

Probleme:

Laborator 5:

1. (lab 8) Sa se creeze un tablespace de tip undo numit undo_ubd11 care sa aiba un fis de data undo11 cu dim de un 1M e:\\student

```
CREATE UNDO TABLESPACE undo_ubd DATAFILE  
'E:/Student/undo11.dbf' SIZE 1M;
```

2. Sa se arate din dictionar numele fisierul si starea tablespaceului permanent aferent userului current

```
select tablespace_name, status from dba_tablespaces where  
tablespace_name = (select default_tablespace from dba_users  
where username = 'UBD1');
```

```
select username,default_tablespace from dba_users where  
username like 'UBD1';
```

```
*select tablespace_name, file_name FROM dba_data_files where  
tablespace_name like 'BD_DATA';
```

3. Sa se arate din dictionar numele tablespace-urilor permanente si temporare setate ca default

```
*select property_value from database_properties where  
property_name like '%DEFAULT%'
```

UNION

```
select default_tablespace from dba_users,dba_tablespaces where  
default_tablespace=tablespace_name and contents like  
'%PERMANENT';
```

```
select tablespace_name, contents from dba_tablespaces where  
tablespace_name in (select distinct default_tablespace from  
dba_users) and contents IN ('PERMANENT', 'TEMPORARY');
```

```
select property_name,property_value from database_properties  
where property_name in  
( 'DEFAULT_TEMP_TABLESPACE','DEFAULT_PERMANENT  
_TABLESPACE');
```

4. (lab 4)Aratati din dictionar care e cel mai mare fis temp, nr bytes
afereat userului curent data creerii si starea lui

```
select name, creation_time, status, bytes from v$tempfile where  
bytes = (select max(bytes) from v$tempfile);
```

5. Sa se creeze un tablespace temporar ubdtemp care sa aiba un fisier
de date ubd1 cu dimensiunea de 2 MB creat in folderul student pe
discul E

```
create temporary tablespace ubdtemp tempfile  
'E:/Student/ubd1.dbf' size 2M;
```

6. (lab 6)Sa se adauge un nou fisier de date ubd12 cu dimensiunea de
1 M la tablespace permanent curent si apoi sa se verifice in
dictionar daca a fost creat

```
alter tablespace BD_DATA add datafile 'E:\Student\ubd12.dbf'  
size 1M;
```

```
select tablespace_name, file_name FROM dba_data_files where  
tablespace_name like 'BD_DATA';
```

7. (lab 6)Sa se mareasca dim fis ubd1 la 4 M si apoi sa se verifice in
dictionar noua dimensiune

```
alter database tempfile 'E:/Student/ubd1' resize 5M;
```

```
select * from dba_temp_files where tablespace_name =  
'TEMP_UBD';
```

8. (lab 4) Sa se afiseze din dictionar numele fis temporare aferente userului curent aflate in starea online , data cand au fost create si dimensiunea lor in blocuri

```
select name,creation_time, block_size from v$tempfile where  
status = 'ONLINE';
```

9. (lab 6) Sa se creeze un tablespace temp temp_ubd care sa aibe un fisier de date ubd1.dbf cu dimensiunea de 2 mega si cu o extensie uniforma de 1 M

```
CREATE TEMPORARY TABLESPACE temp_ubd TEMPFILE  
'E:/Student/ubd01.dbf' SIZE 2M EXTENT MANAGEMENT  
LOCAL UNIFORM SIZE 1M;
```

LABORATOR 6

1. Sa se arate din dictionar data cand a fost creata baza de date curenta si data cand au fost resetate fisierele de log (data sa fie in format ora-minut-secunda)

```
SELECT to_char(enable_time, 'DD-MM-YYY HH:MI:SS')  
data_create FROM V$THREAD;
```

```
*SELECT to_char(resetlogs_time, 'DD-MM-YYY HH:MI:SS')  
data_reset FROM V$DATABASE;
```

2. (lab 5) Sa se arate din dictionar numele instanei curente si grupurile fisierele de log aflate in starea open

```
SELECT instance, status, current_group# FROM V$THREAD  
where status='OPEN';
```

3. (lab 3) Sa se arate din dictionar care este structura tabelara a tablei emp din userul current

```
SELECT * FROM user_tab_columns where owner='UBD1'
and table_name='EMP';
```

4. (lab 3) Sa se creeze o cheie externa pe tabla emp ca fiind o relatie pe coloana deptno din tabela dept; Apoi sa se verifice in dictionar daca a fost creata constrangerea prin vizualizarea tuturor constrangerilor aferente userului ubd1.

```
alter table emp add constraint emp_fk foreign key(deptno)
references dept(deptno);
```

```
select owner,constraint_name,constraint_type,table_name
from user_constraints where owner='UBD1';
```

5. Sa se arata din dictionar numele instantei curente, host, data startarii si starea instantei

```
select a.instance,
TO_CHAR(a.open_time,'DD:MM:YYYY'),TO_CHAR(a.open
_time,'HH:MI:SS'),a.status,b.host_name from v$thread a,
v$instance b where a.instance = b.instance_name;
```

