000007A', '04/06/2007', 80);

INSERT INTO ABD02_COLABORACION(ong, socio, fechaalta, cuota)
VALUES(2, '30000008A', '04/06/2007', 50);

INSERT INTO ABD02_COLABORACION(ong, socio, fechaalta, cuota)
VALUES(2, '30000009A', '04/06/2007', 95);

INSERT INTO ABD02_COLABORACION(ong, socio, fechaalta, cuota)
VALUES(2, '30000010A', '04/06/2007', 300);
```

```
INSERT INTO ABD02_COLABORACION(ong, socio, fechaalta, cuota)
VALUES(12, '30000007A','04/06/2007',550);

INSERT INTO ABD02_COLABORACION(ong, socio, fechaalta, cuota)
VALUES(14, '30000007A','04/06/2007',450);

INSERT INTO ABD02_COLABORACION(ong, socio, fechaalta, cuota)
VALUES(10, '30000008A','04/06/2007',150);

INSERT INTO ABD02_COLABORACION(ong, socio, fechaalta, cuota)
VALUES(12, '30000008A','04/06/2007',50);

INSERT INTO ABD02_COLABORACION(ong, socio, fechaalta, cuota)
VALUES(14, '30000009A','04/06/2007',50);

INSERT INTO ABD02_COLABORACION(ong, socio, fechaalta, cuota)
VALUES(10, '30000010A','04/06/2007',50);

INSERT INTO ABD02_COLABORACION(ong, socio, fechaalta, cuota)
VALUES(12, '55555555A','04/06/2007',50);

INSERT INTO ABD02_COLABORACION(ong, socio, fechaalta, cuota)
VALUES(12, '88888888A','04/06/2007',50);

INSERT INTO ABD02_COLABORACION(ong, socio, fechaalta, cuota)
VALUES(10, '88888888A','04/06/2007',50);

INSERT INTO ABD02_COLABORACION(ong, socio, fechaalta, cuota)
VALUES(10, '77777777A','04/06/2007',50);

INSERT INTO ABD02_COLABORACION(ong, socio, fechaalta, cuota)
VALUES(14, '77777777A','04/06/2007',50);

COMMIT
```

	ONG	SOCIO	FECHAALTA	CUOTA
1		8 10101010A	04/06/07	50
2		10 10101010A	04/06/07	50
3		12 10101010A	04/06/07	50
4		4 30000000A	04/06/07	50
5		4 30000002A	04/06/07	50
6		4 30000003A	04/06/07	100
7		6 30000004A	04/06/07	40
8		2 30000007A	04/06/07	80
9		2 30000008A	04/06/07	50
10		14 30000009A	04/06/07	450
11		12 30000009A	04/06/07	50
12		14 30000009A	04/06/07	50
13		10 30000010A	04/06/07	50
14		12 55555555A	04/06/07	50
15		12 88888888A	04/06/07	50
16		10 88888888A	04/06/07	50
17		10 77777777A	04/06/07	50
18		14 77777777A	04/06/07	50
19		44 55555555A	04/06/07	50
20		44 30000004A	04/06/07	50
21		44 30000008A	04/06/07	50
22		4 30000001A	04/06/07	70
23		6 30000005A	04/06/07	45
24		6 30000006A	04/06/07	25
25		2 30000009A	04/06/07	95
26		2 30000010A	04/06/07	300
27		12 30000007A	04/06/07	550
28		10 30000008A	04/06/07	150
29		8 10101010A	14/06/10	50

-- Datos tabla participacion.

```

INSERT INTO ABD02_PARTICIPACION(ong, idproyecto, idaccion,
trabajador)
VALUES(0,101,1101,'10101010A');

INSERT INTO ABD02_PARTICIPACION(ong, idproyecto, idaccion,
trabajador)
VALUES(0,102,1201,'10101010A');

INSERT INTO ABD02_PARTICIPACION(ong, idproyecto, idaccion,
trabajador)
VALUES(0,101,1102,'10101010A');

INSERT INTO ABD02_PARTICIPACION(ong, idproyecto, idaccion,
trabajador)
VALUES(0,102,1202,'10101010A');

INSERT INTO ABD02_PARTICIPACION(ong, idproyecto, idaccion,
trabajador)

```

```
VALUES(2,201,2101,'12222222A');

INSERT INTO ABD02_PARTICIPACION(ong, idproyecto, idaccion,
trabajador)
VALUES(2,201,2101,'14444444A');

INSERT INTO ABD02_PARTICIPACION(ong, idproyecto, idaccion,
trabajador)
VALUES(2,202,2202,'12222222A');

INSERT INTO ABD02_PARTICIPACION(ong, idproyecto, idaccion,
trabajador)
VALUES(2,202,2203,'13333333A');

INSERT INTO ABD02_PARTICIPACION(ong, idproyecto, idaccion,
trabajador)
VALUES(2,203,2301,'13333333A');

INSERT INTO ABD02_PARTICIPACION(ong, idproyecto, idaccion,
trabajador)
VALUES(2,203,2301,'14444444A');

INSERT INTO ABD02_PARTICIPACION(ong, idproyecto, idaccion,
trabajador)
VALUES(12,701,7101,'77777777A');

INSERT INTO ABD02_PARTICIPACION(ong, idproyecto, idaccion,
trabajador)
VALUES(12,701,7102,'77777777A');

INSERT INTO ABD02_PARTICIPACION(ong, idproyecto, idaccion,
trabajador)
VALUES(4,301,3301,'12121212A');

INSERT INTO ABD02_PARTICIPACION(ong, idproyecto, idaccion,
trabajador)
VALUES(6,401,4101,'44444444A');

INSERT INTO ABD02_PARTICIPACION(ong, idproyecto, idaccion,
trabajador)
VALUES(8,501,5102,'66666666A');
```

```
INSERT INTO ABD02_PARTICIPACION(ong, idproyecto, idaccion,  
trabajador)  
VALUES(14,801,8102,'88888888A');
```

```
COMMIT
```

	ONG	IDPROYECTO	IDACCION	TRABAJADOR
1	0	101	1101 10101010A	
2	0	101	1102 10101010A	
3	0	102	1201 10101010A	
4	0	102	1202 10101010A	
5	2	201	2101 12222222A	
6	2	201	2101 14444444A	
7	2	202	2202 12222222A	
8	2	202	2203 13333333A	
9	2	203	2301 13333333A	
10	2	203	2301 14444444A	
11	4	301	3301 12121212A	
12	6	401	4101 44444444A	
13	8	501	5102 66666666A	
14	12	701	7101 77777777A	
15	12	701	7102 77777777A	
16	14	801	8102 88888888A	

5. ÍNDICES

5.1. Mover índices a otro tablespace. Mostrar con un SELECT el tablespace en el que se ha creado cada uno de los índices de ejercicios anteriores (nombre de la tabla, del índice y del tablespace del índice). Mover dichos índices (todos) al tablespace ABDXX_INDICES. Mostrar con el mismo SELECT que, efectivamente, han cambiado de tablespace.

```
select TABLE_NAME, INDEX_NAME, TABLESPACE_NAME  
from DBA_INDEXES  
where table_owner= 'ABD02';
```

TABLE_NAME	INDEX_NAME	TABLESPACE_NAME
1 ABD02_ACCION	ACCION_PK	ABD02_DATOS
2 ABD02_COLABORACION	COLABORACION_PK	ABD02_DATOS
3 ABD02_ONG	ONG_PK	ABD02_DATOS
4 ABD02_ONG	CLAVE_UNIQUE	ABD02_DATOS
5 ABD02_PAP	ID02_PAP	ABD02_INDICES
6 ABD02_PARTICIPACION	PARTICIPACION_PK	ABD02_DATOS
7 ABD02_PROYECTO	PROYECTO_PK	ABD02_DATOS
8 ABD02_SOCIO	SOCIO_PK	ABD02_DATOS
9 ABD02_TRABAJADOR	TRABAJADOR_PK	ABD02_DATOS

```
ALTER USER abd02  
QUOTA UNLIMITED  
ON ABD02_INDICES;
```

```
Tarea terminada en 0,063 segundos  
user ABD02 alterado.
```

```
ALTER INDEX ACCION_PK REBUILD TABLESPACE ABD02_INDICES;
```

```
ALTER INDEX COLABORACION_PK REBUILD TABLESPACE ABD02_INDICES;

ALTER INDEX ONG_PK REBUILD TABLESPACE ABD02_INDICES;

ALTER INDEX CLAVE_UNIQUE REBUILD TABLESPACE ABD02_INDICES;

ALTER INDEX PARTICIPACION_PK REBUILD TABLESPACE ABD02_INDICES;

ALTER INDEX PROYECTO_PK REBUILD TABLESPACE ABD02_INDICES;

ALTER INDEX SOCIO_PK REBUILD TABLESPACE ABD02_INDICES;

ALTER INDEX TRABAJADOR_PK REBUILD TABLESPACE ABD02_INDICES;

ALTER INDEX ID02_PAP REBUILD TABLESPACE ABD02_INDICES;

ALTER TABLE ABD02_SOCIO MOVE TABLESPACE ABD02_INDICES;
```

Error que empieza en la línea 21 del comando:

```
ALTER INDEX SOCIO_PK REBUILD TABLESPACE ABD02_INDICES
Informe de error:
```

Error SQL: ORA-28650: No se puede reconstruir el índice principal en una IOT

28650. 00000 - "Primary index on an IOT cannot be rebuilt"

*Cause: An attempt is made to issue alter index rebuild on IOT-TOP

*Action: Use Alter table MOVE to reorganize the table(IOT) index SYS_C00626134 alterado.

Hay una excepción, el índice SOCIO_PK, al estar en una IOT, no puede ser movido con el comando que acabamos de usar. Se debe a que la tabla socio, aunque nosotros la veamos como una tabla, no es una tabla, es un índice, por tanto, para cambiar el índice de tablespace hay que cambiar la tabla completa.

```
select TABLE_NAME, INDEX_NAME, TABLESPACE_NAME
from DBA_INDEXES
where table_owner= 'ABD02';
```

	TABLE_NAME	INDEX_NAME	TABLESPACE_NAME
1	AED02_ACCION	ACCION_PK	ABD02_INDICES
2	AED02_COLABORACION	COLABORACION_PK	ABD02_INDICES
3	AED02_ONG	ONG_PK	ABD02_INDICES
4	AED02_ONG	CLAVE_UNIQUE	ABD02_INDICES
5	ABD02_PAP	ID02_PAP	ABD02_INDICES
6	ABD02_PARTICIPACION	PARTICIPACION_PK	ABD02_INDICES
7	AED02_PROYECTO	PROYECTO_PK	ABD02_INDICES
8	AED02_SOCIO	SOCIO_PK	ABD02_INDICES
9	ABD02_TRABAJADOR	TRABAJADOR_PK	ABD02_INDICES

5.2.Crear índices en el tablespace ABDXX_INDICES.

- En ABDXX_SOCIO, crear un índice llamado ABDXX_SOCIO_NOMBRE sobre la columna nombre.

```
create index ABD02_SOCIO_NOMBRE
ON ABD02_SOCIO(nombre)
TABLESPACE ABD02_INDICES;
```

- En ABDXX_TRABAJADOR, un índice llamado ABDXX_TRABAJADOR_FECHAINGRESO, de tipo global, con dos particiones por rango de la columna fechaingreso.

```
create index ABD02_TRABAJADOR_FECHAINGRESO
ON ABD02_TRABAJADOR(fechaingreso)
GLOBAL PARTITION BY RANGE (fechaingreso)
(
    PARTITION p1_ABD02_TRABAJADOR_INGRESO      VALUES LESS
    THAN(TO_DATE('01/01/2012' , 'dd/mm/yyyy'))
    TABLESPACE ABD02_INDICES,
```

```
PARTITION p2_ABD02_TRABAJADOR_INGRESO VALUES LESS THAN  
(MAXVALUE)  
TABLESPACE ABD02_INDICES);
```

- En ABDXX_TRABAJADOR, un índice llamado ABDXX_TRABAJADOR_ESVOLUNTARIO, de tipo mapa de bits (bitmap) sobre la columna esvoluntario.

```
ALTER TABLESPACE ABD02_INDICES ADD DATAFILE 'ABD02_INDICES2.dbf'  
size 256K reuse  
autoextend on maxsize 512K;
```

```
CREATE BITMAP INDEX ABD02_TRABAJADOR_ESVOLUNTARIO  
ON ABD02_TRABAJADOR(esVoluntario)  
LOCAL TABLESPACE ABD02_INDICES;
```

- En ABDXX_COLABORACION, un índice llamado ABDXX_COLABORACION_FECHAALTA, de tipo particionado local, sobre la columna fechaalta.

