

SUBP

Vježbe - Osnove administracije Oracle21c

Sadržaj

- Enterprise Manager Database Express, SQL*Plus i SQL Developer
- Pokretanje i gašenje instance baze podataka
- Tablespaces
- Korisnici
- Objekti šeme
- Managing undo
- Database security
- Listener
- Proactive Maintenance
- Performance Management
- Backup and Recovery

Enterprise Manager Database Express, SQL*Plus i SQL Developer

Enterprise Manager Database Express

- Predstavlja WEB orijentisani alat koji omogućava administratorima da obavljaju neke osnovne administratorske zadatke i da prije svega nadgledaju i prate rad baze podataka
- Ovaj alat je naznačen kao zastarjeo i planirano je potpuno izbacivanje ovog alata od verzije 23c
- Radi na standardnom portu 5500 sem ukoliko tokom instalacije (ili naknadnim modifikacijama) port nije postavljen na drugu vrijednost

Enterprise Manager Database Express

- Pristupa mu se unošenjem sledećeg URL- a u Internet pretraživač
 - <https://localhost:5500/em/login>
- Kredencijali
 - username: sys - ako se prijavljujemo kao administrator sa najvišim privilegijama
 - password: oracle - vrijednost zadata prilikom instalacije
 - container name: - opciono navođenje naziva kontejnera na koji se prijavljujemo (podrazumjevano CDB\$ROOT)
- Praćenje tablespaces- ova izborom opcije
 - Storage -> Tablespace

SQL*Plus

- Otvoriti CMD
- Logovanje putem sql*plus- a
 - sqlplus
`<username>/<password>[@//<ip_adresa>[:<port>]/<naziv_kontejnera>] [as sysdba] @skripta.sql`
- Logovanje putem sql*plus- a kao sys administrator
 - `sqlplus sys/oracle as sysdba`
- Prikazivanje naziva kontejnera u koji smo logovani
 - `show con_name;`

SQL*Plus

- Prebacivanje u drugi kontejner nakon logovanja
 - `alter session set container=<naziv_kontejnera>;`
- Logovanje putem sql*plus- a kao sys administrator direktno u ORCLPDB1
 - `sqlplus sys/oracle@//localhost:1521/orclpdb1 as sysdba`

SQL*Plus - Primer

- Na desktop- u napraviti folder vezbe
- U folderu napraviti fajl hr_dep.sql
- U fajl upisati
 - `select * from departments;`
- Pozicionirati se u folder vezbe u CMD- u
- Logovati se preko sqlplus- a i prilikom logovanja izvršiti skriptu hr_dep.sql
 - `sqlplus hr/hr@//localhost:1521/orclpdb1 @hr_dep.sql`

SQL Developer

- Napraviti sys konekcije na CDB (orcl) i PDB (orclpdb1) kao i hr konekciju na PDB (orclpdb1)

	SYS administrator na CDB	SYS administrator na PDB	HR korisnik na PDB
Connection name	sys_cdb	sys_pdb	hr
Username	sys	sys	hr
Password	oracle	oracle	hr
Role	SYSDBA	SYSDBA	default
Save password	Yes	Yes	Yes
Hostname	localhost	localhost	localhost
Port	1521	1521	1521
SID	orcl		
Service name		orclpdb1	orclpdb1

Pokretanje i gašenje instance baze podataka

Pokretanje i gašenje instance baze podataka

- Logovati se preko SQL*Plus- a kao sys korisnik
- Pregledanje status instance
 - `select status from v$instance;`
- Gašenje instance
 - `shutdown [immediate|normal|transactional];`
- Pokretanje instance u nomount režimu
 - `startup [nomount|mount|open];`
- Prebacivanje instance u viši režim
 - `alter database <režim>;`

