Obrada transakcija, izolacija i zaključavanje u Oracle-u

Mentor: doc. dr Aleksandar Stanimirović
Student: Vladana Stojiljković
br. ind 1135

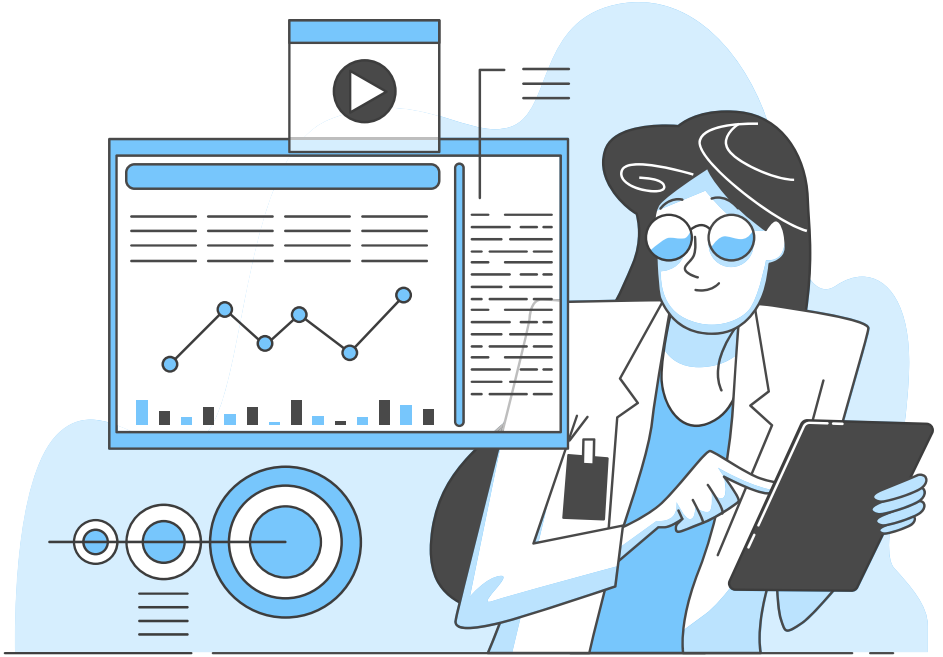
Sadržaj

01 Transakcije

02 Izolacija

03 Zaključavanje

04 Zaključak



01

Transakcije

Transakcije

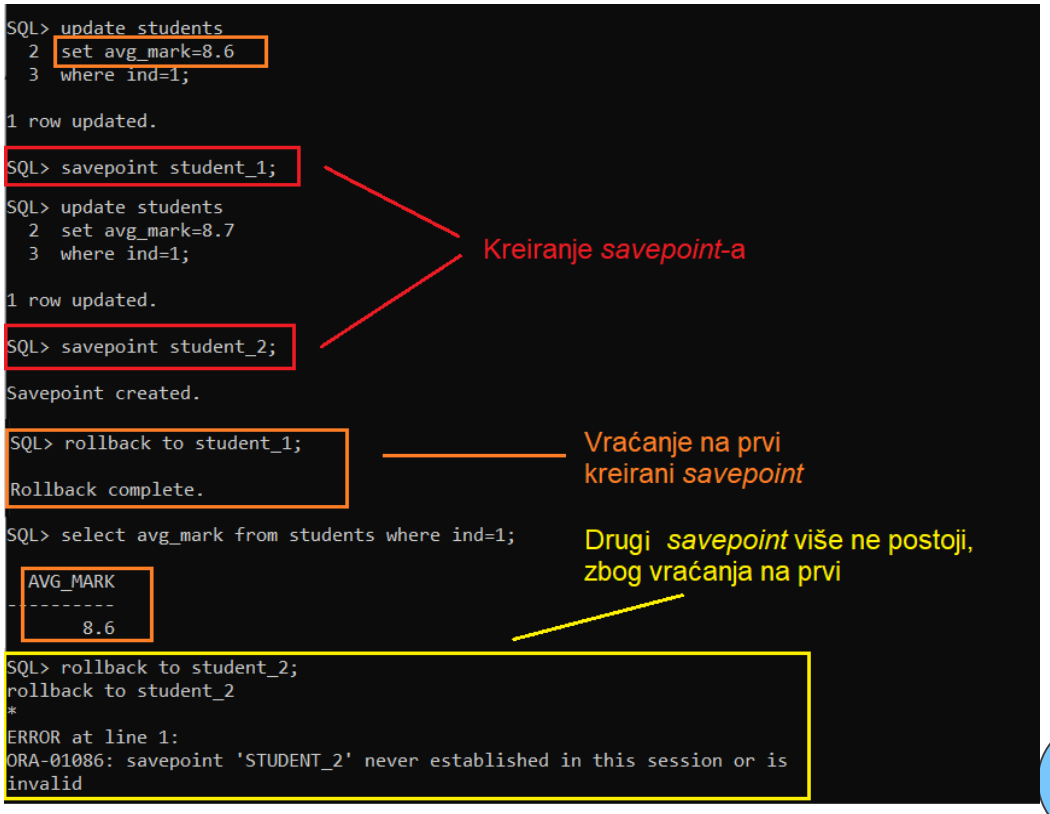
- Konzistentnost
- ACID svojstva
- *Commit*
- *Rollback*
- *UNDO* segmenti

```
SQL> set transaction name 'students_update';

Transaction set.

SQL> update students set avg_mark = 9.4
  2  where ind = 1;
SQL> select xid, name, status
  2  from v$transaction;
XID
----------------
NAME
--------------------------------------------------------------------------------
STATUS
----------------
07000F0055050000
students_update
ACTIVE
SQL> rollback;

Rollback complete.

SQL> select xid, name, status
  2  from v$transaction;

no rows selected

SQL> update students set avg_mark = 9.5 where ind = 1;

1 row updated.

SQL> commit;

Commit complete.

SQL> select xid, name, status
  2  from v$transaction;

no rows selected
```