```
CREATE INDEX ABD02_COLABORACION_FECHAALTA  
ON ABD02_COLABORACION(fechaalta)  
LOCAL TABLESPACE ABD02_INDICES;
```

5.3. Visualizar con un SELECT las particiones del índice ABDXX_TRABAJADOR_FECHAINGRESO (nombre del índice, nombre de la partición y valor máximo de la partición).

```
select index_name as nombre_indice, partition_name as  
nombre_particion, High_value_length as valor_maximo_particion  
from dba_ind_partitions
```

```
where      index_owner=      'ABD02'      and      index_name      =
'ABD02_TRABAJADOR_FECHAINGRESO';
```

NOMBRE_INDICE	NOMBRE_PARTITION	VALOR_MAXIMO_PARTITION
1 ABD02_TRABAJADOR_FECHAINGRESO	P1_ABD02_TRABAJADOR_INGRESO	83
2 ABD02_TRABAJADOR_FECHAINGRESO	P2_ABD02_TRABAJADOR_INGRESO	8

5.4.Mostrar con SELECT las particiones y subparticiones del índice ABDXX_COLABORACION_FECHAALTA (nombre del índice, nombre de la partición, nombre de la subpartición y valor máximo de la (ojo!) partición).

```
select      B.index_name,      B.partition_name,      A.subpartition_name,
B.high_value_length
from
((
    select *
    from dba_ind_subpartitions
    where      index_owner=      'ABD02'      and      index_name      =
'ABD02_COLABORACION_FECHAALTA'
) A
left outer join
(select *
from dba_ind_partitions
    where      index_owner=      'ABD02'      and      index_name      =
'ABD02_COLABORACION_FECHAALTA'
        )      B      on      A.index_name=B.index_name      and
A.partition_name=B.partition_name  )
;
```

INDEX_NAME	PARTITION_NAME	SUBPARTITION_NAME	HIGH_VALUE_LENGTH
1 ABD02_COLABORACION_FECHAALTA	COLAB_ANTIGUA	SYS_SUBP2704	83
2 ABD02_COLABORACION_FECHAALTA	COLAB_ANTIGUA	SYS_SUBP2705	83
3 ABD02_COLABORACION_FECHAALTA	COLAB_RECIENTE	SYS_SUBP2706	8
4 ABD02_COLABORACION_FECHAALTA	COLAB_RECIENTE	SYS_SUBP2707	8

6. VISTAS MATERIALIZADAS

6.1.Crear una vista materializada llamada ABDXX_VM_ONG que muestre cuántos socios colaboradores tiene cada ONG (la vista ha de tener dos columnas: ONG y num_socios), con carga inmediata y refresco rápido (incremental) al hacer COMMIT.

- **Crear la vista materializada.**

Para crear la vista materializada con refresco rápido necesitamos un log de los cambios que se realizan.

```
CREATE MATERIALIZED VIEW LOG ON ABD02_COLABORACION with ROWID, SEQUENCE, PRIMARY KEY INCLUDING NEW VALUES;
```

materialized view log ABD02_COLABORACION creado.

```
CREATE MATERIALIZED VIEW ABD02_VM_ONG
BUILD IMMEDIATE
REFRESH FAST ON COMMIT
AS SELECT ong ONG, COUNT(*) CUANTOS_SOCIOS
FROM ABD02_COLABORACION
GROUP BY ong;
materialized view ABD02_VM_ONG creado.
```

- **Mostrar con un SELECT el contenido de la vista.**

```
select *  
from abd02_vm_ong;
```

	ONG	CUANTOS_SOCIOS
1	2	4
2	4	4
3	6	3
4	8	1
5	10	5
6	12	5
7	14	3

- **Si es necesario, refrescar los datos de la vista.**
No es necesario, ya esta cargado
- **Volver a mostrar con un SELECT el contenido de la vista.**

```
select *  
from abd02_vm_ong;
```

	ONG	CUANTOS_SOCIOS
1	2	4
2	4	4
3	6	3
4	8	1
5	10	5
6	12	5
7	14	3

- **Insertar una fila en la tabla ABDXX_COLABORACION.**
INSERT INTO ABD02_COLABORACION(ong, socio, fechaalta, cuota)
VALUES(8, '55555555A', '04/06/2007', 50);
- **Si es necesario, volver a refrescar los datos de la vista.**
Sí, es necesario ya que hemos modificado la tabla
- **Volver a mostrar con un SELECT el contenido de la vista.**

```
select *  
from abd02_vm_ong;
```

	ONG	CUANTOS_SOCIOS
1	2	4
2	4	4
3	6	3
4	8	2
5	10	5
6	12	5
7	14	3

- Borrar la fila insertada en la tabla ABDXX_COLABORACION.

```
delete from ABD02_COLABORACION
where socio='55555555A' and ong=8;
```

- Si es necesario, volver a refrescar los datos de la vista.

Sí, es necesario ya que hemos modificado la tabla

- Volver a mostrar con un SELECT el contenido de la vista.

```
select *
from abd02_vm_ong;
```

	ONG	CUANTOS_SOCIOS
1	2	4
2	4	4
3	6	3
4	8	1
5	10	5
6	12	5
7	14	3

6.2.Eliminar la vista y repetir el ejercicio creando la misma vista materializada pero con carga diferida y refresco completo bajo demanda.

```
drop materialized view abd02_vm_ong;
DROP MATERIALIZED VIEW LOG ON ABD02_COLABORACION;
materialized view ABD02_VM_ONG borrado.
```

materialized view log ABD02_COLABORACION borrado.

- **Crear la vista materializada.**

```
CREATE MATERIALIZED VIEW ABD02_VM_ONG
TABLESPACE ABD02_DATOS
BUILD DEFERRED
REFRESH COMPLETE ON DEMAND
AS SELECT ong ONG, COUNT(*) CUANTOS_SOCIOS
FROM ABD02_COLABORACION
GROUP BY ong;
```

materialized view ABD02_VM_ONG creado.

- **Mostrar con un SELECT el contenido de la vista.**

```
select *
from abd02_vm_ong;
```

ONG	CUANT...

- **Si es necesario, refrescar los datos de la vista.**

Sí, es necesario refrescar para mostrar los datos, para ello llamando a la función:
execute DBMS_MVIEW.REFRESH('ABD02_VM_ONG','c');

- **Volver a mostrar con un SELECT el contenido de la vista**

```
select *
from abd02_vm_ong;
```

	ONG	CUANTOS_SOCIOS
1	2	4
2	4	4
3	6	3
4	8	1
5	10	5
6	12	5
7	14	3

- Insertar una fila en la tabla ABDXX_COLABORACION.

```
INSERT INTO ABD02_COLABORACION(ong, socio, fechaalta, cuota)
VALUES(8, '55555555A', '04/06/2007', 50);
```

- Si es necesario, volver a refrescar los datos de la vista.

Sí, es necesario refrescar para mostrar los datos, para ello llamando a la función:
 execute DBMS_MVIEW.REFRESH('ABD02_VM_ONG','c');

- Volver a mostrar con un SELECT el contenido de la vista.

```
SELECT *
FROM abd02_vm_ong;
```

	ONG	CUANTOS_SOCIOS
1	2	4
2	4	4
3	6	3
4	8	2
5	10	5
6	12	5
7	14	3

- Borrar la fila insertada en la tabla ABDXX_COLABORACION.

```
delete from ABD02_COLABORACION
where socio='55555555A' and ong=8;
```

- Si es necesario, volver a refrescar los datos de la vista.

Sí, es necesario refrescar para mostrar los datos, para ello llamando a la función:
 execute DBMS_MVIEW.REFRESH('ABD02_VM_ONG','c');

- Volver a mostrar con un SELECT el contenido de la vista.

```
SELECT *
FROM abd02_vm_ong;
```

	ONG	CUANTOS_SOCIOS
1	2	4
2	4	4
3	6	3
4	8	1
5	10	5
6	12	5
7	14	3

6.3.Destruir la vista y repetir el ejercicio completo, creando la misma vista materializada pero con carga inmediata y refresco completo al hacer COMMIT.

```
drop materialized view abd02_vm_ong;
```

- Crear la vista materializada.

```
CREATE MATERIALIZED VIEW LOG ON ABD02_COLABORACION with ROWID, SEQUENCE, PRIMARY
KEY INCLUDING NEW VALUES;
```

```
CREATE MATERIALIZED VIEW ABD02_VM_ONG
TABLESPACE ABD02_DATOS
BUILD IMMEDIATE
REFRESH COMPLETE ON COMMIT
AS SELECT ong ONG, COUNT(*) CUANTOS_SOCIOS
FROM ABD02_COLABORACION
GROUP BY ong;
```

materialized view log ABD02_COLABORACION creado.

materialized view ABD02_VM_ONG creado.

- Mostrar con un SELECT el contenido de la vista.

```
SELECT *
FROM abd02_vm_ong;
```

	ONG	CUANTOS_SOCIOS
1	2	4
2	4	4
3	6	3
4	8	1
5	10	5
6	12	5
7	14	3

- Si es necesario, refrescar los datos de la vista.

No es necesario, ya esta cargado

- Volver a mostrar con un SELECT el contenido de la vista.

`SELECT *`

`FROM abd02_vm_ong;`

	ONG	CUANTOS_SOCIOS
1	2	4
2	4	4
3	6	3
4	8	1
5	10	5
6	12	5
7	14	3

- Insertar una fila en la tabla ABDXX_COLABORACION.

`INSERT INTO ABD02_COLABORACION(ong, socio, fechaalta, cuota)`
`VALUES(8, '55555555A', '04/06/2007', 50);`

- Si es necesario, volver a refrescar los datos de la vista.

Sí, es necesario llamar a commit para actualizar.

- Volver a mostrar con un SELECT el contenido de la vista.

`SELECT *`

`FROM abd02_vm_ong;`

	ONG	CUANTOS_SOCIOS
1	2	4
2	4	4
3	6	3
4	8	2
5	10	5
6	12	5
7	14	3

- Borrar la fila insertada en la tabla ABDXX_COLABORACION.

```
delete from ABD02_COLABORACION
where socio='55555555A' and ong=8;
```

- Si es necesario, volver a refreshar los datos de la vista.

Sí, es necesario llamar a commit para actualizar.

- Volver a mostrar con un SELECT el contenido de la vista.

```
SELECT *
FROM abd02_vm_ong;
```

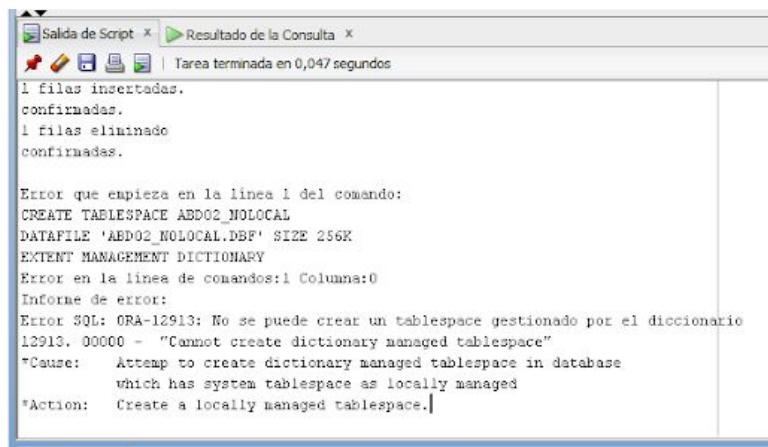
	ONG	CUANTOS_SOCIOS
1	2	4
2	4	4
3	6	3
4	8	1
5	10	5
6	12	5
7	14	3

7. TABLESPACES: tipos y modos

7.1. Crear un tablespace ABDXX_NOLOCAL, de 256K, no manejado localmente. Justificar los resultados.

```
CREATE TABLESPACE ABD02_NOLOCAL
DATAFILE 'ABD02_NOLOCAL.DBF' SIZE 256K
EXTENT MANAGEMENT DICTIONARY;
```

El tablespace SYSTEM ha sido creado en local. Por lo tanto, no podemos crear un tablespace que no esté manejado localmente.



The screenshot shows the Oracle SQL Developer interface with two tabs: 'Salida de Script' and 'Resultado de la Consulta'. The 'Resultado de la Consulta' tab displays the following output:

```
1 filas insertadas.
confirmadas.
1 filas eliminado
confirmadas.