Pokretanje i gašenje instance baze podataka

- Probati startovanje u nomount režimu, a zatim opet odraditi startup u mount režim. Efekti?
- Nakon startovanja u nomount režimu, probati prebacivanje u open režim. Efekti?
- Kroz SQL Developer ne možemo da radimo pokretanje i gašenje instance dok god se konektujemo na podrazumjevani način, potrebno je modifikovati način konekcije SQL Developer- a na instancu baze podataka

Pokretanje i gašenje instance baze podataka

- Oracle održava logove, skladišti informacije o događajima nad bazom podataka u nestrukturirani (plain tekstualni) fajl kao i u strukturirani (xml) fajl
- U nazivu log fajlova se često nalazi i SID instance
- Putem SQL Developer- a možemo postaviti upit da bismo vidjeli lokaciju ovih fajlova
 - `select * from v$diag_info;`
- Postoji više tipova log fajlova, od interesa su nam Diag Trace i Diag Alert log fajlovi
 - pregledati njihov sadržaj na lokacijama koje su dobijene pokretanjem upita
 - da li postoje informacije o pokretanju i gašenju instance koje smo sprovodili?

Tablespaces

Uvod

- Administratorima baze podataka pored uvida u korisničke podatke omogućen je i uvid u meta podatke, podatke koji su sadržani u riječniku podataka (data dictionary)
- Postoje tri tipa pogleda
 - ALL_* – pogledi koji sadrže informacije o objektima kojima ulogovani korisnik ima pravo pristupa uključujući i objekte čiji je vlasnik. Ovi pogledi obezbjeđuju detalje o objektima koji su vidljivi korisniku, uzimajući u obzir dodijeljene privilegije.
 - DBA_* – pogledi sadrže informacije o svim objektima u cijeloj bazi podataka. Ovi pogledi su prvenstveno namjenjeni administratorima (DBAs) i zahtevaju visok stepen privilegije. Pružaju obimne detalje o svim objektima u zavisnosti od vlasništva nad njima.
 - USER_* – pogledi sadrže informacije o objektima čiji je vlasnik trenutno ulogovani korisnik.

Uvod

- Pored standardnih pogleda postoje i dinamički pogledi koji pružaju informacije o radu instance
 - to su pogledi koji imaju prefiks v\$
 - npr. v\$instance
- **Voditi računa nad kojom konekcijom (kojim kontejnerom) se izvršavaju upiti i naredbe!**
 - Određeni objekti iz CDB kontejnera su vidljivi na nivou PDB, dok to ne mora da važi u suprotnom smjeru
 - Tokom ovog kursa uglavnom ćemo raditi administraciju nad PDB kontejnerom

Tablespaces

- Prikaz postojećih tablespaces- ova
 - `select * from dba_tablespaces;`
- Prikaz postojećih datafiles- ova
 - `select * from dba_data_files;`
- Upotrebnom dinamičkih pogleda možemo dobiti slične podatke
 - `select * from v$tablespace;`
 - `select * from v$datafile;`
 - razlike?
- Prikaz tablespace- a, pripadajućih datafile- ova i veličine datafile- a u megabajtima
 - `select dt.tablespace_name, df.file_name, df.bytes/1024/1024 AS
file_size_mb
from dba_tablespaces dt inner join dba_data_files df on
dt.tablespace_name=df.tablespace_name;`

Tablespaces

- Kreiranje tablespace- a
 - create tablespace inventory
datafile
'C:\oracle21c_setup\app\oracle\oradata\ORCL\DATAFILE\inventory01.dbf'
size 50M autoextend on next 10M maxsize unlimited
logging
extent management local uniform size 1M;
- Dodavanje fajla u postojeći tablespace
 - alter tablespace inventory
add datafile
'C:\oracle21c_setup\app\oracle\oradata\ORCL\DATAFILE\inventory02.dbf'
size 100M autoextend on next 50M maxsize 1G;

