

Universidad Complutense de Madrid  
Facultad de Informática  
Máster en Ingeniería Informática - Administración de Bases de Datos

---

## **Práctica 4**

---

20 de octubre de 2020

Daniel Bastarrica Lacalle  
Jose Javier Cortés Tejada

## Índice

<b>1. Creación básica de un usuario de la base de datos.</b>	<b>3</b>
1.1. Ejercicio 1 . . . . .	3
<b>2. Consulta de los <i>tablespaces</i> y <i>datafiles</i> de la base de datos.</b>	<b>4</b>
2.1. Ejercicio 2 . . . . .	4
2.2. Ejercicio 3 . . . . .	4
2.3. Ejercicio 4 . . . . .	5
2.4. Ejercicio 5 . . . . .	6
2.5. Ejercicio 6 . . . . .	7
2.6. Ejercicio 8 . . . . .	7
2.7. Ejercicio 9 . . . . .	8
2.8. Ejercicio 10 . . . . .	8
<b>3. Otras estructuras de almacenamiento.</b>	<b>9</b>
3.1. Ejercicio 11 . . . . .	9
3.2. Ejercicio 12 . . . . .	9
3.3. Ejercicio 13 . . . . .	10
3.4. Ejercicio 14 . . . . .	10
3.5. Ejercicio 15 . . . . .	10
<b>4. Manejo de las estructuras de datos en memoria</b>	<b>11</b>
4.1. Ejercicio 16 . . . . .	11

## 1. Creación básica de un usuario de la base de datos.

### 1.1. Ejercicio 1

Indica en la memoria del ejercicio las instrucciones SQL DCL que deben ejecutarse (las que SQLDeveloper genera automáticamente).

---

```
1  -- USER SQL
2  CREATE USER usuario1 IDENTIFIED BY "usuario1"
3  DEFAULT TABLESPACE "USERS"
4  TEMPORARY TABLESPACE "TEMP";
5
6  -- QUOTAS
7  ALTER USER usuario1 QUOTA 100M ON USERS;
8
9  -- ROLES
10 GRANT "CONNECT" TO usuario1 ;
11
12 -- SYSTEM PRIVILEGES
13 GRANT CREATE TRIGGER TO usuario1 ;
14 GRANT ALTER SESSION TO usuario1 ;
15 GRANT CREATE MATERIALIZED VIEW TO usuario1 ;
16 GRANT CREATE VIEW TO usuario1 ;
17 GRANT CREATE TABLE TO usuario1 ;
18 GRANT CREATE SEQUENCE TO usuario1 ;
19 GRANT CREATE PROCEDURE TO usuario1 ;
```

---

Esto consulta dota al usuario de:

- permiso para conectarse.
- privilegios para alterar la sesión.
- privilegios para crear vistas, tablas, triggers, secuencias, procedimientos y vistas materializadas en su propio *schema*

## 2. Consulta de los *tablespaces* y *datafiles* de la base de datos.

### 2.1. Ejercicio 2

Indica en la memoria del ejercicio el nombre de los *tablespaces* de la instalación de Oracle y los que están habilitados para realizar copia de seguridad.

	TS#	NAME	INCLUDED_IN_DATABASE_BACKUP	BIGFILE	FLASHBACK_ON	ENCRYPT_IN_BACKUP
1	0	SYSTEM	YES	NO	YES	(null)
2	1	SYSAUX	YES	NO	YES	(null)
3	2	UNDOTBS1	YES	NO	YES	(null)
4	4	USERS	YES	NO	YES	(null)
5	3	TEMP	NO	NO	YES	(null)

Figura 1: *tablespaces* por defecto en la instalación de Oracle

En la figura 1 tenemos listados los *tablespaces* que vienen por defecto en la instalación de Oracle. Tenemos un total de 5 *tablespaces*, todos ellos habilitados para copia de seguridad salvo *TEMP*.

### 2.2. Ejercicio 3

Entra en el usuario *oracle* de la máquina virtual y verifica los nombres y tamaños de los ficheros del SO que corresponden a los *datafiles*. Incluye el listado del directorio (*ls -l*) en la memoria del ejercicio.