Error que empieza en la linea 1 del comando:
CREATE TABLESPACE ABD02_NOLOCAL
DATAFILE 'ABD02_NOLOCAL.DBF' SIZE 256K
EXTENT MANAGEMENT DICTIONARY
Error en la linea de comandos:1 Columna:0
Informe de error:
Error SQL: ORA-12913: No se puede crear un tablespace gestionado por el diccionario
12913. 00000 - "Cannot create dictionary managed tablespace"
Cause: Attempt to create dictionary managed tablespace in database
which has system tablespace as locally managed
Action: Create a locally managed tablespace.
```

7.2. Tablespaces READONLY. Razonar cada resultado obtenido al realizar estas acciones:
En el tablespace ABDXX_DATOS crear una tabla ABDXX_R de 32KB.

```
CREATE TABLE ABD02_R(nombre varchar(20))
TABLESPACE ABD02_datos
STORAGE (INITIAL 32k);
table ABD02_R creado.
```

Insertar una fila en la tabla ABDXX_R.

```
INSERT INTO ABD02_R VALUES (1);
1 filas insertadas.
```

Poner el tablespace ABDXX_DATOS en modo READONLY.

```
ALTER TABLESPACE ABD02_DATOS READ ONLY;
tablespace ABD02_DATOS alterado.
```

A partir de este momento, el tablespace en cuestión pasa al modo READ-ONLY. Por defecto, estaba en READ-WRITE.

Insertar otra fila en la tabla ABDXX_R. Razonar el resultado obtenido.

```
INSERT INTO ABD02_R VALUES (2);
```

Lo normal es que falle al intentar meter una nueva fila. Se debe a que la tabla está en un tablespace en modo sólo lectura, por tanto, no se podrá modificar el contenido de la tabla.

The screenshot shows the Oracle SQL Developer interface. In the top window, the script editor contains the following code:

```
1 CREATE TABLE ABD02_R(nombr varchar(20))
2 TABLESPACE ABD02_datos
3 STORAGE (INITIAL 32k);
4
5 INSERT INTO ABD02_R VALUES (1);
6
7 ALTER TABLESPACE ABD02_DATOS READ ONLY;
8 INSERT INTO ABD02_R VALUES (2);
```

In the bottom window, the results of the query show an error message:

```
*Action:
  table ABD02_R created.
  1 filas insertadas.
  tablespace ABD02_DATOS altered.

Error que empieza en la linea 8 del comando:
INSERT INTO ABD02_R VALUES (2)
Informes de errores:
Error SQL: ORA-00372: el archivo 17 no se puede modificar en este momento
ORA-01110: archivo de datos 17: 'C:\ORACLE\PRODUCT11.2.0\ORACLE_1\DATABASE\ABD02_DATOS1.DBF'
00372.. 00000 - "file is cannot be modified at this time"
*Cause:
  attempting to modify the contents of a file that cannot be
  modified. The file is most likely part of a read only tablespace
  but may be in the process of going offline, or the database may
  be in the process of closing.
*Action: check the status of the file and its tablespace
```

Mostrar con un SELECT el contenido de la tabla ABDXX_R.

```
SELECT *
FROM ABD02_R;
```

The screenshot shows the Oracle SQL Developer interface. In the top window, the script editor contains the following code:

```
1 CREATE TABLE ABD02_R(nombr varchar(20))
2 TABLESPACE ABD02_datos
3 STORAGE (INITIAL 32k);
4
5 INSERT INTO ABD02_R VALUES (1);
6
7 ALTER TABLESPACE ABD02_DATOS READ ONLY;
8 INSERT INTO ABD02_R VALUES (2);
9
10 SELECT *
11 FROM ABD02_R;
```

In the bottom window, the results of the query show the following output:

NOMBRE
1

La operación en cuestión, no supone ningún tipo de problema porque si se permiten las operaciones de lectura.

Borrar la tabla ABDXX_R. Justificar el comportamiento del sistema.

```
DROP TABLE ABD02_R;
```

Mientras se encuentra en modo solamente de lectura, no permite llevar a cabo la inserción de datos, por ejemplo, pero si la eliminación, como puede comprobarse en este ejercicio.

Volver a poner el tablespace ABDXX_DATOS en modo READWRITE.

```
ALTER TABLESPACE ABD02_DATOS READ WRITE;
```

tablespace ABD02_DATOS alterado.

De nuevo el tablespace vuelve al modo READ-WRITE, por lo que volverían a poder realizarse todas las operaciones como anteriormente.

7.3. Tablespaces OFFLINE. Justificar los resultados obtenidos al realizar cada una de estas acciones:

En el tablespace ABDXX_DATOS crear una tabla ABDXX_O de 64KB.

```
CREATE TABLE ABD02_O(
nombre varchar(50)
)
TABLESPACE ABD02_datos
STORAGE (INITIAL 64k);
```

table ABD02_O creado.

Insertar una fila en la tabla ABDXX_O.

```
INSERT INTO ABD02_O VALUES (1);
```

Poner el tablespace ABDXX_DATOS en modo OFFLINE.

```
ALTER TABLESPACE ABD02_DATOS OFFLINE;
```

Insertar otra fila en la tabla ABDXX_O.

```
INSERT INTO ABD02_O VALUES (2);
```

```

12
13 DROP TABLE ABD02_R;
14 ALTER TABLESPACE ABD02_DATOS READ WRITE;
15
16
17 CREATE TABLE ABD02_0(
18 nombre varchar(50)
19 )
20 TABLESPACE ABD02_datos
21 STORAGE (INITIAL 64k);
22
23 INSERT INTO ABD02_0 VALUES (1);
24 ALTER TABLESPACE ABD02 DATOS OFFLINE;
25 INSERT INTO ABD02_0 VALUES (2);
26

```

Salida de Script | Resultado de la Consulta X

Tarea terminada en 0,045 segundos

Error SQL: ORA-00904: "ID": identificador no válido
00904. 00000 - "%s: invalid identifier"

*Cause:
*Action:
1 filas insertadas.
tablespace ABD02_DATOS alterado.

Error que empieza en la linea 25 del comando:
INSERT INTO ABD02_0 VALUES (2)

Informes de error:

Error SQL: ORA-00376: el archivo 17 no se puede leer en este momento
ORA-01110: archivo de datos 17: 'C:\ORACLE\PRODUCT\11.2.0\DBHOME_1\DATABASE\ABD02_DATOS1.DBF'
00376. 00000 - "file %s cannot be read at this time"

*Cause: attempting to read from a file that is not readable. Most likely
the file is offline.

*Action: Check the state of the file. Bring it online

Al estar en modo OFFLINE deja de estar disponible y por eso no se puede insertar la fila.

Mostrar con un SELECT el contenido de la tabla ABDXX_O.

```
SELECT *
FROM abd02_o;
```

```

15
16
17 CREATE TABLE ABD02_0(
18 nombre varchar(50)
19 )
20 TABLESPACE ABD02_datos
21 STORAGE (INITIAL 64k);
22
23 INSERT INTO ABD02_0 VALUES (1);
24 ALTER TABLESPACE ABD02_DATOS OFFLINE;
25 INSERT INTO ABD02_0 VALUES (2);
26
27
28 SELECT *
29 FROM abd02_o;

```

Salida de Script | Resultado de la Consulta X

En Ejecución:SELECT *FROM abd02_o en 0 segundos

ORA-00375: el archivo 17 no se puede leer en este momento
ORA-01110: archivo de datos 17: 'C:\ORACLE\PRODUCT\11.2.0\DBHOME_1\DATABASE\ABD02_DATOS1.DBF'
00375. 00000 - "file %s cannot be read at this time"

*Cause: attempting to read from a file that is not readable. Most likely
the file is offline.

*Action: Check the state of the file. Bring it online

La tabla no se puede leer porque el tablespace está offline. En modo OFFLINE, el tablespace dejó de estar disponible, para que un usuario acceda a los datos.

Borrar la tabla ABDXX_O.

```
DROP TABLE ABD02_0;
```

Aunque el tablespace esté offline se puede borrar la tabla. Se debe a que la operación de borrado afecta al diccionario de datos y no al tablespace.

Volver a poner ONLINE el tablespace ABDXX_DATOS.

```
ALTER TABLESPACE ABD02_DATOS ONLINE;
```

7.4. Tablespaces UNDO. Responder razonadamente a las siguientes cuestiones:

Mostrar el tablespace de undo activo en este momento.

```
SELECT TABLESPACE_NAME, STATUS, CONTENTS  
FROM dba tablespaces  
WHERE CONTENTS = 'UNDO' AND STATUS='ONLINE';
```



TABLESPACE_NAME	STATUS	CONTENTS
UNDOTBS1	ONLINE	UNDO

Se busca el tablespace de undo.

Crear un tablespace de undo ABDXX_UNDO de 256KB.

```
CREATE UNDO TABLESPACE ABD02_UNDO  
DATAFILE 'ABD02_UNDOEJ7.DBF'  
SIZE 256 K;  
undo TABLESPACE creado.
```

Se crea un tablespace de undo, para ello sólo hay que indicar el parámetro “UNDO”, a la hora de crear un tablespace.

Crear una tabla ABDXX_U de 32k en el nuevo tablespace ABDXX_UNDO.

```
CREATE TABLE ABD02_U(nombre varchar(3))  
TABLESPACE ABD02_UNDO  
STORAGE (INITIAL 32 k);
```

table ABD02_U creado.

Mostrar con un SELECT el nombre de todos los segmentos de rollback de la base de datos, indicando el tablespace en el que están y si están online u offline.

```
SELECT SEGMENT_NAME, TABLESPACE_NAME, STATUS  
FROM dba_rollback_segs;
```

SEGMENT_NAME	TABLESPACE_NAME	STATUS
1 SYSTEM	SYSTEM	ONLINE
2 _SYSSMU1_1510548437\$	UNDOTBS1	ONLINE
3 _SYSSMU2_2082490410\$	UNDOTBS1	ONLINE
4 _SYSSMU3_991555123\$	UNDOTBS1	ONLINE
5 _SYSSMU4_2369290268\$	UNDOTBS1	ONLINE
6 _SYSSMU5_1018230376\$	UNDOTBS1	ONLINE
7 _SYSSMU6_1834113595\$	UNDOTBS1	ONLINE
8 _SYSSMU7_137577880\$	UNDOTBS1	ONLINE
9 _SYSSMU8_1557854099\$	UNDOTBS1	ONLINE
10 _SYSSMU9_1126410412\$	UNDOTBS1	ONLINE
11 _SYSSMU10_3176102001\$	UNDOTBS1	ONLINE
12 _SYSSMU11_736218092\$	UNDOTBS1	OFFLINE
13 _SYSSMU12_3795579306\$	UNDOTBS1	OFFLINE
14 _SYSSMU13_932667993\$	UNDOTBS1	OFFLINE
15 _SYSSMU14_1287061253\$ ABD01_UNDO	ABD01_UNDO	OFFLINE
16 _SYSSMU15_1517554312\$ ABD02_UNDO	ABD02_UNDO	OFFLINE

En este caso, sólo hay que utilizar el select adecuado, para obtener la información. Buscando para ello, en dba_rollback_seg.

¿Cómo se podrían poner online los segmentos de rollback del tablespace ABDXX_UNDO?

Ejemplo: ALTER ROLLBACK SEGMENT NOMBRE_SEGMENTO ONLINE.

```
ALTER SYSTEM TABLESPACE ABD02_UNDO ONLINE;
```

```
ALTER SYSTEM SET UNDO_TABLESPACE=ABD02_UNDO;
```

La idea sería, en primer lugar hacer que todos los segmentos de rollback del tablespace se encuentren disponibles. Luego se hace lo mismo con el tablespace ABD02_UNDO. Y para finalizar, se hace que el tablespace ABD02_UNDO se convierta en el tablespace de UNDO del sistema.

Borrar completamente el tablespace ABDXX_UNDO. Comprobar si es necesario borrar la tabla ABDXX_U (en el caso de existir ésta) y el fichero empleado en la creación del tablespace.

```
SELECT * FROM ABD02_U;
```



```
DROP TABLESPACE ABD02_UNDO  
INCLUDING CONTENTS AND DATAFILES;  
DROP TABLE ABD02_U;  
tablespace ABD02_UNDO borrado.
```

Utilizando lo anterior, se consigue la eliminación completa del tablespace, además de la tabla y el fichero. En caso contrario, se producirían fallos en otros ejercicios posteriores.

7.5. Tablespaces TEMPORALES. Justificar los resultados obtenidos tras realizar cada acción:

Mostrar con un SELECT el nombre del tablespace temporal por defecto del usuario ABDXX. Anótelo, pues luego deberá recordar cuál era.