commit i *rollback* okončavaju transakciju

Transakcije

- TCL naredbe
- Distribuirane transakcije
- Autonomne transakcije



Autonomne transakcije

Nezavisnost

```
SQL> create or replace procedure non_auto_insert
  2  as
  3  begin
  4  insert into moja (id) values (5);
  5  commit;
  6  end;
  7  /

Procedure created.
SQL> begin
  2  insert into moja (id) values (10);
  3  non_auto_insert;
  4  rollback;
  5  end;
  6  /

PL/SQL procedure successfully completed.

SQL> select * from moja;

        ID
----------
        10
         5
```

```
SQL> create or replace procedure auto_insert
  2  as
  3  pragma autonomous_transaction;
  4  begin
  5  insert into moja (id) values (3);
  6  commit;
  7  end;
  8  /

Procedure created.

SQL> delete from moja;

2 rows deleted.

SQL> commit;

Commit complete.

SQL> begin
  2  insert into moja (id) values (13);
  3  auto_insert;
  4  rollback;
  5  end;
  6  /

PL/SQL procedure successfully completed.

SQL> select * from moja;

        ID
----------
         3
```

02

Izolacija

Izolacija

Prljava čitanja

Neponavljajuća čitanja

Fantomska čitanja

Nivoi izolacije

01 *READ COMMITTED nivo*

- Fantomska čitanja
- Izgubljeni upisi

```
SQL> select * from professors;

NAME                          SALARY
-------------------- ----------
adela                          12500
huan                           10500

SQL> update professors
  2
SQL> update professors set salary=13500
  2  where name='adela';
```

T1

```
SQL> set transaction isolation level read committed;

Transaction set.

SQL> select * from professors;

NAME                          SALARY
-------------------- ----------
adela                          12500
huan                           10500

SQL> update professors set salary=9500
  2  where name='huan';

1 row updated.
```

T2

```
SQL> insert into professors (name, salary)
  2  values ('carmen', 25000);

1 row created.
```

T1

čeka da T1 izvrši *commit*

```
SQL> select * from professors;

NAME                          SALARY
-------------------- ----------
adela                          12500
huan                           9500

SQL> update professors set salary=30000
  2  where name='adela';
-
```

T2

```
SQL> commit;

Commit complete.
```

T1

```
1 row updated.

SQL> select * from professors;

NAME                          SALARY
-------------------- ----------
adela                          30000
huan                           9500
carmen                         25000

SQL> commit;

Commit complete.
```

T2

```
SQL> select * from professors;

NAME                          SALARY
-------------------- ----------
adela                          30000
huan                           9500
carmen                         25000
```