6. Sa se arate din dictionar numele si starea instantei curente si data ultimului checkpoint in format cu ora, minut, secunda

```
select
instance,status,TO_CHAR(checkpoint_time,'DD:MM:YYYY'),
TO_CHAR( checkpoint_time,'HH:MI:SS') from v$thread;
```

7. Sa se arate care sunt grantificarile facute de userul scott si pe ce tabele si cine e beneficiarul

```
SELECT TABLE_NAME, PRIVILEGE, GRANTEE FROM  
USER_TAB_PRIVS WHERE GRANTOR = 'SCOTT';
```

8. (lab 3)Sa se arate care sunt tabelele la care proprietar e userul ubd1

```
select table_name from user_tables;
```

```
select owner, object_name from all_objects where owner =  
'UBD1' and object_type = 'TABLE';
```

9. (lab 7)Care este spatiul liber ca numar de blocuri in tablespace-ul permanent aferent userului current

```
select tablespace_name, sum(blocks) from dba_free_space  
where tablespace_name='BD_DATA' group by  
tablespace_name;
```

10. (lab 7)Sa se creeze o copie a tablei emp numita emp1_ubd apoi sa se arate din dictionar proprietarul , numele, tipul si marimea in blocuri aacestei tabele

```
create table emp1_ubd as select * from emp;
```

```
select owner, segment_name,segment_type, blocks from  
dba_segments where segment_name='EMP1_UBD';
```

11. Sa se arate care sunt tabelele partitionate aferente userului system(dba_extents)

```
select owner, partition_name from dba_segments where  
segment_type='TABLE PARTITION' and owner='SYSTEM';
```

12. Sa se arate din dictionar cea mai mare extensie de segment de rollback, numele txt in care este creat segmental si dimensiunea in bytes a extensiei

```
select segment_name, tablespace_name, bytes from  
dba_segments where segment_type='ROLLBACK' and  
extents= (select max(extents) from dba_segments where  
segment_type='ROLLBACK' group by tablespace_name);
```

13. Sa se arate din dictionar marimea in bytes si blocuri a tablespaceului temporar aferent userului curent si in ce fisier este create spatiul respectiv

```
select FILE_NAME, BYTES, BLOCKS from  
DBA_TEMP_FILES where TABLESPACE_NAME =  
'BD_TEMP';
```

14. Sa se arate din dictionar numele tablespaceului permanent afernt userului curent si care e extensia initiala a lui

```
select tablespace_name, initial_extent from  
DBA_TABLESPACES where tablespace_name = 'BD_DATA';
```

15. Sa se arate care sunt indecsii creati pe tabelele din userul scott , in ce table space sunt creati si cate blocuri au alocate

```
select tablespace_name,blocks,segment_name from  
dba_segments where owner='SCOTT'and  
segment_type='INDEX';
```

16. Sa se arate din dictionar numele celui mai mare segment de rollback, numele tablespace-ului pe care este creat si dimensiunea lui in bytes

```
select segment_name from dba_segments where segment_type  
= 'ROLLBACK';
```

```
select segment_name, tablespace_name, MAX(bytes)
from dba_extents where segment_name in (select
segment_name from dba_segments where segment_type =
'ROLLBACK') group by segment_name, tablespace_name;
```

LABORATOR 7

1. (lab 8)Faceti o lista cu numele, marimea in bytes si starea segmentelor de undo din sesiunea curenta

```
select a. name, b.rssize, b.status from v$rollname a INNER
JOIN v$rollstat b ON a.USN = b.USN;
```

2. (lab 8)Faceti o inserare in tabela emp, apoi faceti o lista cu adresa tranzactiei, segmentul de rollback pe care il foloseste, numarul de blocuri generate si numele fisierului de rollback utilizat.

```
insert into emp values (1000,
'TEST','TRANZACT',1111,sysdate, 100,0,10);
```

```
select a.addr, a.xidusn, a.used_ublk, b.file_name from
v$transaction a left join dba_data_files b on a.start_ubafil =
b.file_id;
```

3. (lab 8)Faceti o lista cu userul current, adresa sesiunii, adresa tranzactiei si numarul segmentului de rollback folosit in sesiunea curenta

```
select a.username, a.saddr, b.addr, b.start_ubablk from
v$session a inner join v$transaction b on a. saddr = b.ses_addr;
```

```
select a.username, a.saddr, a.taddr , b.name from v$session a,
v$rollname b where b.name = (select x.name from v$rollname
```

x, v\$transaction y where x.usn=y.xidusn) and
a.username='UBD1';

4. (lab 8)Faceti o lista cu numele, tipul si starea tablespace+ului
alocat pt segmentele temporare de sortare din sesiunea curenta
precum si nr maxim de blocuri de sortare alocate fiecarui
segment temporar

```
select a.tablespace_name, a.max_sort_size,a.max_sort_blocks,  
b.contents from v$sort_segment a inner join dba_tablespaces b  
on a.tablespace_name = b.tablespace_name;
```