```
SELECT USERNAME, DEFAULT_TABLESPACE, TEMPORARY_TABLESPACE  
FROM DBA_USERS  
WHERE USERNAME='ABD02';
```

A screenshot of a database management interface showing the results of a SQL query. The window has tabs for 'Salida de Script' and 'Resultado de la Consulta'. The 'Resultado de la Consulta' tab is active and shows the following output:

USERNAME	DEFAULT_TABLESPACE	TEMPORARY_TABLESPACE
ABD02	ABD02_DATOS	ABD02_TEMP

The table has three columns: 'USERNAME', 'DEFAULT_TABLESPACE', and 'TEMPORARY_TABLESPACE'. There is one row with the values 'ABD02', 'ABD02_DATOS', and 'ABD02_TEMP' respectively. The interface includes standard toolbar icons like back, forward, and search, and a status bar at the bottom indicating 'Todas las Filas Recuperadas: 1 en 0,063 segundos'.

Crear un tablespace temporal ABDXX_TEMPORAL del menor tamaño posible.

Tal y como se explicó en el ejercicio 2.2 sabemos que la mínima extensión posible tiene que ser de 5 bloques. Por lo tanto, creamos un tablespace temporal con un tamaño de 40K ($8K \times 5 = 40$ K).

```
CREATE TEMPORARY TABLESPACE ABD02_TEMPORAL  
TEMPFILE 'ABD02_TEMPORAL.DBF'  
SIZE 40K REUSE  
EXTENT MANAGEMENT LOCAL UNIFORM SIZE 1K;  
temporary TABLESPACE creado.
```

Cómo puede verse, solo hay que indicar el parámetro “TEMPORARY”, para indicar que el tablespace es temporal.

Ponerlo en modo readonly.

No existe para uno temporal, No se puede poner en modo READ ONLY.

Ponerlo en modo offline. Si tiene éxito, volver a ponerlo online.

```
ALTER TABLESPACE ABD02_TEMPORAL OFFLINE;
```

The screenshot shows the Oracle SQL Developer interface. In the top-left corner, there's a menu bar with 'Control de Versiones', 'Herramientas', and 'Ayuda'. Below the menu is a toolbar with various icons. The main area has two tabs: 'Hoja de Trabajo' (Worksheet) and 'Generador de Consultas' (Query Builder). The 'Hoja de Trabajo' tab contains the following SQL code:

```
1 ALTER TABLESPACE ABD02_TEMPORAL OFFLINE;
2
3 SELECT *
4 FROM dba_tablespaces;
5
6 SELECT tablespace_name, file_name
7 FROM dba_temp_files;
```

Below the worksheet, there are two tabs: 'Salida de Script' (Script Output) and 'Resultado de la Consulta' (Query Result). The 'Salida de Script' tab shows the error message:

```
*Cause: invalid option for alter of temporary tablespace was specified
*Action: Specify one of the valid options
        TEMPFILE ONLINE, TEMPFILE OFFLINE
Tablespace ABD02_TEMPORAL altered.
```

It also lists errors from previous attempts to alter the tablespace:

```
Error que empieza en la linea 1 del comando:
ALTER TABLESPACE ABD02_TEMPORAL ONLINE
Informe de error:
Error SQL: ORA-03217: opción no válida para ALTER de TEMPORARY TABLESPACE
03217. 00000 - "invalid option for alter of TEMPORARY TABLESPACE"
*Cause: invalid option for alter of temporary tablespace was specified
*Action: Specify one of the valid options
        TEMPFILE ONLINE, TEMPFILE OFFLINE

Error que empieza en la linea 1 del comando:
ALTER TABLESPACE ABD02_TEMPORAL OFFLINE
Informe de error:
Error SQL: ORA-03217: opción no válida para ALTER de TEMPORARY TABLESPACE
03217. 00000 - "invalid option for alter of TEMPORARY TABLESPACE"
*Cause: invalid option for alter of temporary tablespace was specified
*Action: Specify one of the valid options
        TEMPFILE ONLINE, TEMPFILE OFFLINE
```

La instrucción ALTER TABLESPACE toma los archivos de datos fuera de línea, así como también el espacio de tabla, pero no se puede usar para alterar el estado de un espacio de tabla temporal o sus archivos de temp.

¿Cómo se podría establecer como el tablespace temporal por defecto de la base de datos?

```
ALTER DATABASE
DEFAULT TEMPORARY TABLESPACE ABD02_TEMPORAL;
```

Simplemente se utiliza lo anterior que provoca lo deseado.

```

2
3   SELECT *
4   FROM dba tablespaces;
5
6   SELECT tablespace_name, file_name
7   FROM dba_temp_files;
8
9   ALTER DATABASE
10  DEFAULT TEMPORARY TABLESPACE ABD02_TEMPORAL;

```

Salida de Script | Resultado de la Consulta | Tarea terminada en 0,031 segundos

*Cause: invalid option for alter of temporary tablespace was specified
*Action: Specify one of the valid options
 TEMPFILE ONLINE, TEMPFILE OFFLINE

Error que empieza en la linea 9 del comando:
ALTER DATABASE
DEFAULT TEMPORARY TABLESPACE ABD02_TEMPORAL
Informe de error:
Error SQL: ORA-01031: privilegios insuficientes
01031. 00000 - "insufficient privileges"
*Cause: An attempt was made to change the current username or password
without the appropriate privilege. This error also occurs if
attempting to install a database without the necessary operating
system privileges.
When Trusted Oracle is configured in DBMS MAC, this error may occur
if the user was granted the necessary privilege at a higher label
than the current login.
*Action: Ask the database administrator to perform the operation or grant
the required privileges.
For Trusted Oracle users getting this error although granted the
appropriate privilege at a higher label, ask the database
administrator to regrant the privilege at the appropriate label.

Renombrarlo a ABDXX_TEMP1.

ALTER TABLESPACE ABD02_TEMPORAL RENAME TO ABD02_TEMP1;

tablespace ABD02_TEMPORAL alterado.

```

1  ALTER TABLESPACE ABD02_TEMPORAL OUTLINE;
2
3   SELECT *
4   FROM dba tablespaces;
5
6   SELECT tablespace_name, file_name
7   FROM dba_temp_files;
8
9   ALTER DATABASE

```

Salida de Script | Resultado de la Consulta | Todas las Fichas Recuperadas: 25 en 0 segundos

TABLESPACE_NAME	BLOCK_SIZE	INITIAL_EXTENT	NEXT_EXTENT	MIN_EXTENTS	MAX_EXTENTS	MAX_SIZE	PCT_INCREASE	MIN_EXTENT	STATUS	CONTENTS	LOGGING	FORCE_LOGGING	EXTENT_MANAGEMENT	ALLOCATION_TYPE	PLUGGED_IN	SH
8 EGRRADO	8192	65536	(null)	1	2147483645	2147483645	(null)	65536	ONLINE	PERMANENT	LOGGING	NO	LOCAL	SYSTEM	NO	AUTO
9 EGRRADOTEMP	8192	1048576	1048576	1	(null)	2147483645	0	1048576	ONLINE	TEMPORARY	NOLOGGING	NO	LOCAL	UNIFORM	NO	MANU
10 TAG_DATOS	8192	65536	(null)	1	2147483645	2147483645	(null)	65536	ONLINE	PERMANENT	LOGGING	NO	LOCAL	SYSTEM	NO	AUTO
11 TAG_TEMP	8192	1048576	1048576	1	(null)	2147483645	0	1048576	ONLINE	TEMPORARY	NOLOGGING	NO	LOCAL	UNIFORM	NO	MANU
12 DSI_DATOS	8192	65536	(null)	1	2147483645	2147483645	(null)	65536	ONLINE	PERMANENT	NOLOGGING	NO	LOCAL	SYSTEM	NO	AUTO
13 DSI_TEMP	8192	1048576	1048576	1	(null)	2147483645	0	1048576	ONLINE	TEMPORARY	NOLOGGING	NO	LOCAL	UNIFORM	NO	MANU
14 ABD02_DATOS	8192	32768	32768	1	2147483645	2147483645	0	32768	ONLINE	PERMANENT	LOGGING	NO	LOCAL	UNIFORM	NO	MANU
15 ABDXX_DATOS	8192	16384	16384	1	2147483645	2147483645	0	16384	ONLINE	PERMANENT	LOGGING	NO	LOCAL	UNIFORM	NO	MANU
16 ABDXX_INDICKS	8192	40960	40960	1	2147483645	2147483645	0	40960	ONLINE	PERMANENT	LOGGING	NO	LOCAL	UNIFORM	NO	AUTO
17 ABD03_DATOS	8192	32768	32768	1	2147483645	2147483645	0	32768	ONLINE	PERMANENT	LOGGING	NO	LOCAL	UNIFORM	NO	MANU
18 ABD03_INDICES	8192	40960	40960	1	2147483645	2147483645	0	40960	ONLINE	PERMANENT	LOGGING	NO	LOCAL	UNIFORM	NO	AUTO
19 ABD01_DATOS	8192	32768	32768	1	2147483645	2147483645	0	32768	ONLINE	PERMANENT	LOGGING	NO	LOCAL	UNIFORM	NO	MANU
20 ABD01_INDICES	8192	40960	40960	1	2147483645	2147483645	0	40960	ONLINE	PERMANENT	LOGGING	NO	LOCAL	UNIFORM	NO	AUTO
21 ABD01_UNDO	8192	65536	(null)	1	2147483645	2147483645	(null)	65536	ONLINE	UNDO	LOGGING	NO	LOCAL	SYSTEM	NO	MANU
22 ABD02_TEMP1	8192	16384	16384	1	(null)	2147483645	0	16384	ONLINE	TEMPORARY	NOLOGGING	NO	LOCAL	UNIFORM	NO	MANU
23 ABD_DATOS	8192	65536	(null)	1	2147483645	2147483645	(null)	65536	ONLINE	PERMANENT	LOGGING	NO	LOCAL	SYSTEM	NO	AUTO
24 ABD_TEMP	8192	1048576	1048576	1	(null)	2147483645	0	1048576	ONLINE	TEMPORARY	NOLOGGING	NO	LOCAL	UNIFORM	NO	MANU
25 ABD02_INDICES	8192	40960	40960	1	2147483645	2147483645	0	40960	ONLINE	PERMANENT	LOGGING	NO	LOCAL	UNIFORM	NO	AUTO

Crear una tabla (no temporal) ABDXX_T en dicho tablespace.

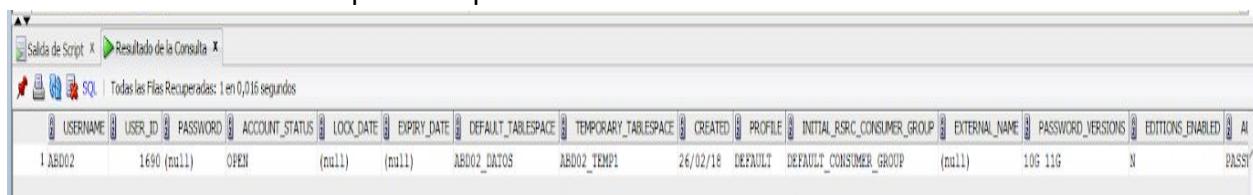
```
CREATE TABLE ABD02_T (nombre varchar(20) )
TABLESPACE ABD02_TEMP1;
```

table ABD02_T creado.

Asignar al usuario ABDXX el tablespace ABDXX_TEMP1 como su tablespace temporal por defecto.

```
ALTER USER ABD02 TEMPORARY TABLESPACE ABD02_TEMP1;
select *
from dba_users
where username= 'ABD02';
```

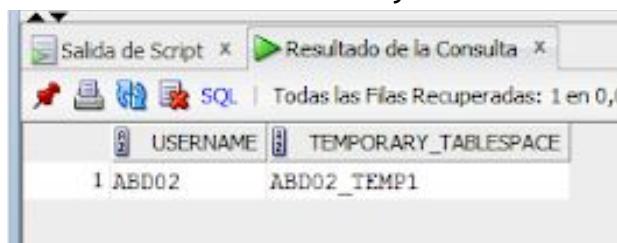
Se utiliza Alter user, y es tan sencillo como indicar el usuario y el tablespace deseado, de ese modo se modifica el tablespace temporal del usuario en cuestión.



USERNAME	USER_ID	PASSWORD	ACCOUNT_STATUS	LOCK_DATE	EXPIRY_DATE	DEFAULT_TABLESPACE	TEMPORARY_TABLESPACE	CREATED	PROFILE	INITIAL_RSRC_CONSUMER_GROUP	EXTERNAL_NAME	PASSWORD_VERSIONS	EDITIONS_ENABLED	LAST_PASSWORD_CHANGE
ABD02	1690 (null)	OPEN	(null)	(null)	ABD02_DATOS	ABD02_TEMP1	26/02/18	DEFAULT	DEFAULT_CONSUMER_GROUP	(null)	10G 11G	N	ABD02	2018-02-26 10:45:16

Mostrar de nuevo el nombre del tablespace temporal por defecto del usuario ABDXX.