En la figura 2 tenemos el resultado de una consulta *SQL* para ver los datos de los *datafiles* y de donde obtenemos el *path* donde se ubican los *datafiles*. A continuación tenemos la ejecución del comando *ls -l* que pide el enunciado sobre el *path* ya mencionado.

	STATUS	NAME	BYTES
1	SYSTEM	/u01/app/oracle/oradata/orcl/system01.dbf	713031680
2	ONLINE	/u01/app/oracle/oradata/orcl/sysaux01.dbf	545259520
3	ONLINE	/u01/app/oracle/oradata/orcl/undotbs01.dbf	94371840
4	ONLINE	/u01/app/oracle/oradata/orcl/users01.dbf	5242880

Figura 2: Nombre y tamaño de los *datafile* del sistema

```

1 oracle@ubuntu32vb:/u01/app/oracle/oradata/orcl$ ls -l
2 total 1492276
3 -rw-r----- 1 oracle oinstall 9748480 oct 16 17:28 control01.ctl
4 -rw-r----- 1 oracle oinstall 52429312 oct 16 17:27 redo01.log
5 -rw-r----- 1 oracle oinstall 52429312 oct 16 16:57 redo02.log
6 -rw-r----- 1 oracle oinstall 52429312 oct 16 16:57 redo03.log
7 -rw-r----- 1 oracle oinstall 545267712 oct 16 17:27 sysaux01.dbf
8 -rw-r----- 1 oracle oinstall 713039872 oct 16 17:26 system01.dbf
9 -rw-r----- 1 oracle oinstall 30416896 oct 16 17:06 temp01.dbf

```

```

10 -rw-r----- 1 oracle oinstall 94380032 oct 16 17:26 undotbs01.dbf
11 -rw-r----- 1 oracle oinstall 5251072 oct 16 17:15 users01.dbf

```

## 2.3. Ejercicio 4

Utiliza la siguiente consulta para ver los segmentos de un *tablespace*

```
SELECT * FROM dba_segments WHERE TABLESPACE_NAME = 'USERS';
```

Localiza los segmentos de los que es propietario el usuario *usuario1* e indica en la memoria del ejercicio los nombres y tipos de segmento que aparecen en la lista. Hay muchos tipos de segmento en una base de datos. Investiga las columnas de la vista *dba\_segments* para crear una consulta que muestre todos los tipos de segmento de todos los *tablespaces* de la base de datos. Incluye la consulta y el resultado en la memoria del ejercicio.

```

1 SELECT * FROM dba_segments WHERE TABLESPACE_NAME = 'USERS' and owner = 'USUARIO1';

```

	OWNER	SEGMENT_NAME	SEGMENT_TYPE
1	USUARIO1	MITABLA	TABLE
2	USUARIO1	ZIPCODES	TABLE
3	USUARIO1	EMPLOYEES	TABLE
4	USUARIO1	PARTS	TABLE
5	USUARIO1	CUSTOMERS	TABLE
6	USUARIO1	ORDERS	TABLE
7	USUARIO1	ODETAILS	TABLE
8	USUARIO1	CATALOGE	TABLE
9	USUARIO1	STUDENTS	TABLE
10	USUARIO1	COURSES	TABLE
11	USUARIO1	COMPONENTS	TABLE
12	USUARIO1	ENROLLS	TABLE
13	USUARIO1	SCORES	TABLE
14	USUARIO1	SYS_C0010832	INDEX
15	USUARIO1	SYS_C0010834	INDEX
16	USUARIO1	SYS_C0010839	INDEX
17	USUARIO1	SYS_C0010841	INDEX
18	USUARIO1	SYS_C0010844	INDEX
19	USUARIO1	SYS_C0010850	INDEX
20	USUARIO1	SYS_C0010890	INDEX
21	USUARIO1	SYS_C0010893	INDEX
22	USUARIO1	SYS_C0010901	INDEX
23	USUARIO1	SYS_C0010909	INDEX
24	USUARIO1	SYS_C0010914	INDEX
25	USUARIO1	SYS_C0010922	INDEX

**Figura 3:** Resultado de la consulta que muestra el nombre y el tipo de segmento de los que es propietario el usuario *usuario1*.