5. Sa se faca o lista cu numele , marimea in bytes si starea
segmentelor undo din sesiunea curenta

```
select a.usn,b.name,rssize,status from v$rollstat a, v$rollname  
b where a.usn = b.usn;
```

6. (lab 8)Sa se faca o lista cu userul curent , idul sesiunii
curente , starea ei si tablespaceul permanent ascoiat

```
select a.username,a.sid,b.tablespace_name , a.status from  
v$session a , dba_rollback_segs b where a.username ='UBD1'  
and b.owner='UBD1';
```

7. Sa se faca o lista cu idul sesiunii curente userul curent
numele si tipul tablespaceului pt segmentele temporare de
sortare , nr de extensii si numarul total de blocuri alocate

```
select a.sid,a.username , t.tablespace_name , t.total_extents ,  
t.total_blocks,d.status from v$session a, v$sort_segment  
t,dba_temp_files v where v.tablespace_name =  
t.tablespace_name and t.CURRENT_USERS = a.USER#;
```


LABORATOR 8

1. Faceti o lista cu numele tabelelor, tablespace-ul permanent aferent si proprietarul pt toate tabelele din userul current

```
SELECT table_name, tablespace_name, owner FROM  
dba_tables WHERE owner = 'UBD1' and  
tablespace_name='BD_DATA';
```

2. Lista cu numele, tipul si data crearii tuturor triggerilor din baza de date creati in anul 2002.

```
select object_name, object_type, created from dba_objects  
where object_type = 'TRIGGER' and to_char(created,  
'YYYY') = 2002;
```

3. (lab 10)Verificati in dictionar numarul de blocuri alocate unui index si care este procentul utilizat din spatiul alocat

```
ANALYZE INDEX scott.emp_name_idx VALIDATE  
STRUCTURE;
```

```
SELECT name, blocks, used_space, pct_used FROM  
index_stats WHERE name = 'EMP_NAME_IDX';
```

4. Afisati din dictionar numele instantei curente, numele indecsilor si tabelele aferente

```
SELECT index_name, table_name, instance_name from  
dba_indexes a INNER JOIN v$instance b ON a.instances =  
b.instance_number;
```

LABORATOR 10

1. Sa se creeze un index de tip arb emp1_idx pe col deptno+empno a tabeli emp1, care e o copie a tabeli emp. Verif apoi in dict component indexului.

```
CREATE TABLE emp1 AS SELECT * FROM emp;
```

```
SELECT * FROM emp1;
```

```
CREATE INDEX scott.emp1_idx ON scott.emp1(empno, deptno)PCTFREE 30 STORAGE(INITIAL 200K NEXT 200K PCTINCREASE 0 MAXEXTENTS 50) TABLESPACE USERS;
```

```
SELECT index_name, table_owner, table_name, column_name  
FROM dba_ind_columns WHERE index_name='EMP1_IDX';
```

2. Sa se faca apoi o extensie de 100k pt indexul creat intr-un fis de date ubd1 care are marimea 1Mb.

```
alter tablespace users add datafile  
'D:/app/Tablespaces/ubd3.dbf' size 1M;
```

```
alter index emp1_idx allocate extent(size 100k datafile  
'D:/app/Tablespaces/ubd3.dbf');
```

3. Verif in dict nr de blocuri alocate indexului si procentul utilizat din spatial alocat.

```
ANALYZE INDEX emp1_idx VALIDATE STRUCTURE;  
SELECT name, blocks, pct_used FROM index_stats;
```

4. Dict: numele instantei crt, numele indecsilor si tabeli aferente.

```
select a.instance_name, b.index_name, b.table_name from  
dba_indexes b, v$instance a where b.owner='SCOTT';
```

1. sa se creeze un view deptview pe toate coloanele din tabela dept din userul current apoi sa se arate din dictionar proprietarul, numele, tipul si data cand a fost creat.

```
create view dept_view as select * from dept;
```

```
select * from dba_objects where object_name like  
'dept_view';
```

```
select owner, object_type, object_name, created from  
dba_objects where object_name like 'DEPT_VIEW';
```

2. sa se faca o copie a tablei emp, emp_new din userul curent si sa se elimine coloana comm, si sa se verifice in dictionar noua structura

```
create table emp_new as select * from emp;
```

```
alter table emp_new drop column comm cascade constraints  
checkpoint 1000;
```

```
select column_name, data_type from dba_tab_column where  
table_name = 'EMP_NEW';
```

```
desc emp_new;
```

3. in copia emp_new, dezactivati coloana job apoi activati-o la loc.

```
alter table emp_new set unused column job cascade  
constraints;
```

```
alter table emp_new add job varchar2(9);
```

4. sa se aloce tabelii emp_new o extensie de 100k, in table space ul permanent aferent userului curent.

```
select property_value from database_properties where  
property_name like  
'DEFAULT%PERMANENT%TABLESPACE'; -> USERS
```

```
alter table emp_new allocate extent (size 100k);
```