```
SELECT USERNAME, TEMPORARY_TABLESPACE
FROM DBA_USERS
WHERE USERNAME='ABD02';
```



USERNAME	TEMPORARY_TABLESPACE
ABD02	ABD02_TEMP1

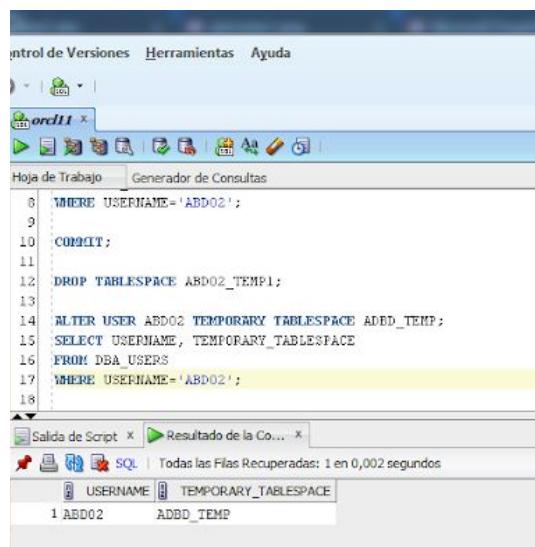
Reasignar al usuario ABDXX su tablespace temporal por defecto original.

Como recordamos del apartado 1 de este ejercicio el tablespace temporal por defecto original de este usuario era el tablespace 'TEMP'.

```
ALTER USER ABD02 TEMPORARY TABLESPACE ADBD_TEMP;
```

Igual que anteriormente, pero con el indicado.

user ABD02 alterado.



The screenshot shows the Oracle SQL Developer interface. In the top-left corner, there's a toolbar with various icons. Below it is a menu bar with 'Control de Versiones', 'Herramientas', and 'Ayuda'. The main window has tabs for 'Hoja de Trabajo' and 'Generador de Consultas'. The 'Hoja de Trabajo' tab is active, displaying a script in the following code block:

```
6 WHERE USERNAME='ABD02';
7
8 COMMIT;
9
10
11
12 DROP TABLESPACE ABD02_TEMP1;
13
14 ALTER USER ABD02 TEMPORARY TABLESPACE ADBD_TEMP;
15 SELECT USERNAME, TEMPORARY_TABLESPACE
16 FROM DBA_USERS
17 WHERE USERNAME='ABD02';
18
```

Below the script, there's a 'Salida de Script' tab and a 'Resultado de la Co...' tab. The 'Resultado de la Co...' tab shows the output of the query in the previous step:

USERNAME	TEMPORARY_TABLESPACE
1 ABD02	ADBD_TEMP

Borrar por completo de la base de datos el tablespace ABDXX_TEMP1.

```
DROP TABLESPACE ABD02_TEMP1;
tablespace ABD02_TEMP1 borrado.
```

8. OPTIMIZACIÓN

8.1. Mostrar el modo de optimización de consultas de la sesión.

```
show parameter optimizer_mode;
```

NAME	TYPE	VALUE
optimizer_mode	string	ALL_ROWS

8.2. Borrar todas las estadísticas que puedan existir relacionadas con el esquema ABDXX [mediante una invocación del procedimiento DBMS_STATS.delete_schema_stats(...);]

```
EXECUTE DBMS_STATS.delete_schema_stats('ABD02');
```

bloque anónimo terminado

8.3. Computar las estadísticas de todos los elementos (objetos) del esquema ABDXX [invocación del procedimiento DBMS_UNUTILITY.analyze_schema(...);].

```
EXECUTE DBMS_UNUTILITY.analyze_schema('ABD02', 'COMPUTE');
```

Hubo que borrar las tablas creadas en temporal y undo, porque no dejaba.

bloque anónimo terminado

8.4. Mostrar con tres SELECT las estadísticas de las tablas, las estadísticas de las columnas de las tablas, y las estadísticas de los índices.

```
select *
from dba_tables;
```

SQ | Se han recuperado 50 filas en 0,126 segundos

OWNER	TABLE_NAME	TABLESPACE_NAME	CLUSTER_NAME	LOG_NAME	STATUS	PCT_FREE	PCT_USED	INC_TRANS	MAX_TRANS	INITIAL_EXTENT	NEXT_EXTENT	MIN_EXTENTS	MAX_EXTENTS	PC
1 SYS	COL\$	SYSTEM	C.COL\$#	(null)	VALID	0	0	0	0	57344	1048576	1	214743645	PC
2 SYS	CDE\$	SYSTEM	C.CDE\$#	(null)	VALID	0	0	0	0	57344	1048576	1	214743645	
3 SYS	UE\$	SYSTEM	C.FILE\$_BLOCK\$#	(null)	VALID	0	0	0	0	24576	1048576	1	214743645	
4 SYS	SEG\$	SYSTEM	C.FILE\$_BLOCK\$#	(null)	VALID	0	0	0	0	24576	1048576	1	214743645	
5 SYS	MLO\$	SYSTEM	C.MLO\$#	(null)	VALID	0	0	0	0	65536	1048576	1	214743645	
6 SYS	SLD\$	SYSTEM	C.SLD\$#	(null)	VALID	0	0	0	0	65536	1048576	1	214743645	
7 SYS	ICOL\$	SYSTEM	C.ICOL\$#	(null)	VALID	0	0	0	0	139264	204800	1	214743645	
8 SYS	IND\$	SYSTEM	C.IND\$#	(null)	VALID	0	0	0	0	139264	204800	1	214743645	
9 SYS	COL\$	SYSTEM	C.COL\$#	(null)	VALID	0	0	0	0	139264	204800	1	214743645	
10 SYS	CLU\$	SYSTEM	C.CLU\$#	(null)	VALID	0	0	0	0	139264	204800	1	214743645	
11 SYS	TAB\$	SYSTEM	C.CTAB\$#	(null)	VALID	0	0	0	0	139264	204800	1	214743645	
12 SYS	LDR\$	SYSTEM	C.CLDR\$#	(null)	VALID	0	0	0	0	139264	204800	1	214743645	
13 SYS	COLTYPE\$	SYSTEM	C.COL\$#	(null)	VALID	0	0	0	0	139264	204800	1	214743645	
14 SYS	SUBCOLTYPE\$	SYSTEM	C.CTAB\$#	(null)	VALID	0	0	0	0	139264	204800	1	214743645	
15 SYS	NTAB\$	SYSTEM	C.CTAB\$#	(null)	VALID	0	0	0	0	139264	204800	1	214743645	
16 SYS	REFCONG	SYSTEM	C.CTAB\$#	(null)	VALID	0	0	0	0	139264	204800	1	214743645	
17 SYS	OPTTYPE\$	SYSTEM	C.CTAB\$#	(null)	VALID	0	0	0	0	139264	204800	1	214743645	
18 SYS	ICOLLEP\$	SYSTEM	C.CTAB\$#	(null)	VALID	0	0	0	0	139264	204800	1	214743645	

```
select *
from dba_tab_col_statistics;
```

SQ | Se han recuperado 50 filas en 0,149 segundos

OWNER	TABLE_NAME	COLUMN_NAME	NUM_DISTINCT	LOW_VALUE	HIGH_VALUE	DENSITY	NUM_NULLS	NUM_BUCKETS
1 SYS	JAVA\$COMPILETIMEOPTIONS	PROPERTY	0 (null)	(null)	(null)	0	0	0.02%
2 SYS	JAVA\$COMPILETIMEOPTIONS	OWNER#	0 (null)	(null)	(null)	0	0	0.02%
3 SYS	JAVA\$COMPILETIMEOPTIONS	OWNER#	0 (null)	(null)	(null)	0	0	0.02%
4 SYS	JAVA\$JVM\$DBC	LOG	0 (null)	(null)	(null)	0	0	0.02%
5 SYS	JAVA\$JVM\$DBC	PATH	0 (null)	(null)	(null)	0	0	0.02%
6 SYS	JAVA\$JVM\$DBC	ID	0 (null)	(null)	(null)	0	0	0.02%
7 SYS	JAVA\$JVM\$RUNTIMEPARAMETERS	FLAGS	0 (null)	(null)	(null)	0	0	0.02%
8 SYS	JAVA\$JVM\$RUNTIMEPARAMETERS	OWNER#	0 (null)	(null)	(null)	0	0	0.02%
9 SYS	JAVA\$JVM\$STATUS	PUNTING	1 44614C5345	44614C5345	1	0	0	1.02%
10 SYS	JAVA\$JVM\$STATUS	RNTHRTIME	0 (null)	(null)	(null)	0	1	0.02%
11 SYS	JAVA\$JVM\$STATUS	EXECD	0 (null)	(null)	(null)	0	1	0.02%
12 SYS	JAVA\$JVM\$STATUS	INPROGRESS	14E	4E	1	0	1	1.02%
13 SYS	JAVA\$JVM\$STATUS	ACTION	1 4C4F4144	4C4F4144	1	0	1	1.02%
14 SYS	JAVA\$JVM\$STEPS4DONE	STEP	0 (null)	(null)	(null)	0	0	0.02%
15 SYS	JAV\$4MC\$	LENGTH	17 C20203	C20203	0,0588235294117647	0	0	1.02%
16 SYS	JAV\$4MC\$	PIECE#	180	80	1	0	1	1.02%
17 SYS	JAV\$4MC\$	METHOD#	11 80	C11C	0,0909090909090909	0	0	1.02%
18 SYS	JAV\$4MC\$	OBJ#	12 C3033347	C306041B	0,0294117647058824	0	12 02%	

```
select *
from dba_indexes;
```

OWNER	INDEX_NAME	INDEX_TYPE	TABLE_OWNER	TABLE_NAME	TABLE_TYPE	UNIQUENESS	COMPRESSION	PREFIX_LENGTH	TABLESPACE_NAME	INI_TRANS	MAX_TRANS	INITIAL_EXTENT	NEXT_EXTENT	PC
1 XDB	XDB\$ID	FUNCTION-BASED DOMAIN	XDB	XDB\$RESOURCE	TABLE	NONUNIQUE	DISABLED	(null)	(null)	0	0	(null)	(null)	
2 XDB	SC_XDB	FUNCTION-BASED DOMAIN	XDB	XDB\$SECURITYCLASS	TABLE	NONUNIQUE	DISABLED	(null)	(null)	0	0	(null)	(null)	
3 XDB	PRIN_XDB	FUNCTION-BASED DOMAIN	XDB	XDB\$PRINCIPALS	TABLE	NONUNIQUE	DISABLED	(null)	(null)	0	0	(null)	(null)	
4 XDB	XDB\$ACL_XDB	FUNCTION-BASED DOMAIN	XDB	XDB\$ACL	TABLE	NONUNIQUE	DISABLED	(null)	(null)	0	0	(null)	(null)	
5 BDG2308	SYS_C00644462	NORMAL	BDG2308	SECCIONREVISTA	TABLE	UNIQUE	DISABLED	(null)	BDGRADO	2	255	(null)	(null)	
6 BDG2308	SYS_C00644456	NORMAL	BDG2308	NUMEROREVISTA	TABLE	UNIQUE	DISABLED	(null)	BDGRADO	2	255	(null)	(null)	
7 BDG2308	SYS_C00644452	NORMAL	BDG2308	PERIODISTAFREELANCE	TABLE	UNIQUE	DISABLED	(null)	BDGRADO	2	255	(null)	(null)	
8 BDG2308	SYS_C00644383	NORMAL	BDG2308	REVISTA	TABLE	UNIQUE	DISABLED	(null)	BDGRADO	2	255	65536	1048576	
9 BDG2308	SYS_C00644451	NORMAL	BDG2308	PERIODISTAFREELANCE	TABLE	UNIQUE	DISABLED	(null)	BDGRADO	2	255	65536	1048576	
10 BDG2308	SYS_C00644382	NORMAL	BDG2308	REVISTA	TABLE	UNIQUE	DISABLED	(null)	BDGRADO	2	255	65536	1048576	
11 BDG2308	SYS_C00644391	NORMAL	BDG2308	PERIODISTAPLANTILLA	TABLE	UNIQUE	DISABLED	(null)	BDGRADO	2	255	(null)	(null)	
12 BDG2308	SYS_C00644390	NORMAL	BDG2308	PERIODISTAPLANTILLA	TABLE	UNIQUE	DISABLED	(null)	BDGRADO	2	255	(null)	(null)	
13 BDG2101	COLAB_FK_REV_FREELANCE	NORMAL	BDG2101	COLABORACION	TABLE	UNIQUE	DISABLED	(null)	BDGRADO	2	255	(null)	(null)	
14 BDG2101	SYS_C00659540	NORMAL	BDG2101	ARTICULO	TABLE	UNIQUE	DISABLED	(null)	BDGRADO	2	255	(null)	(null)	
15 BDG2101	SECCION_PK_NOMBRE	NORMAL	BDG2101	SECCION_REVISTA	TABLE	UNIQUE	DISABLED	(null)	BDGRADO	2	255	(null)	(null)	
16 BDG2101	NUMERO_PK	NORMAL	BDG2101	NUMERO	TABLE	UNIQUE	DISABLED	(null)	BDGRADO	2	255	(null)	(null)	
17 BDG2101	SYS_C00659521	NORMAL	BDG2101	REVISTA	TABLE	UNIQUE	DISABLED	(null)	BDGRADO	2	255	65536	1048576	
18 BDG2101	SYS_C00659520	NORMAL	BDG2101	REVISTA	TABLE	UNIQUE	DISABLED	(null)	BDGRADO	2	255	65536	1048576	

**8.5.Mostrar con un SELECT los índices definidos sobre las tablas ABDXX_ONG y ABDXX_PROYECTO
(nombre de la tabla, nombre del índice, tipo de índice).**