Tablespaces

- Konektovati se na PDB i prikazati sve tabele koje pripadaju USERS tablespace- u
 - `select *`
`from dba_tables`
`where tablespace_name = 'USERS';`
- Prikaz segmenata koji pripadaju tablespace- u (mogu se vidjeti i objekti koji pripadaju datom segmentu) – probati ovo kao sys u CDB i PDB i kao hr korisnik
 - `select *`
`from user_segments`
`where tablespace_name = 'USERS';`
 - `select *`
`from dba_segments`
`where tablespace_name = 'USERS';`

Tablespaces

- Prikaz extent- a koji pripadaju tablespace- u (mogu se vidjeti i objekti koji pripadaju datom extent- u) – probati ovo kao sys u CDB i PDB i kao hr korisnik
 - `select *`
`from dba_extents`
`where tablespace_name = 'USERS';`
 - `select *`
`from user_extents`
`where tablespace_name = 'USERS';`
- Brisanje tablespace- a
 - `drop tablespace inventory [including contents and datafiles] [cascade constraints];`

Korisnici

Uvod

- Ne preporučuje se kreiranje korisnika koji treba da budu administratori na nivou CDB jer mi već imamo kreirane predefinisane korisnike na nivou CDB koji mogu da administriraju bilo koju PDB
- Pregledanje podataka o korisnicima
 - `select * from all_users;`
 - `select * from dba_users;`
 - `select * from user_users;`

Korisnici

- Kreiranje običnog korisnika kao sys admin u pdb- u
 - `create user korisnik identified by ftn;`
- Probati login kroz sql developer, a zatim kroz sql*plus. Rezultat?
- Dodati privilegiju create session
 - `grant create session to korisnik;`
- Promjeniti da default tablespace za korisnika bude inventory umjesto users
 - `alter user korisnik default tablespace inventory;`
- Probati kreiranje tabele sa novim korisnikom nakon logovanja. Rezultat?
 - `create table radnik (mbr integer, constraint radnik_pk primary key (mbr));`

Korisnici

- Dodati ulogu (role) resource
 - `grant resource to korisnik;`
 - da bi se vidjeli efekti moraće da se uradi reconnect
- Pregled privilegija
 - `select grantee, privilege
from dba_sys_privs
order by grantee;`

Korisnici

- Pregled ko ima ulogu resource
 - `select *`
`from dba_role_privs`
`where granted_role = 'RESOURCE';`
- Pregled privilegija koje uključuje uloga resource
 - `select *`
`from dba_sys_privs`
`where grantee = 'RESOURCE';`

Uloge

- Pregled svih uloga
 - `select *`
`from dba_roles;`
- Pregled svih uloga koje su važeće u trenutnoj sesiji
 - `select *`
`from session_roles;`
- Pregled hijerarhije uloga
 - `select *`
`from role_role_privs;`
- Kreiranje uloge bez lozinke
 - `create role novarola not identified;`
- Kreiranje uloge sa lozinkom
 - `create role novarolapass identified by ftn;`

Privilegije

- select - dozvoljava korisniku da postavlja upite
- insert - dozvoljava korisniku da radi upis podataka
- update - dozvoljava korisniku da modifikuje podatke
- delete - dozvoljava korisniku da briše podatke
- execute - dozvoljava korisniku da izvrši uskladištene procedure i funkcije
- Postaviti upit za dobavljanje podataka iz tabele hr.departments kao novokreirani korisnik. Efekti?
 - `select * from hr.departments;`

Privilegije

- Privilegije mogu da dodjeljuju administratori baze podataka (sys admin)
- Sintaksa za dodjeljivanje privilegije
 - `grant <naziv_privilegije> on object to korisnik_ili_uloga;`
- Kreirati object level privilegiju za dobavljanje podataka iz tabele hr.departments i dodijeliti je ulozi novarola
 - `grant select on hr.departments to novarola;`
- Dodijeliti ulogu novarola novokreiranom korisniku
 - `grant novarola to korisnik;`
- Ponovo postaviti upit za dobavljanje podataka iz hr.deparments kao novokreirani korisnik. Efekti?
- `*_TAB_PRIVS` – pregled object level privilegija dodjeljenih ulozi
- `*_SYS_PRIVS` – pregled system level privilegija