---

```
1 SELECT distinct segment_type FROM dba_segments;
```

---

	SEGMENT_TYPE
1	LOBINDEX
2	INDEX PARTITION
3	TABLE PARTITION
4	NESTED TABLE
5	ROLLBACK
6	LOB PARTITION
7	LOBSEGMENT
8	INDEX
9	TABLE
10	CLUSTER
11	TYPE2 UNDO

**Figura 4:** Resultado de la consulta que muestra todos los tipos de segmentos de todos los *tablespaces* de la base de datos.

## 2.4. Ejercicio 5

Indica en la memoria cual sería la consulta que nos proporcione la posición de los bloques del *extent* en el datalle.

En la figura 5 se muestra la posición donde comienzan los bloques del extent. El primero de ellos, el bloque con *BLOCK\_ID* 368, comienza en la posición de memoria 3014656 (bytes).

---

```
1 SELECT block_id, block_id*bytes/blocks as block_position
2 FROM dba_extents
3 WHERE owner = 'USUARIO1' AND segment_name='MITABLA';
```

---

	BLOCK_ID	BLOCK_POSITION
1	368	3014656
2	376	3080192
3	384	3145728
4	392	3211264

**Figura 5:** Resultado de la consulta que calcula la posición en memoria (en *bytes*) donde empieza cada bloque del *extent*

## 2.5. Ejercicio 6

Los *extents* de un segmento pueden estar en distintos *datafiles*. ¿Cómo podemos saber en qué *datafile* está cada *extent*? Indícalo en la memoria del ejercicio.

La columna *file\_id* de la tabla *dba\_extents* nos indica a qué *datafile* pertenece cada *extent*.

## 2.6. Ejercicio 8

Comprueba que las vistas *vtablespace* y *vdatafile* se han actualizado convenientemente y se ha creado un fichero en el directorio correspondiente del sistema operativo. Adjunta a la memoria el resultado de las consultas. Ahora crea una tabla en este *tablespace*:

```
CREATE TABLE mitbl (micol VARCHAR2(100)) TABLESPACE mitbs;
```

Puedes insertar registros en esta tabla con la siguiente sentencia (o con un bloque PLSQL como el del apartado 3.3):

```
INSERT INTO mitbl (SELECT 'dábale arroz a la zorra el abad'
FROM DUAL CONNECT BY level <= 15000);
COMMIT;
```

TS#	NAME	INCLUDED_IN_DATABASE_BACKUP	BIGFILE	FLASHBACK_ON	ENCRYPT_IN_BACKUP
1	0 SYSTEM	YES	NO	YES	(null)
2	1 SYSAUX	YES	NO	YES	(null)
3	2 UNDOTBS1	YES	NO	YES	(null)
4	4 USERS	YES	NO	YES	(null)
5	3 TEMP	NO	NO	YES	(null)
6	6 MITBS	YES	NO	YES	(null)

**Figura 6:** Lista de *tablespace* de la instalación Oracle tras haber ejecutado las instrucciones SQL de este apartado. Con respecto a lo mostrado en el apartado 2 se ha añadido uno nuevo (sexta entrada en la lista).

FILE#	NAME	CREATION_CHANGE#	CREATION_TIME	TS#	RFILE#	STATUS	ENABLED
1	1 /u01/app/oracle/oradata/orcl/system01.dbf	7	13/08/09	0	1	SYSTEM	READ WRITE
2	2 /u01/app/oracle/oradata/orcl/sysaux01.dbf	2164	13/08/09	1	2	ONLINE	READ WRITE
3	3 /u01/app/oracle/oradata/orcl/undotbs01.dbf	752002	13/08/09	2	3	ONLINE	READ WRITE
4	4 /u01/app/oracle/oradata/orcl/users01.dbf	18243	13/08/09	4	4	ONLINE	READ WRITE
5	5 /u01/app/oracle/oradata/orcl/midf01.dbf	914312	16/10/19	6	5	ONLINE	READ WRITE

**Figura 7:** Lista de *datafiles* actualizada donde se ha creado un nuevo fichero (*midf01.dbf*) tras haber creado la tabla *mitbl*.