```
select table_name, index_name, index_type
from dba_indexes
where table_name= 'ABD02_ONG' or table_name='ABD02_PROYECTO';
```

TABLE_NAME	INDEX_NAME	INDEX_TYPE
1 ABD02_PROYECTO	PROYECTO_PK	NORMAL
2 ABD02_ONG	ONG_PK	NORMAL
3 ABD02_ONG	CLAVE_UNIQUE	NORMAL

USERNAME	DEFAULT_TABLESPACE	TEMPORARY_TABLESPACE
1 ABD02	ABD02_DATOS	ABD0_TEMP

```
DROP TABLE ABD02_U;
```

8.6.Activar la autotraza para mostrar el plan de ejecución (sin estadísticas) de las sentencias SELECT que ejecutará en los siguientes ejercicios. Quizá deba desactivarla en ocasiones, por ejemplo para ejecutar sentencias de creación o destrucción de elementos del esquema.

```
set autotrace on explain;
```

8.7 Optimizar consultas. Para ello, a) ejecutar cada consulta indicada; b) mostrar y analizar su plan de ejecución; c) crear los índices necesarios (en el tablespace de índices) para optimizarla, d) volver a mostrar el plan optimizado de la consulta; y e) justificar los resultados.

A) Mejorar el rendimiento de la siguiente consulta, que muestra las ONG de Murcia.

```

SQL | Todas las Filas Recuperadas: 14 en 0,016 segundos
PLAN_TABLE_OUTPUT
1 Plan hash value: 213915877
2
3
4 | Id  | Operation          | Name      | Rows  | Bytes | Cost (%CPU) | Time      | Pstart| Pstop |
5 -----
6 | 0  | SELECT STATEMENT   |           | 1    | 48   | 3   (0) | 00:00:01 |        |        |
7 | 1  |  PARTITION HASH ALL|           | 1    | 48   | 3   (0) | 00:00:01 |        |        |
8 |* 2  |   TABLE ACCESS FULL| ABD02_ONG | 1    | 48   | 3   (0) | 00:00:01 |        |        |
9 -----
10
11 Predicate Information (identified by operation id):
12
13
14 2 - filter("PROVINCIA"='Murcia')

```

Creamos para ello un índice que facilite la búsqueda de las provincias.

```
create index ej8a
on abd02_ong(provincia)
tablespace abd02_indices;
```

Explain plan for

```
SELECT codigo, nombre, email, telf
FROM ABD02_ONG
WHERE provincia='Murcia';
```

```
SELECT plan_table_output
      FROM TABLE(DBMS_XPLAN.DISPLAY());
```

Con esto sería suficiente para optimizar.

Ahora usamos el índice creado a ver que tal la estadística.

```

PLAN_TABLE_OUTPUT
1 Plan hash value: 2232321069
2
3
4 | Id  | Operation          | Name      | Rows  | Bytes | Cost (%CPU) | Time      | Pstart| Pstop |
5 -----
6 | 0  | SELECT STATEMENT   |           | 1    | 48   | 3   (0) | 00:00:01 |        |        |
7 | 1  |  TABLE ACCESS BY GLOBAL INDEX ROWID| ABD02_ONG | 1    | 48   | 3   (0) | 00:00:01 | ROWID | ROWID |
8 |* 2  |   INDEX RANGE SCAN  | EJ8A     | 1    | 1    | 1   (0) | 00:00:01 |        |        |
9 -----
10
11 Predicate Information (identified by operation id):
12
13
14 2 - access("PROVINCIA"='Murcia')

```

Se ve que ahora usa del índice en vez de la tabla, y además el uso de cpu es menor.

Se puede observar como ahora se utiliza el índice creado. Para que el índice fuese interesante había que crearlo, utilizando el campo provincia, que es el que utiliza la sentencia.

B) Optimizar la siguiente consulta, que muestra las ONG de Murcia cuyo campo de acción es 'COOPERACIÓN PARA EL DESARROLLO'.

```
explain plan for
  select codigo, nombre, email, telf
  from abd02_ong
    where provincia='Murcia' and campo='COOPERACIÓN PARA EL
DESARROLLO';
```

```
SELECT plan_table_output
  FROM TABLE(DBMS_XPLAN.DISPLAY());
```

```
PLAN_TABLE_OUTPUT
1 Plan hash value: 2232321069
2
3
4 | Id  | Operation          | Name      | Rows  | Bytes | Cost (%CPU) | Time      | Pstart| Pstop |
5 -----
6 | 0  | SELECT STATEMENT   |           |       |       |            | 00:00:01 |        |        |
7 |* 1 |  TABLE ACCESS BY GLOBAL INDEX ROWID| ABD02_ONG | 1    | 65   | 3 (0)    | 00:00:01 | ROWID | ROWID |
8 |* 2 |  INDEX RANGE SCAN   | EJ8A     | 1    | 1    | 1 (0)    | 00:00:01 |        |        |
9
10
11 Predicate Information (identified by operation id):
12
13
14   1 - filter("CAMPO"="COOPERACIÓN PARA EL DESARROLLO")
15   2 - access("PROVINCIA"="Murcia")
```

```
create index ej8b
  on abd02_ong(provincia, campo)
  tablespace abd02_indices ;

explain plan for
  select codigo, nombre, email, telf
  from abd02_ong
    where provincia='Murcia' and campo='COOPERACIÓN PARA EL
DESARROLLO';
```

```
SELECT plan_table_output
  FROM TABLE(DBMS_XPLAN.DISPLAY());
```

```

PLAN_TABLE_OUTPUT
1 Plan hash value: 2079189871
2
3
4 | Id  | Operation          | Name      | Rows  | Bytes | Cost (%CPU) | Time      | Pstart| Pstop |
5 -----
6 | 0  | SELECT STATEMENT   |           | 1    | 65   | 2   (0) | 00:00:01 |        |        |
7 | 1  |  TABLE ACCESS BY GLOBAL INDEX ROWID| ABD02_ONG | 1    | 65   | 2   (0) | 00:00:01 | ROWID | ROWID |
8 |* 2 |  INDEX RANGE SCAN   | EJ8B     | 1    | 1    | 1   (0) | 00:00:01 |        |        |
9
10
11 Predicate Information (identified by operation id):
12
13
14 2 - access("PROVINCIA"='Murcia' AND "CAMPO"='COOPERACIÓN PARA EL DESARROLLO')

```

La idea es bastante similar, a la del apartado anterior se crea un índice, utilizando los campos clave de la búsqueda, en este caso provincia y por campo de actividad. Y de ese modo, se consigue que se optimice.

C) Mejorar el rendimiento de esta consulta, que muestra las ONG de Murcia cuyo campo de acción es 'COOPERACIÓN PARA EL DESARROLLO' y que tienen proyectos en ETIOPÍA.

```

EXPLAIN PLAN FOR
SELECT codigo, nombre, email, telf
FROM ABD02_ONG O
WHERE O.provincia='Murcia' AND O.campo='COOPERACIÓN PARA EL
DESARROLLO'
AND O.codigo IN (SELECT ong FROM ABD02_PROYECTO P
WHERE P.pais='ETIOPIA');

```

```

SELECT plan_table_output
  FROM TABLE(DBMS_XPLAN.DISPLAY());

```

PLAN_TABLE_OUTPUT										
1 Plan hash value: 952432672										
2										
3										
4 Id Operation Name Rows Bytes Cost (%CPU) Time Pstart Pstop										
5										
6 0 SELECT STATEMENT 1 75 5 (0) 00:00:01										
7 1 NESTED LOOPS SEMI 1 75 5 (0) 00:00:01										
8 2 TABLE ACCESS BY GLOBAL INDEX ROWID ABD02_ONG 1 65 2 (0) 00:00:01 ROWID ROWID										
9 * 3 INDEX RANGE SCAN EJOB 1 1 (0) 00:00:01										
10 * 4 TABLE ACCESS CLUSTER ABD02_PROYECTO 3 30 3 (0) 00:00:01										
11 * 5 INDEX RANGE SCAN ID02_PAP 1 1 (0) 00:00:01										
12										
13										
14 Predicate Information (identified by operation id):										
15										

```
create index ej8c
  on abd02_proyecto(ong,pais)
  tablespace abd02_indices;
```

```
EXPLAIN PLAN FOR
SELECT codigo, nombre, email, telf
FROM ABD02_ONG O
WHERE O.provincia='Murcia' AND O.campo='COOPERACIÓN PARA EL
DESARROLLO'
AND O.codigo IN (SELECT ong FROM ABD02_PROYECTO P
WHERE P.pais='ETIOPIA');
```

```
SELECT plan_table_output
  FROM TABLE(DBMS_XPLAN.DISPLAY());
```

PLAN_TABLE_OUTPUT										
1	Plan hash value: 2164013755									
2										
3										
4	Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time	Pstart	Pstop	
5	<hr/>									
6	0	SELECT STATEMENT		1	75	2 (0)	00:00:01			
7	1	NESTED LOOPS SEMI		1	75	2 (0)	00:00:01			
8	2	TABLE ACCESS BY GLOBAL INDEX ROWID	ABD02_ONG	1	65	2 (0)	00:00:01	ROWID	ROWID	
9	*	3	INDEX RANGE SCAN	EJ8B	1	1 (0)	00:00:01			
10	*	4	INDEX RANGE SCAN	EJ8C	3	30 (0)	00:00:01			
11	<hr/>									
12										
13	Predicate Information (identified by operation id):									
14	<hr/>									
15										

Como puede verse, se utiliza el índice creado en este apartado, cuyos campos son ong y país, ya que en esta última consulta, parte de la misma, busca las ong en un país concreto, de ese modo se puede obtener rápidamente las ong del país concreto. Y luego se utilizará el índice del apartado anterior, que como se dijo utiliza los campos provincia y campo. Aunque para, podría decirse, segunda búsqueda se partiría de las ong que cumplieron la primera condición. Por tanto, se utilizan dos índices, el del apartado anterior y el creado en este apartado.

8.8.Asegurarse de que la autotraza queda desactivada.

```
set autotrace off;
Rastreo Automático Desactivado
```

8.9.Eliminar los índices creados en este ejercicio.

1	TABLE_NAME	2	INDEX_NAME	3	INDEX_TYPE
1	ABD02_PROYECTO	PROYECTO_PK	NORMAL		
2	ABD02_PROYECTO	EJ8C	NORMAL		
3	ABD02_ONG	ONG_PK	NORMAL		
4	ABD02_ONG	EJ8A	NORMAL		
5	ABD02_ONG	CLAVE_UNIQUE	NORMAL		
6	ABD02_ONG	EJ8B	NORMAL		

Ahora eliminamos los que hemos creado para este ejercicio

```
drop index ej8a;
drop index ej8b;
drop index ej8c;
```

8.10. Volver a mostrar con un SELECT los índices definidos sobre las tablas ABDXX_ONG y ABDXX_PROYECTO (nombre de la tabla, nombre del índice, tipo de índice), y comprobar que han quedado los que había antes del ejercicio 8.