Profili

- Kreiranje profila pod nazivom noviprofil sa sljedećim parametrima
 - CPU/Session: 6000, znači da u sesiji CPU može da se koristi 60 sekundi, nakon toga će korisnik biti izlogovan
 - CPU/Call: 1000, znači da ako se jedna komanda izvršava duže od 10 sekundi, biće prekinuta
 - Connect Time: 30, znači da 30 minuta traje konekcija prije nego što korisnik biva izlogovan
 - Idle Time: 15, znači 15 minuta ako korisnik ništa ne radi, biće izlogovan
 - Reads/Session: 5000, može 5000 blokova da pročita po sesiji
 - Reads/Call: 1000, može 1000 blokova da pročita po komandi
 - Private SGA: DEFAULT, prostor za private SGA
 - Composite Limit: DEFAULT, kombinacija svih limita
 - PASSWORD_LIFE_TIME: 120, šifra ističe posle 120 dana
 - PASSWORD_GRACE_TIME: nakon/tokom 30 dana kreće da upozorava o isteku/promjeni
 - FAILED_LOGIN_ATTEMPTS: 3, ako korisnik ne unese 3 puta dobro šifru, zaključava se nalog
 - PASSWORD_VERIFY_FUNCTION - može biti bilo koja funkcija

Profili

- Kreiranje profila pod nazivom noviprofil
 - ```
CREATE PROFILE noviprofil
LIMIT CPU_PER_SESSION 6000
CPU_PER_CALL 1000
CONNECT_TIME 30
IDLE_TIME 15
LOGICAL_READS_PER_SESSION 5000
LOGICAL_READS_PER_CALL 1000
COMPOSITE_LIMIT DEFAULT
PRIVATE_SGA DEFAULT
FAILED_LOGIN_ATTEMPTS 3
PASSWORD_LIFE_TIME 120
PASSWORD_GRACE_TIME 30;
```

# Profili

- Modifikacija parametra password\_grace\_time profila noviprofil
  - `alter profile noviprofil limit password_grace_time 7;`
- Dodavanje profila korisniku
  - `alter user korisnik profile noviprofil;`
- Izmjeniti kvotu za novokreiranog korisnika nad tablespace-om inventory da bude 200Mb
  - `alter user korisnik quota 200M on inventory;`

Objekti šeme

---

## Objekti šeme

- Šema je kolekcija objekata baze podataka koji pripadaju određenom korisniku
- Prilikom kreiranja korisnika kreira se i šema sa tim nazivom

# Tabele

- Naredbe treba izvršavati sa novokreiranim korisnikom
- Kreirati tabelu radnik
  - Tablespace: inventory
  - Kolone: mbr integer, ime varchar2(30), prz varchar2(30), plt integer
  - Ograničenje not null za mbr, ime i prz
  - Primarni ključ: mbr
  - Plata mora biti veća od 30000
  - ```
create table radnik (mbr integer,  
    ime varchar2(30) not null,  
    prz varchar2(30) not null,  
    plt integer,  
    constraint radnik_pk primary key(mbr),  
    constraint plt_ch check (plt>30000))  
tablespace inventory;
```

Tabele

- Dodati novu kolonu pol u tabelu radnik
 - `alter table radnik add pol char(1);`
- Pregledati sadržaj tabele
- Logovati se kroz SQL plus i dodati 2, 3 reda
 - `insert into radnik values (10, 'A1', 'B1', 35000, 'Z');`
 - `insert into radnik values (20, 'A2', 'B2', 45000, 'M');`
 - `insert into radnik values (30, 'A3', 'B3', 55000, 'Z');`