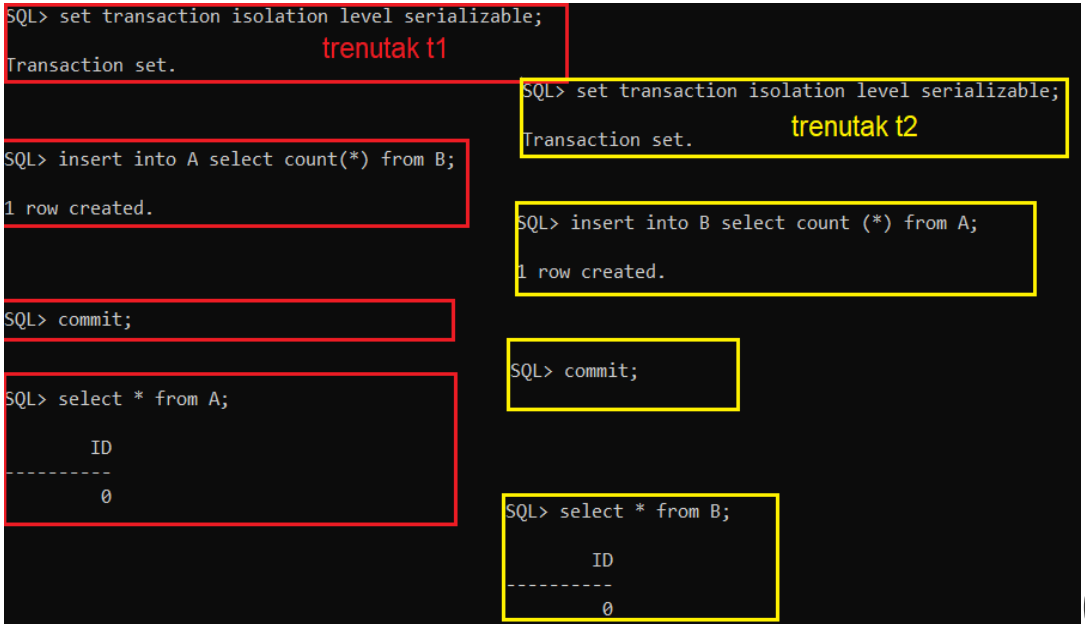
T1

Nivoi izolacije

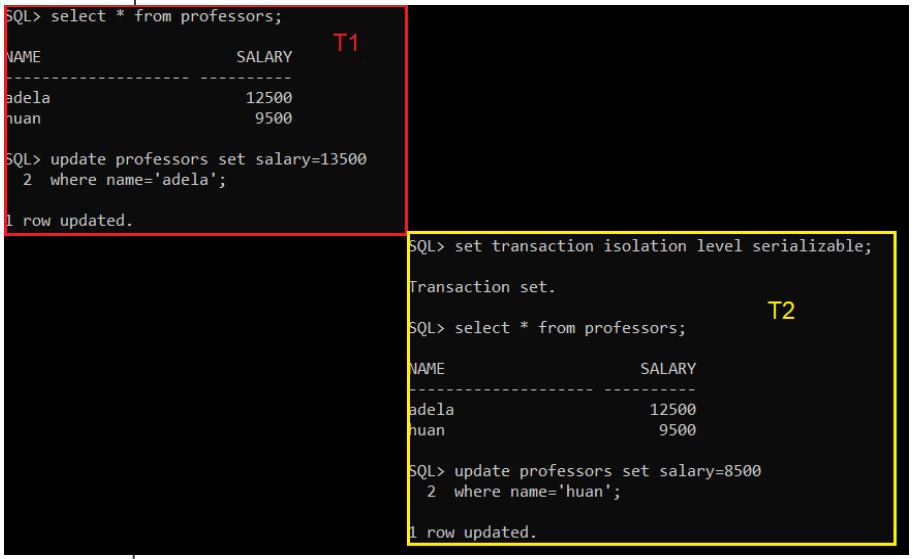
02

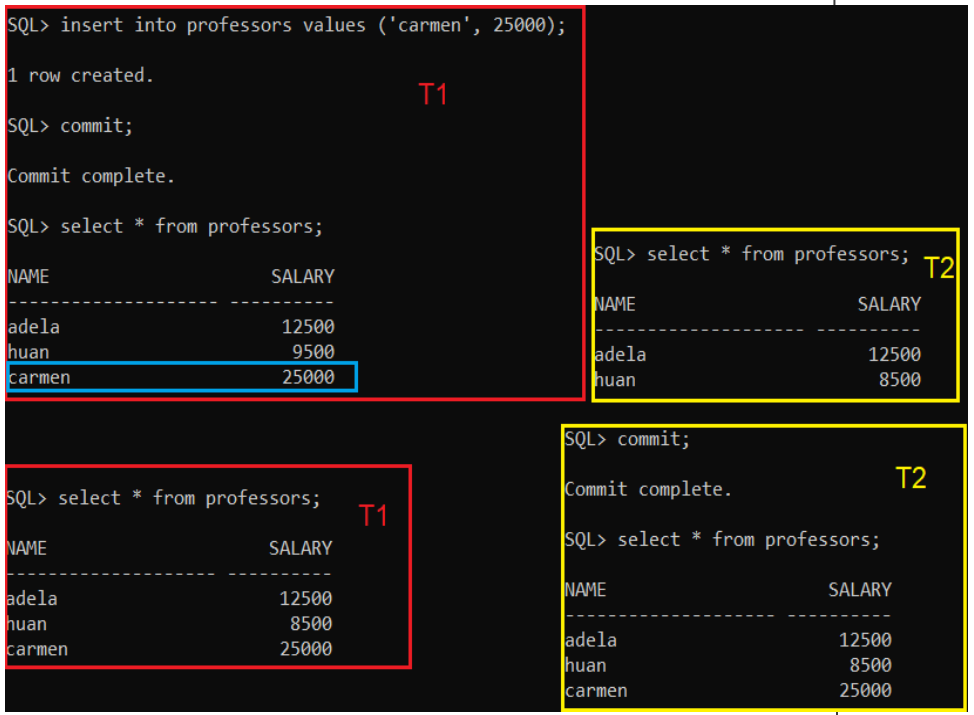
SERIALIZABLE nivo

Serijsko izvršenje



SERIALIZABLE nivo izolacije



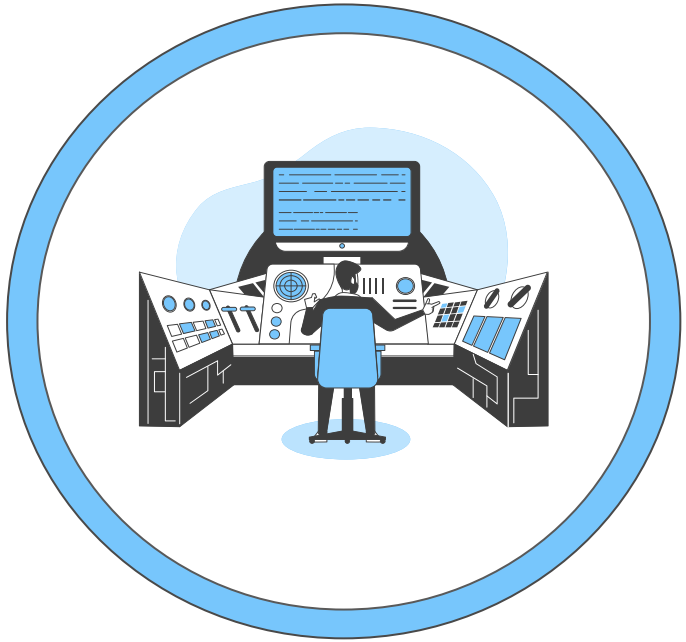


Nivoi izolacije

03

READ ONLY nivo

- Konzistentna čitanja
- SYS vrši ažuriranje
- Generisanje izveštaja



03

Zaključavanje

Zaključavanje

01 Brave

02 Zaključavanje vrste

03 Zaključavanje tabele

```
SQL> update test_locks set id=11
  2  where id=1;

1 row updated.
```
T1

```
SQL> commit;

Commit complete.
```
T1

```
SQL> update test_locks set id=111
  2  where id=11;

1 row updated.

SQL> select * from test_locks;

        ID
----------
       111
         2
         3
```
T1

```
SQL> update test_locks set id=21
  2  where id=1;
```
T2

```
0 rows updated.
```
T2

```
SQL> select * from test_locks;

        ID
----------
        11
         2
         3
```
T2

```
SQL> select * from test_locks where id=2
  2  for update;

        ID
----------
         2

SQL>
```
T1

može jer ta vrsta nije zaključana

ne može jer T1 ima bravu nad tabelom

T2

```
SQL> update test_locks set id=33
  2  where id=3;

1 row updated.

SQL> alter table test_locks add name varchar(10);
```

Automatsko zaključavanje

01 *DML* brave

- TX
- TM
- Onemogućavanje brava

```
SQL> update test set id=23 where id=22;

1 row updated.

SQL>
```

Sesija 1

Sesija 2

```
SQL> select username, v$lock.sid from v$lock, v$session where v$lock.type='TX' and v$lock.sid=v$session.sid and v$session.username = USER;

USERNAME
------------------------------------------------------------------------------
       SID
----------
SYS
       743
```

```
SQL> COMMIT;

Commit complete.
```