## 2.7. Ejercicio 9

Consulta los *extents* del segmento correspondiente a los datos de esta tabla (reutiliza alguna consulta anterior).

```
1 SELECT * FROM dba_extents
2 WHERE tablespace_name LIKE 'MITBS' AND segment_name like 'MITBL';
```

	OWNER	SEGMENT_NAME	PARTITION_NAME	SEGMENT_TYPE	TABLESPACE_NAME	EXTENT_ID	FILE_ID	BLOCK_ID	BYTES	BLOCKS	RELATIVE_FNO
1	SYSTEM	MITBL	(null)	TABLE	MITBS	0	5	8	65536	8	5
2	SYSTEM	MITBL	(null)	TABLE	MITBS	1	5	16	65536	8	5
3	SYSTEM	MITBL	(null)	TABLE	MITBS	2	5	24	65536	8	5
4	SYSTEM	MITBL	(null)	TABLE	MITBS	3	5	32	65536	8	5
5	SYSTEM	MITBL	(null)	TABLE	MITBS	4	5	40	65536	8	5
6	SYSTEM	MITBL	(null)	TABLE	MITBS	5	5	48	65536	8	5
7	SYSTEM	MITBL	(null)	TABLE	MITBS	6	5	56	65536	8	5
8	SYSTEM	MITBL	(null)	TABLE	MITBS	7	5	64	65536	8	5
9	SYSTEM	MITBL	(null)	TABLE	MITBS	8	5	72	65536	8	5
10	SYSTEM	MITBL	(null)	TABLE	MITBS	9	5	80	65536	8	5
11	SYSTEM	MITBL	(null)	TABLE	MITBS	10	5	88	65536	8	5

## 2.8. Ejercicio 10

Contesta a las siguientes preguntas: ¿Qué ocurre si ejecutas por segunda vez la consulta de inserción anterior? ¿Por qué? Puedes utilizar *Enterprise Manager* para ver el estado y ocupación de los *tablespaces*: en la pestaña *Server*, accede al enlace *Tablespaces* para ver cuál puede ser el motivo del mensaje de error proporcionado si vuelves a ejecutar la consulta de inserción.

Seleccionar	Nombre	Tamaño Asignado (MB)	Espacio Usado (MB)	Espacio Usado Asignado (%)	Ampliación Automática	Espacio Libre Asignado (MB)	Estado
<input checked="" type="radio"/>	MITBS	1,0	1,0	100,0	NO	0,0	✓
<input type="radio"/>	SYSAUX	520,0	491,7	94,6	YES	28,3	✓
<input type="radio"/>	SYSTEM	680,0	672,8	98,9	YES	7,2	✓
<input type="radio"/>	TEMP	29,0	1,0	3,4	YES	28,0	✓
<input type="radio"/>	UNDOTBS1	90,0	7,5	8,3	YES	82,5	✓
<input type="radio"/>	USERS	5,0	3,1	61,2	YES	1,9	✓

Tamaño Total Asignado (GB) 1,29    ✓ Online    ✗ Offline    📖 Sólo Lectura  
 Total Usado (GB) 1,15  
 Espacio Total Libre Asignado (GB) 0,14

**Figura 8:** Estado de los *tablespaces* existentes en la instalación de Oracle.

Como vemos en la figura 8, el tamaño máximo asignado al *tablespace* MITBS es de un 1MB. Ya está ocupado y no es extensible, por lo tanto no se pueden insertar más datos.