Tabele

- Pregledati dodate podatke kroz SQL Developer. Rezultat?
- Potrebno je odraditi commit nakon insert naredbi
- Da bi se vidjeli detaljni podaci o tabelama u svojstvu sys korisnika postaviti upit
 - `select *`
`from dba_tables`
`where table_name='RADNIK';`
- Da bi se vidjeli detaljni podaci o kolonama u svojstvu sys korisnika postaviti upit
 - `select *`
`from dba_tab_columns`
`where table_name='RADNIK';`

Tabele

- Brisanje tabele – kreirati tabelu radnik1 sa istim kolonama kao što ih ima tabela radnik
 - DROP, briše podatke i definiciju tabele. Ako se ne stavi PURGE opcija na kraju, sve ostaje u Recycle Bin i može da se vrati.
 - DELETE, briše samo podatke, može da se uradi rollback. Prostor ostane zauzet.
 - TRUNCATE, briše samo podatke, ali ne može da se uradi rollback. Mnogo brži od DELETE. Potencijalno se oslobađa prostor.
- Odraditi drop tabele radnik1
- Pregledati kroz riječnik podataka šta ima u recyclebin- u
 - `select *`
`from recyclebin;`

Indeksi

- Pregled indeksa čiji je vlasnik novokreirani korisnik u svojstvu sys korisnika
 - `select *`
`from dba_indexes`
`where owner = 'KORISNIK';`
- Kao korisnik kreirati bitmap indeks nad kolonom pol za tablespace zadati inventory
 - `create bitmap index pol_ind on radnik (pol) tablespace inventory;`
- Pregledati naredbe za kreiranje različitih tipova indeksa!

Pogledi

- Kreirati pogled ime_prz_view nad tabelom radnik koji će prikazivati imena i prezimena svih radnika
 - `grant create view to korisnik;`
 - `create or replace view ime_prz_view (ime, prezime)`
`as`
`select ime, prz`
`from radnik;`

Sekvencer

- Kreirati sekvencer pod nazivom seq_mbr koji se uvećava za 10, počinje od 100, minimalna vrijednost je 10, nema maksimalnu vrijednost, nije cikličan i kešira 10 vrijednosti unaprijed
 - `create sequence seq_mbr
increment by 10
start with 100
minvalue 10
nomaxvalue
nocycle
cache 10;`

Sekvencer

- Dodati jedan red u tabelu radnik upotrebnom sekvencera seq_mbr
 - `insert into radnik values (SEQ_MBR.NEXTVAL, 'A', 'B', 50000, 'M');`
 - `commit;`
- Pregled poslednje generisane vrijednosti iz sekvencera
 - `select SEQ_MBR.CURRVAL
from sys.dual;`

Managing undo

Managing undo

- Instanca baze podataka kreira i održava informacije koje se koriste za rollback ili undo promjena nad bazom podataka
- Takve informacije se sastoje od zapisa akcija tokom transakcije prije nego što se uradi commit transakcije
- Ovakvi zapisi se nazivaju undo

Managing undo

- Undo zapisi se koriste za
 - rollback transakcija kada se izda naredba za rollback
 - oporavak (recovery) baze podataka
 - omogućava konzistentnost prilikom čitanja (read-consistency)
 - analizu podataka ranijeg vremenskog trenutka koristeći Oracle Flashback upite
 - oporavak od logičkih oštećenja koristeći Oracle Flashback mogućnosti
- Undo data – kako da se "vratimo nazad"
 - Undo data se čuva u undo segmentima koji se nalaze u undo tablespace- u
 - Data fajlovi od undo tablespace- a bi trebali biti autoextend kako ne bi došlo do pucanja transakcija zbog nedostatka prostora za undo data
- Redo data – kako da "idemo unaprijed" da reprodukujemo neke radnje
 - čuvaju se u redo log files

Managing undo

- Oracle omogućava potpuno automatizovane mehanizme za upravljanje undo informacijama i prostorom
- Sa automatskim undo upravljanjem instanca upravlja undo segmentima u undo tablespace- u
- Ova mogućnost je podrazumjevana
- Podrazumjevani tablespace u kom se čuvaju podaci je UNDOTBS1
- Undo retention period je minimalni vremenski period posmatran od sadašnjeg trenutka unazad u kom će biti zadržane stare undo informacije prije nego što budu prepisane