```
select table_name, index_name, index_type  
from dba_indexes  
where table_name= 'ABD02_ONG' or table_name='ABD02_PROYECTO';
```

	TABLE_NAME	INDEX_NAME	INDEX_TYPE
1	ABD02_PROYECTO	PROYECTO_PK	NORMAL
2	ABD02_ONG	ONG_PK	NORMAL
3	ABD02_ONG	CLAVE_UNIQUE	NORMAL

9. USUARIOS, PERFILES, PERMISOS Y ROLES

9.1. Conectar como usuario ABDXX, crear un nuevo usuario autenticado por la base de datos, con las siguientes especificaciones:

ABDXX_OPERARIO: Su clave inicial es 0P3R4R102. Sin cuota en su tablespace por defecto ABDXX_DATOS, con espacio ilimitado en ABDXX_INDICES, y cuota 0 en el resto. Su tablespace temporal ha de ser uno de los tablespaces temporales ya definidos en la base de datos.

```
CREATE USER ABD02_OPERARIO
  IDENTIFIED BY "0P3R4R10"
  DEFAULT TABLESPACE ABD02_DATOS
  TEMPORARY TABLESPACE ADBD_TEMP
  QUOTA UNLIMITED ON ABD02_INDICES;
```

user ABD02_OPERARIO creado.

9.2. Mostrar con un SELECT los tablespaces por defecto y temporales de ABDXX y ABDXX_OPERARIO (usuario, nombre de su tablespace por defecto, nombre de su tablespace temporal).

```
SELECT USERNAME, DEFAULT_TABLESPACE, TEMPORARY_TABLESPACE
FROM DBA_USERS
WHERE (username= 'ABD02' or username='ABD02_OPERARIO');
```

USERNAME	DEFAULT_TABLESPACE	TEMPORARY_TABLESPACE
1 ABD02_OPERARIO	ABD02_DATOS	ADBD_TEMP
2 ABD02	ABD02_DATOS	ADBD_TEMP

9.3.Mostrar con un SELECT las cuotas de espacio en los tablespaces ABDXX_DATOS y ABDXX_INDICES de ABDXX y de ABDXX_OPERARIO (usuario, tablespace, bytes, maxbytes).

```
SELECT USERNAME, TABLESPACE_NAME, BYTES, MAX_BYTES
FROM DBA_TS_QUOTAS
WHERE USERNAME= 'ABD02' OR USERNAME='ABD02_OPERARIO';
```

USERNAME	TABLESPACE_NAME	BYTES	MAX_BYTES
1 ABD02	ABD02_DATOS	0	-1
2 ABD02_OPERARIO	ABD02_INDICES	0	-1
3 ABD02	ABD02_DATOS	557056	-1
4 ABD02	ABD02_INDICES	778240	-1

9.4.Permitir que el usuario ABDXX_OPERARIO pueda ocupar hasta 2M de espacio en el tablespace ABDXX_DATOS. Y volver a mostrar las cuotas para ambos usuarios.

```
ALTER USER ABD02_OPERARIO quota 2M on ABD02_DATOS;
user ABD02_OPERARIO altered.
```

```
SELECT USERNAME, TABLESPACE_NAME, BYTES, MAX_BYTES
FROM DBA_TS_QUOTAS
WHERE USERNAME= 'ABD02' OR USERNAME='ABD02_OPERARIO';
```

USERNAME	TABLESPACE_NAME	BYTES	MAX_BYTES
1 ABD02	ABD02_DATOS	0	-1
2 ABD02_OPERARIO	ABD02_INDICES	0	-1
3 ABD02	ABD02_DATOS	557056	-1
4 ABD02_OPERARIO	ABD02_DATOS	0	2097152
5 ABD02	ABD02_INDICES	778240	-1

9.5.Crear un perfil denominado ABDXX_PERFIL_OPERARIO, que garanticen estas especificaciones:

- Las contraseñas de ABDXX_OPERARIO expirarán a los 30 días.
- ABDXX_OPERARIO puede iniciar simultáneamente hasta 2 conexiones a la base de datos.

- ABDXX_OPERARIO tiene limitado a 10 minutos el tiempo de inactividad de sus sesiones (pasado ese tiempo se cerrará automáticamente la sesión).
- La cuenta de ABDXX_OPERARIO se bloqueará durante 3 días en el caso de superar los 4 intentos consecutivos fallidos de conexión.

```
CREATE PROFILE ABD02_PERFIL_OPERARIO LIMIT
SESSIONS_PER_USER 2
IDLE_TIME 10
FAILED_LOGIN_ATTEMPTS 4
PASSWORD_LIFE_TIME 30
PASSWORD_LOCK_TIME 3;
```

profile ABD02_PERFIL_OPERARIO creado.

9.6. Asignar el perfil al usuario creado.

```
ALTER USER ABD02_OPERARIO
PROFILE ABD02_PERFIL_OPERARIO;
user ABD02_OPERARIO alterado.
```

9.7. Mostrar con un SELECT el perfil creado en la base de datos y qué límites de recursos establece (nombre del perfil, nombre del recurso limitado, tipo del recurso, y límite). Se recomienda que muestre sólo los límites fijados, y no todos.

```
SELECT *
from dba_profiles
where profile= 'ABD02_PERFIL_OPERARIO' and not limit='DEFAULT';
```

PROFILE	RESOURCE_NAME	RESOURCE_TYPE	LIMIT
1 ABD02_PERFIL_OPERARIO	SESSIONS_PER_USER	KERNEL	2
2 ABD02_PERFIL_OPERARIO	IDLE_TIME	KERNEL	10
3 ABD02_PERFIL_OPERARIO	FAILED_LOGIN_ATTEMPTS	PASSWORD	4
4 ABD02_PERFIL_OPERARIO	PASSWORD_LIFE_TIME	PASSWORD	30
5 ABD02_PERFIL_OPERARIO	PASSWORD_LOCK_TIME	PASSWORD	3

9.8. Conceder a ABDXX_OPERARIO el permiso para consultar la tabla ABDXX_ONG.

```
GRANT SELECT ON ABD02_ONG
TO ABD02_OPERARIO;
```

GRANT correcto.

9.9.Crear el rol denominado ABDXX_ROL_OPERARIO. Conceder a dicho rol los privilegios adecuados sobre elementos del esquema (indicados a continuación), y asignar el rol al usuarios ABDXX_OPERARIO, de forma que se cumplan los siguientes requisitos:

```
CREATE ROLE ABD02_ROL_OPERARIO NOT IDENTIFIED;  
role ABD02_ROL_OPERARIO creado.
```

- **ABDXX_OPERARIO puede consultar cualquier tabla del esquema de las ONG. Salvo la columna sueldo de TRABAJADOR.**

```
GRANT select  
ON ABD02_accion  
TO ABD02_ROL_OPERARIO;
```

```
GRANT select  
ON ABD02_colaboracion  
TO ABD02_ROL_OPERARIO;
```

```
GRANT select  
ON ABD02_ong  
TO ABD02_ROL_OPERARIO;
```

```
GRANT select  
ON ABD02_participacion  
TO ABD02_ROL_OPERARIO;
```

```
GRANT select  
ON ABD02_proyecto  
TO ABD02_ROL_OPERARIO;
```

```
GRANT select  
ON ABD02_socio  
TO ABD02_ROL_OPERARIO;
```

Como no se pueden indicar columnas junto con el privilegio Select. La solución, para que solo puede consultar las columnas de la tabla trabajador que se indica. Es crear una vista, que contenga todas las columnas, menos la columna “sueldo”.

```
CREATE VIEW vistaejer99 AS  
select dni, nombre, ong, fechaingreso, esvoluntario,  
nacimiento, profesion, horas  
from abd02_trabajador;
```

```
GRANT select  
ON vistaejer99
```

```
TO ABD02_ROL_OPERARIO;
```

```
GRANT correcto.
```

```
view VISTAEJER99 creado.
```

```
GRANT correcto.
```

- Puede insertar filas en cualquier tabla, salvo en ABDXX_SOCIO.

```
GRANT insert
```

```
ON ABD02_accion
```

```
TO ABD02_ROL_OPERARIO;
```

```
GRANT insert
```

```
ON ABD02_colaboracion
```

```
TO ABD02_ROL_OPERARIO;
```

```
GRANT insert
```

```
ON ABD02_ong
```

```
TO ABD02_ROL_OPERARIO;
```

```
GRANT insert
```

```
ON ABD02_participacion
```

```
TO ABD02_ROL_OPERARIO;
```

```
GRANT insert
```

```
ON ABD02_proyecto
```

```
TO ABD02_ROL_OPERARIO;
```

```
GRANT insert
```

```
ON ABD02_trabajador
```

```
TO ABD02_ROL_OPERARIO;
```

```
GRANT correcto.
```

```
GRANT correcto.
```

```
GRANT correcto.  
GRANT correcto.  
GRANT correcto.  
GRANT correcto.
```

- Puede actualizar (cambiar valores de) filas en cualquier tabla del esquema, excepto para las columnas fechaalta y cuota de la tabla ABDXX_COLABORACION.

```
GRANT UPDATE  
ON ABD02_accion  
TO ABD02_ROL_OPERARIO;  
  
GRANT UPDATE (ong, socio)  
ON ABD02_colaboracion  
TO ABD02_ROL_OPERARIO;
```

```
GRANT UPDATE  
ON ABD02_ong  
TO ABD02_ROL_OPERARIO;  
  
GRANT UPDATE  
ON ABD02_participacion  
TO ABD02_ROL_OPERARIO;
```

```
GRANT UPDATE  
ON ABD02_proyecto  
TO ABD02_ROL_OPERARIO;
```

```
GRANT UPDATE  
ON ABD02_socio  
TO ABD02_ROL_OPERARIO;
```

```
GRANT UPDATE  
ON ABD02_trabajador  
TO ABD02_ROL_OPERARIO;
```

```
GRANT correcto.  
GRANT correcto.  
GRANT correcto.  
GRANT correcto.  
GRANT correcto.  
GRANT correcto.
```

GRANT correcto.

- Puede eliminar filas de cualquier tabla salvo para la tabla ABDXX_ONG. Primero se crea el rol.

```
GRANT delete  
ON ABD02_accion  
TO ABD02_ROL_OPERARIO;
```

```
GRANT delete  
ON ABD02_colaboracion  
TO ABD02_ROL_OPERARIO;
```

```
GRANT delete  
ON ABD02_participacion  
TO ABD02_ROL_OPERARIO;
```

```
GRANT delete  
ON ABD02_proyecto  
TO ABD02_ROL_OPERARIO;
```

```
GRANT delete  
ON ABD02_socio  
TO ABD02_ROL_OPERARIO;
```

```
GRANT delete  
ON ABD02_trabajador  
TO ABD02_ROL_OPERARIO;
```

```
GRANT correcto.  
GRANT correcto.  
GRANT correcto.  
GRANT correcto.  
GRANT correcto.  
GRANT correcto.
```

Se asigna el rol al usuario:

```
grant ABD02_ROL_OPERARIO to ABD02_OPERARIO;
```

```
grant correcto.
```

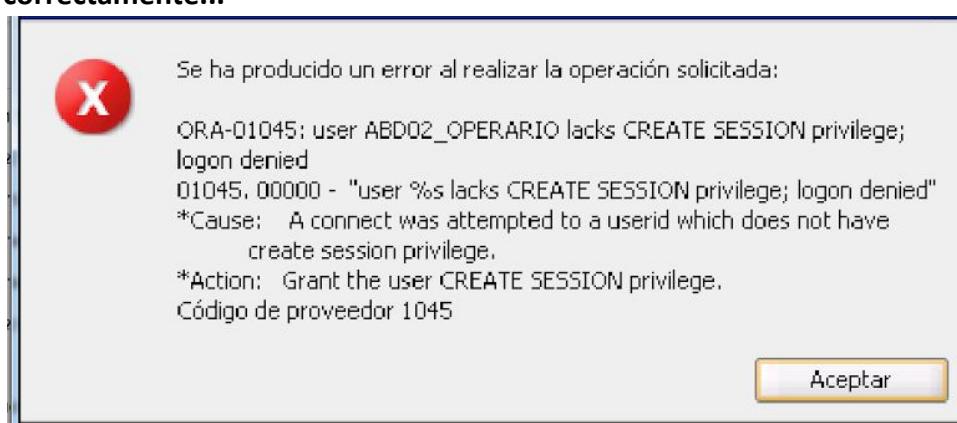
Como puede verse, lo que hay ido haciendo es ir añadiendo para cada tabla concreto los privilegios que se indican en cada momento.

9.10. Mostrar con un SELECT los permisos sobre tablas concedidos a roles (rol, tabla, privilegio), de forma ordenada por rol, tabla y privilegio.