### 3. Otras estructuras de almacenamiento.

#### 3.1. Ejercicio 11

**Detalla cuales son los ficheros del sistema operativo para estas estructuras y su tamaño.**

A continuación tenemos la lista de ficheros del sistema operativo relacionados con la instalación de Oracle. Tenemos ficheros de los tipos *datafile*, *control file* y *redo log file*. Además tenemos un *temp file* y un *undo file*.

```

1 oracle@ubuntu32vb:/u01/app/oracle/oradata/orcl$ ls -l
2 total 1493308
3 -rw-r----- 1 oracle oinstall  9748480 oct 16 18:34 control01.ctl
4 -rw-r----- 1 oracle oinstall  1056768 oct 16 18:26 midf01.dbf
5 -rw-r----- 1 oracle oinstall  52429312 oct 16 18:34 redo01.log
6 -rw-r----- 1 oracle oinstall  52429312 oct 16 16:57 redo02.log
7 -rw-r----- 1 oracle oinstall  52429312 oct 16 16:57 redo03.log
8 -rw-r----- 1 oracle oinstall  545267712 oct 16 18:33 sysaux01.dbf
9 -rw-r----- 1 oracle oinstall  713039872 oct 16 18:34 system01.dbf
10 -rw-r----- 1 oracle oinstall  30416896 oct 16 18:06 temp01.dbf
11 -rw-r----- 1 oracle oinstall  94380032 oct 16 18:34 undotbs01.dbf
12 -rw-r----- 1 oracle oinstall   5251072 oct 16 17:49 users01.dbf

```

#### 3.2. Ejercicio 12

**Utiliza la consulta siguiente para localizar a través de los comandos de Oracle el *controlfile* del sistema:**

```
SELECT * FROM v$controlfile;
```

**¿Cuántos resultados se muestran? Indica en la memoria del ejercicio el nombre completo de los ficheros del sistema operativo que contienen *controlfiles*.**

	STATUS	NAME	IS_RECOVERY_DEST_FILE	BLOCK_SIZE	FILE_SIZE_BKLS
1	(null)	/u01/app/oracle/oradata/orcl/control01.ctl	NO	16384	594
2	(null)	/u01/app/oracle/flash_recovery_area/orcl/control02.ctl	NO	16384	594

**Figura 9:** Resultado de la consulta anterior donde se muestran los nombres completos de los *control files* del sistema.

### 3.3. Ejercicio 13

Utiliza la siguiente consulta para ver los grupos de *redo log files*, cuántos ficheros (miembros, en la terminología de Oracle) forman cada grupo, y qué tamaño tienen:

```
SELECT * FROM v$log;
```

GROUP#	THREAD#	SEQUENCE#	BYTES	BLOCKSIZE	MEMBERS	ARCHIVED	STATUS	FIRST_CHANGE#	FIRST_TIME	NEXT_CHANGE#	NEXT_TIME
1	1	1	10 52428800	512	1	NO	CURRENT	904949 16/10/19	281474976710655 (null)		
2	2	1	8 52428800	512	1	NO	INACTIVE	851980 08/09/16		873215 17/07/19	
3	3	1	9 52428800	512	1	NO	INACTIVE	873215 17/07/19		904949 16/10/19	

**Figura 10:** Resultado de la consulta anterior donde tenemos 3 grupos de *redo files*, cada uno de ellos con un único fichero.

### 3.4. Ejercicio 14

A continuación, utiliza la siguiente consulta para saber cuales son los ficheros del disco que se utilizan para almacenar los *redo log files*:

```
SELECT * FROM v$logfile;
```

**¿Cuántos grupos de *redo log files* hay en el sistema? ¿Cuántos ficheros tiene cada grupo?**

GROUP#	STATUS	TYPE	MEMBER	IS_RECOVERY_DEST_FILE
1	3 (null)	ONLINE	/u01/app/oracle/oradata/orcl/redo03.log	NO
2	2 (null)	ONLINE	/u01/app/oracle/oradata/orcl/redo02.log	NO
3	1 (null)	ONLINE	/u01/app/oracle/oradata/orcl/redo01.log	NO

**Figura 11:** Resultado de la consulta anterior donde se muestra el nombre completo de los ficheros en disco donde se almacenan los *redo files*. Tenemos 3 grupos cada uno de ellos con un fichero.