Sesija 1

Sesija 2

```
SQL> select username, v$lock.sid from v$lock, v$session where v$lock.type='TX' and v$lock.sid=v$session.sid and v$session.username = USER;

no rows selected

SQL>
```

```
SQL> create table tm (col1 int);

Table created.

SQL> insert into tm (col1) values (1);

1 row created.

SQL> select (select username from
  2  v$session where sid = v$lock.sid) username,
  3  sid, v$lock.type from v$lock
  4  where sid = sys_context('userenv', 'sid');

USERNAME
------------------------------------------------------------------------------
       SID TY
---------- --
SYS
       743 AE

SYS
       743 TM

SYS
       743 TX
```

Automatsko zaključavanje

02 *DDL* brave

- Ekskluzivne
- Deljive
- Raščlanjive

```
SQL> create or replace procedure p
  2  as
  3  begin
  4  null;
  5  end;
  6  /

Procedure created.

SQL> exec p;

PL/SQL procedure successfully completed.

SQL> select session_id sid, owner, name, type,
  2  mode_held held, mode_requested req
  3  from dba_ddl_locks
  4  where session_id=(select sid from v$mystat where rownum=1);

       SID
----------
OWNER
--------------------------------------------------------------------------------
NAME
--------------------------------------------------------------------------------
TYPE                                HELD       REQ
----------------------------------- ---------- ---------
       870
SYS
P
Table/Procedure/Type                Null       None


       SID
----------
OWNER
--------------------------------------------------------------------------------
NAME
--------------------------------------------------------------------------------
TYPE                                HELD       REQ
----------------------------------- ---------- ---------
       870
SYS
DBMS_SYSTEM
Table/Procedure/Type                Null       None


       SID
----------
OWNER
--------------------------------------------------------------------------------
NAME
--------------------------------------------------------------------------------
TYPE                                HELD       REQ
----------------------------------- ---------- ---------
       870
SYS
IDGEN1$
Table/Procedure/Type                Null       None
```

```
SQL> alter procedure p compile;

Procedure altered.

PL/SQL procedure successfully completed.

SQL> select session_id sid, owner, name, type,
  2  mode_held held, mode_requested req
  3  from dba_ddl_locks
  4  where session_id=(select sid from v$mystat where rownum=1);

       SID
----------
OWNER
--------------------------------------------------------------------------------
NAME
--------------------------------------------------------------------------------
TYPE                                HELD       REQ
----------------------------------- ---------- ---------
       870
SYS
DBMS_SYSTEM
Table/Procedure/Type                Null       None


       SID
----------
OWNER
--------------------------------------------------------------------------------
NAME
--------------------------------------------------------------------------------
TYPE                                HELD       REQ
----------------------------------- ---------- ---------
       870
SYS
IDGEN1$
Table/Procedure/Type                Null       None
```

Automatsko zaključavanje

03 *Sistemske brave*

Lečevi

- Štiti više objekata
- „obrtanje“ leča

Muteksi

- Jednostavni
- Štite jedan objekat
- Manje memorije
- Veća dostupnost

Interne brave

- Složene
- Brave keša rečnika podataka
- Brave nad fajlovima i logovima
- Brave nad prostorom tabela i undo segmentima

Manuelne i korisnički definisane brave

- Predefinisanje podrazumevanog mehanizma zaključavanja
- *SELECT ... FOR UPDATE* i *LOCK TABLE*
- *DBMS_LOCK*

```
SQL> select * from professors where name='adela'
  2  for update;

NAME                 SALARY
-------------------- ----------
adela                 12500
```

T1

```
SQL> update professors set salary=12800
  2  where name='adela';
```

T2

čeka!

```
SQL> lock table test_table_lock in row share mode;

Table(s) Locked.
```

T1

Druga sesija može da vrši ažuriranje!

```
SQL> update test_table_lock set id=11 where id=1;

1 row updated.
```

T2

```
SQL> lock table test_table_lock in exclusive mode;

Table(s) Locked.
```

T1

Ažuriranje nije moguće zbog ekskluzivnog režima!

```
SQL> update test_table_lock set id=21 where id=11;
```

T2

04

Zaključak

Zaključak

- Transakcije prevode bazu iz jednog konzistentnog stanja u drugo. Oracle transakcije imaju ACID svojstva.
- Nivo izolacije transakcije određuje kako su efekti jedne transakcije vidljivi drugim korisnicima. Postoji 3 nivoa izolacije.
- Mehanizam zaključavanja održava konzistentnost podataka prilikom višekorisničkog pristupa. Oracle vrši automatsko zaključavanje, ali to može da uradi i korisnik.

Hvala na pažnji!