Managing undo

- Podešavanje undo retention perioda se rade modifikacijom inicijalizacionog parametra UNDO_RETENTION
- Pregled inicijalizacionih parametara baze podataka se vrši postavljanjem upita
 - `select *`
`from v$parameter;`
- Ukoliko želimo da vidimo sve undo inicijalizacione parametre postavićemo sledeći upit
 - `select *`
`from v$parameter`
`where name like '%undo%';`

Managing undo

- Ukoliko je naš serverski parametarski fajl u tekstualnom obliku potrebno je pronaći ovaj parametar i zadati mu novu vrijednost, a zatim odraditi restart (shutdown/startup) instance da bi se učitala nova vrijednost
 - UNDO_RETENTION=1800
- Ukoliko je potrebno momentalno postaviti novu vrijednost za undo retention period to se može odraditi upotrebom sledeće select naredbe
 - `alter system set undo_retention=1800;`
 - Prilikom izvršavanja ove naredbe može se pridodati atribut scope koji za vrijednost može uzeti:
 - SPFILE – vrijednost inicijalizacionog parametra se upisuje u parametarski fajl i biće učitana tokom prvog idućeg pokretanja instance
 - MEMORY – vrijednost inicijalizacionog parametra se upisuje u memorijskog područje i biva korištena do prvog gašenja instance, a nakon ponovnog pokretanja uzima se vrijednost koja je bila ranije (koja je zapisana u spfile)

Managing undo

- Ukoliko je potrebno momentalno postaviti novu vrijednost za undo retention period to se može odraditi upotrebom sledeće select naredbe
 - Ukoliko se izostavi SCOPE atribut onda će se nova vrijednost za parametar upisati u spfile, a i u memorijsko područje te će se upotrebljavati sve do gašenja instance, a i nakon prvog idućeg pokretanja
- Voditi računa da vrijednost parametra ne može biti proizvoljna nego je uslovljena konfiguracijom undo tablespace- a!

Database security

Database security

- Bezbjednost postićemo dodavanjem samo neophodnih privilegija korisnicima, kompleksnom autentifikacijom i praćenjem aktivnosti korisnika
- Praćenje aktivnosti korisnika se vrši pregledanjem audit logova u svojstvu administratora baze podataka (sys) odnosno korisnika koji ima privilegije za to
- Lokacija skladištenja takvih logova određena je inicijalizacionim parametrom `AUDIT_TRAIL`
- Moguće vrijednosti za `AUDIT_TRAIL` parametar su
 - DB – praćenje se upisuje u bazu podataka u tabele `SYS.AUD$`
 - XML – praćenje se upisuje u XML fajlove na operativnom sistemu
 - OS – praćenje se upisuje direktno na operativni sistem

Database security

- Pregled inicijalizacionih parametara vezanih za praćenje (audit)
 - `select *`
`from v$parameter`
`where name like '%audit%';`
- Promjena da se zapisi skladište direktno na operativnom sistemu
 - `alter system set audit_trail=os scope=spfile;`
- Promjena destinacije u koju se skladišti praćenje
 - `alter system set audit_trail_dest='<putanja>' scope=spfile;`

Database security

- Primjer podešavanja praćenja objekata
 - `audit select, insert, update, delete on hr.departments;`
- Odraditi select kao hr ili korisnik nad tabelom hr.departments
- Pregledati audit logove
 - ako su na OS- u pronaći fajl
 - ako su u bazi podataka izvršiti
 - `select *`
`from dba_audit_trail;`

Listener

Listener

- Listener predstavlja vezu između Oracle instance i korisnika
- Dodati novi listener upotrebom Net Configuration Assistant- a

- Zadatak za samostalnu izradu:
 - Povezati novi listener sa instancom baze podataka