### 3.5. Ejercicio 15

Los *redo log files* se pueden configurar sin necesidad de parar la base de datos. Prueba a añadir un miembro a la lista de *log files* del grupo activo con el siguiente comando:

```
ALTER DATABASE ADD LOGFILE MEMBER '/u01/app/oracle/oradata/orcl/redo01b.log' TO GROUP 1;
```

GROUP#	STATUS	TYPE	MEMBER	IS_RECOVERY_DEST_FILE
1	3 (null)	ONLINE	/u01/app/oracle/oradata/orcl/redo03.log	NO
2	2 (null)	ONLINE	/u01/app/oracle/oradata/orcl/redo02.log	NO
3	1 (null)	ONLINE	/u01/app/oracle/oradata/orcl/redo01.log	NO
4	1 INVALID	ONLINE	/u01/app/oracle/oradata/orcl/redo01b.log	NO

**Figura 12:** Resultado de la consulta anterior donde se muestra un nuevo *redo file*. Ha sido añadido y se ha marcado como inválido.

## 4. Manejo de las estructuras de datos en memoria

### 4.1. Ejercicio 16

Incluye los resultados en la memoria del ejercicio:

Primero determina los tamaños de las estructuras de datos de la SGA (System Global Area) con la siguiente consulta SQL:

```
select * from v$sgainfo;
```

NAME	BYTES	RESIZEABLE
1 Fixed SGA Size	1338392	No
2 Redo Buffers	5615616	No
3 Buffer Cache Size	406847488	Yes
4 Shared Pool Size	209715200	Yes
5 Large Pool Size	4194304	Yes
6 Java Pool Size	4194304	Yes
7 Streams Pool Size	4194304	Yes
8 Shared IO Pool Size	0	Yes
9 Granule Size	4194304	No
10 Maximum SGA Size	636100608	No
11 Startup overhead in Shared Pool	62914560	No
12 Free SGA Memory Available	0 (null)	

**Figura 13:** Resultado de la consulta anterior donde se muestran los tamaños de las estructuras de datos SGA.

Para las estructuras de datos dinámicas, determina los tamaños mínimo y máximo que pueden tener las estructuras que has visto en clase, así como el tamaño que tienen actualmente:

```
select * from v$sga_dynamic_components;
```

COMPONENT	CURRENT_SIZE	MIN_SIZE	MAX_SIZE
1 shared pool	209715200	209715200	209715200
2 large pool	4194304	4194304	4194304
3 java pool	4194304	4194304	4194304
4 streams pool	4194304	4194304	4194304
5 DEFAULT buffer cache	406847488	406847488	406847488
6 KEEP buffer cache	0	0	0
7 RECYCLE buffer cache	0	0	0
8 DEFAULT 2K buffer cache	0	0	0
9 DEFAULT 4K buffer cache	0	0	0
10 DEFAULT 8K buffer cache	0	0	0
11 DEFAULT 16K buffer cache	0	0	0
12 DEFAULT 32K buffer cache	0	0	0
13 Shared IO Pool	0	0	0
14 ASM Buffer Cache	0	0	0

**Figura 14:** Resultado de la consulta anterior (se han omitido varios campos) donde se muestra el tamaño mínimo y máximo de las estructuras de datos y el tamaño actual de las mismas.

Por último, consulta la información de las PGA (Program Global Areas) y determina: el tamaño máximo que se puede dedicar (allocate) a PGA, el tamaño total dedicado actualmente, y el espacio que se está utilizando actualmente. Utiliza la consulta:

```
select * from v$pgastat;
```

NAME	VALUE	UNIT
1 aggregate PGA target parameter	210763776	bytes
2 aggregate PGA auto target	136710144	bytes
3 global memory bound	42151936	bytes
4 total PGA inuse	58881024	bytes
5 total PGA allocated	86205440	bytes
6 maximum PGA allocated	108170240	bytes
7 total freeable PGA memory	15335424	bytes
8 process count	38	(null)
9 max processes count	45	(null)
10 PGA memory freed back to OS	921370624	bytes
11 total PGA used for auto workareas	0	bytes
12 maximum PGA used for auto workareas	2544640	bytes
13 total PGA used for manual workareas	0	bytes
14 maximum PGA used for manual workareas	0	bytes
15 over allocation count	0	(null)
16 bytes processed	420602880	bytes
17 extra bytes read/written	0	bytes
18 cache hit percentage	100	percent
19 recompute count (total)	2118	(null)