```
select ROLE, TABLE_NAME, PRIVILEGE  
FROM ROLE_TAB_PRIVS;
```

ROLE	TABLE_NAME	PRIVILEGE
1 ABD02_ROL_OPERARIO	ABD02_ACCION	INSERT
2 ABD02_ROL_OPERARIO	ABD02_COLABORACION	UPDATE
3 ABD02_ROL_OPERARIO	ABD02_COLABORACION	INSERT
4 ABD02_ROL_OPERARIO	ABD02_SOCIO	SELECT
5 ABD02_ROL_OPERARIO	ABD02_COLABORACION	UPDATE
6 ABD02_ROL_OPERARIO	ABD02_PARTICIPACION	SELECT
7 ABD02_ROL_OPERARIO	ABD02_ACCION	DELETE
8 ABD02_ROL_OPERARIO	ABD02_ONG	INSERT
9 ABD02_ROL_OPERARIO	ABD02_TRABAJADOR	INSERT
10 ABD02_ROL_OPERARIO	ABD02_ONG	SELECT
11 ABD02_ROL_OPERARIO	ABD02_SOCIO	UPDATE
12 ABD02_ROL_OPERARIO	ABD02_ACCION	SELECT
13 ABD02_ROL_OPERARIO	ABD02_PARTICIPACION	INSERT
14 ABD02_ROL_OPERARIO	ABD02_SOCIO	DELETE
15 ABD02_ROL_OPERARIO	VISTAEJER99	SELECT
16 ABD02_ROL_OPERARIO	ABD02_COLABORACION	DELETE
17 ABD02_ROL_OPERARIO	ABD02_COLABORACION	SELECT
18 ABD02_ROL_OPERARIO	ABD02_PROYECTO	UPDATE
19 ABD02_ROL_OPERARIO	ABD02_PROYECTO	INSERT
20 ABD02_ROL_OPERARIO	ABD02_PROYECTO	SELECT
21 ABD02_ROL_OPERARIO	ABD02_TRABAJADOR	DELETE
22 ABD02_ROL_OPERARIO	ABD02_ACCION	UPDATE
23 ABD02_ROL_OPERARIO	ABD02_TRABAJADOR	UPDATE
24 ABD02_ROL_OPERARIO	ABD02_ONG	UPDATE
25 ABD02_ROL_OPERARIO	ABD02_PARTICIPACION	DELETE
26 ABD02_ROL_OPERARIO	ABD02_PARTICIPACION	UPDATE
27 ABD02_ROL_OPERARIO	ABD02_PROYECTO	DELETE

9.11. Conectar a la base de datos con el usuario ABDXX_OPERARIO. Solucione los posibles problemas que puedan surgir, y explique lo que ha pasado. Una vez que esté conectado correctamente...



Para solucionar el problema, que podía surgir.

```
GRANT CREATE SESSION TO ABD02_OPERARIO;
```

GRANT correcto.

Visualizar los roles activos que tiene el usuario en esta sesión (nombre del rol).

```
SELECT ROLE  
FROM SESSION_ROLES;
```

ROLE
ABD02_ROL_OPERARIO

Mostrar los privilegios de objeto y de sistema obtenidos directamente y no a través de roles, que posee el usuario (nombre del privilegio, tabla [o null, si es de sistema]).

```
SELECT      a.privilege      Privilegio_Sistema,      b.privilege  
Privilegio_Objeto, b.Table_name  
FROM (select privilege from USER_SYS_PRIVS) a  
FULL OUTER JOIN (select privilege, table_name from USER_TAB_PRIVS) b  
ON a.privilege = b.privilege;
```

PRIVILEGIO_SISTEMA	PRIVILEGIO_OBJETO	TABLE_NAME
1 (null)	SELECT	ABD02_ONG
2 CREATE SESSION	(null)	(null)

Listar con SELECT el nombre de las ONG con sede en Murcia, su email y el número de socios colaboradores que tienen, siempre que tengan al menos 3 colaboradores, ordenado por nombre de ONG. (columnas: nombre, email, total_colaboradores). Mostrar el resultado tras la ejecución y comentar.

```
SELECT nombre, email, total_colaboradores  
FROM ABD02.ABD02_ONG JOIN  
      (SELECT ong , COUNT(*) total_colaboradores  
       FROM ABD02.ABD02_COLABORACION  
       GROUP BY ong)  
      on codigo = ONG  
WHERE provincia = 'Murcia' and total_colaboradores > = 3  
order by nombre;
```

NOMBRE	EMAIL	TOTAL_COLABORADORES
Copaelve	Copaelve@hotmail.es	3

Como hay privilegios, cuando se obtiene la sentencia adecuada se consigue la información sin problemas.

Obtener con SELECT el nombre de cada trabajador, profesión y sueldo siempre que haya participado en más de 3 acciones distintas. (columnas: nombre, profesión, sueldo). Listar por orden alfabético de sus nombres. Mostrar lo que aparece en pantalla tras la ejecución y explicar.

```
SELECT nombre, profesion, sueldo
FROM ABD02.ABD02_trabajador JOIN
     (SELECT trabajador, COUNT(*) participa_accion
      FROM ABD02.ABD02_participacion
      GROUP BY trabajador)
      on dni =
trabajador
WHERE participa_accion > = 3
order by nombre;
```

ORA-01031: privilegios insuficientes
01031. 00000 - "Insufficient privileges"
*Cause: An attempt was made to change the current username or password without the appropriate privilege. This error also occurs if attempting to install a database without the necessary operating system privileges.
When Trusted Oracle is configured in DBMS MAC, this error may occur if the user was granted the necessary privilege at a higher label than the current login.
*Action: Ask the database administrator to perform the operation or grant the required privileges.
For Trusted Oracle users getting this error although granted the appropriate privilege at a higher label, ask the database administrator to regrant the privilege at the appropriate label.
Error en la linea: 48, columna: 11

El problema surge, porque el usuario 'ABD02_OPERARIO' no tiene privilegios suficientes sobre la columna sueldo de la tabla "TRABAJADOR", y por tanto no puede consultarla. Como alternativa, lo que podría hacerse es hacer la misma consulta anterior pero sobre la vista (esta vista tiene todas las columnas menos la de sueldo). Sobre esta vista se puede realizar la consulta, porque hay privilegios, pero no se podría consultar la columna sueldo.

Un ejemplo de lo anterior sería:

```

SELECT nombre, profesion
FROM ABD02.vistaejer99 JOIN
     (SELECT trabajador, COUNT(*) participa_accion
      FROM ABD02.ABD02_participacion
      GROUP BY trabajador)
      on dni =
trabajador
WHERE participa_accion > = 3
order by nombre;

```

NOMBRE	PROFESION
1 Eusebio	Escritora

Modificar la cuota de un socio (elija uno) de cierta ONG (elija una) que desea aportar 50€ más al mes a partir de este momento. Mostrar lo que aparece en pantalla tras la ejecución y razonar.

```

UPDATE abd02.ABD02_COLABORACION SET
abd02.ABD02_COLABORACION.cuota= abd02.ABD02_COLABORACION.cuota + 50
WHERE abd02.ABD02_COLABORACION.org = '8' AND
abd02.ABD02_COLABORACION.socio= '10101010A';

```

Error que empieza en la linea 8 del comando:
UPDATE abd02.ABD02_COLABORACION SET
abd02.ABD02_COLABORACION.cuota= abd02.ABD02_COLABORACION.cuota + 50
WHERE abd02.ABD02_COLABORACION.org = '8' AND abd02.ABD02_COLABORACION.socio= '10101010A'
Error en la linea de comandos:8 Columna:13
Informe de error:
Error SQL: ORA-01031: privilegios insuficientes
01031. 00000 - "insufficient privileges"
*Cause: An attempt was made to change the current username or password
without the appropriate privilege. This error also occurs if
attempting to install a database without the necessary operating
system privileges.
When Trusted Oracle is configured in DBMS MAC, this error may occur
if the user was granted the necessary privilege at a higher label
than the current login.
*Action: Ask the database administrator to perform the operation or grant
the required privileges.
For Trusted Oracle users getting this error although granted the
appropriate privilege at a higher label, ask the database
administrator to regrant the privilege at the appropriate label.

No se permite porque no hay privilegios adecuados para realizar esa operación.

Cambiar el teléfono de una ONG (elija una cualquiera). Mostrar lo que aparece en pantalla tras la ejecución y comentar.

```
UPDATE ABD02.ABD02_ONG SET  
ABD02.ABD02_ONG.telf='968454545'  
WHERE ABD02.ABD02_ONG.codigo='8';
```

1 filas actualizadas.

```
SELECT *  
FROM ABD02.ABD02_ONG  
WHERE CODIGO='8';
```

SQL Todas las Filas Recuperadas: 1 en 0,001 segundos						
	CODIGO	NOMBRE	EMAIL	TELF	PROVINCIA	CAMPO
1	8	Lucha Humana	luhu@hotmail.es	968454545	Pontevedra	derechos humanos 55555555A

Como se dieron los privilegios adecuados, se puede realizar sin problemas.

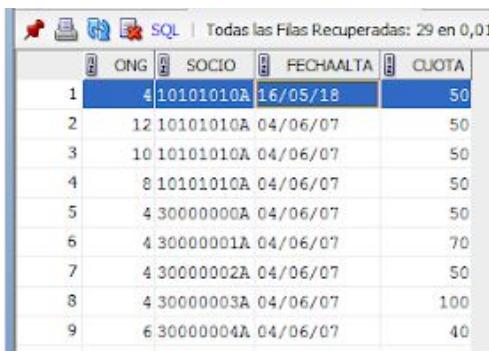
Registrar la nueva colaboración de un socio (elija uno) con una ONG (elija una) indicando la fecha actual (del sistema) como fecha de alta y una cuota consecuente con las cuotas que el mismo socio ya aporta a otras ONG. Mostrar lo que aparece en pantalla tras la ejecución y comentar.

Como puede verse, se utiliza: "Select sysdate from dual", para obtener la fecha actual del momento de la inserción. De ese modo, cuando la nueva colaboración tendrá esa fecha. Para que la cuota sea consecuente con el resto de aportaciones realizadas por ese socio al resto de ong's, se decidió obtener la media de todas las aportaciones realizadas por ese socio. Para ello, se utilizó:

```
select AVG(cuota)  
from abd02.abd02_colaboracion  
where abd02.abd02_colaboracion.socio = '10101010A';
```

Así, la cuota que se introduce es consecuente con sus demás aportaciones.

```
insert into abd02.abd02_colaboracion values (4,'10101010A',(select  
sysdate from dual),( select AVG(cuota)  
from abd02.abd02_colaboracion  
where abd02.abd02_colaboracion.socio = '10101010A'));
```



	ONG	SOCIO	FECHAALTA	CUOTA
1	4 10101010A	16/05/18		50
2	12 10101010A	04/06/07		50
3	10 10101010A	04/06/07		50
4	8 10101010A	04/06/07		50
5	4 30000000A	04/06/07		50
6	4 300000001A	04/06/07		70
7	4 300000002A	04/06/07		50
8	4 300000003A	04/06/07		100
9	6 300000004A	04/06/07		40

Si se pueden insertar nuevas filas sobre la tabla colaboración, la única que tenía restricciones era la de socio.

9.12. Por último, eliminar de la base de datos el usuario, perfil y rol con los que ha trabajado en este ejercicio 9.

Para poder eliminar el usuario debemos de cerrar sesión, ya que no es posible hacer esto estando el usuario conectado.

```
drop role ABD02_ROL_OPERARIO CASCADE;
drop profile ABD02_PERFIL_OPERARIO CASCADE ;
drop user ABD02_OPERARIO cascade;
```

```
role ABD02_ROL_OPERARIO borrado.
profile ABD02_PERFIL_OPERARIO borrado.
user ABD02_OPERARIO borrado.
```