Proactive Maintenance

Proactive Maintenance

- Automatic Workload Repository (AWR) je Oracle repozitorijum u kojem se smeštaju statistike koje će se kasnije analizirati, a na osnovu kojih će se davati preporuke za optimizaciju baze podataka
- Analiziranjem statistika mogu se npr. pokrenuti redovni zadaci za backup podataka
- Neki zadaci mogu automatski da se obave, a neki zahtevaju intervenciju administratora
- Statistike se koriste i od strane Query Optimizer-a da odabere najbolji Execution Plan
- Pokrenuti naredbu
 - `select num_rows from dba_tables where owner='KORISNIK' and table_name='RADNIK';`
- Dodati 10 novih redova u tabelu korisnik.radnik u PDB bazi kao sys korisnik i ponovo pokrenuti naredbu iznad
- Efekti?

Proactive Maintenance

- Statistike se prikupljaju povremeno, ali mogu i manuelno da se pokrenu
- Napraviti PL/SQL blok

- BEGIN

```
        DBMS_STATS.GATHER_TABLE_STATS('KORISNIK', 'RADNIK');
```

```
    END;
```

- Pokrenuti upit ponovo
- Kontrola automatskog prikupljanja statistika pomoću

- BEGIN

```
        DBMS_AUTO_TASK_ADMIN.[ENABLE|DISABLE] (  
            client_name => 'auto optimizer stats collection',
```

```
            operation    => NULL,
```

```
            window_name => NULL
```

```
        );
```

```
    END;
```

Proactive Maintenance

- Upit u data dictionary da bi se vidjele promjene
 - `COL CLIENT_NAME FORMAT a31;`
 - `SELECT CLIENT_NAME, STATUS
FROM DBA_AUTOTASK_CLIENT
WHERE CLIENT_NAME = 'auto optimizer stats collection';`
- Moguće podešavati vremenski okvir prikupljanja statistika pomoću
 - `BEGIN
DBMS_SCHEDULER.SET_ATTRIBUTE (
 'MONDAY_WINDOW',
 'repeat_interval',
 'freq=daily;byday=MON;byhour=05;byminute=0;bysecond=0'
);
END;`

Perfomance Management

Performance Management

- Performanse rada instance baze podataka se mogu pratiti kroz WEB aplikaciju Oracle Enterprise Manager Database Express
- Ovoj aplikaciji pristupa se preko adrese
 - <https://localhost:5500/em>
- Kredencijali za logovanje
 - username: sys
 - password: oracle
 - container name: ostaviti prazno

Performance Management

- Performance
 - Performance Hub
 - praćenje aktivnosti vezanih za sesije
 - praćenje upotrebe resursa poput CPU i broja I/O
 - SQL Performance Analyzer
- SQL Tuning Advisor
 - Preporuke za prikupljanje statistika, ispravljanje grešaka u procjenama izvršavanja SQL naredbi, daje preporuke da se kreira indeks, daje preporuke za izmjenu execution plana...

Execution plan

- Prikaz execution plan- a
- Logovati se kao sys u PDB pomoću SQL Developer- a
- Uključiti autotrace režim da bi se prikazali rezultati SQL naredbe i execution plan
 - `set autotrace [traceonly | on];`
- Izvršiti sledeći upit (kao script, ne kao statement)
 - `select /*+gather_plan_statistics*/ count(*)
from hr.employees
where salary=2800;`

Execution plan

- Logovati se kao sys u PDB pomoću sqlplus- a
 - `set autotrace [traceonly | on];`
- Izvršiti sledeći upit
 - `select count(*)
from hr.employees
where salary=2800;`

Execution plan

- U execution plan- u se vidi da je full table scan morao da se odradi da bi se prebrojali radnici
- Sort aggregate ne znači da se radio sort, već se on prikazuje kada se radi agregacija na nivou cijelih redova
- Kreirati običan index nad hr.employees salary
 - `create index sal_ind on hr.employees(salary);`
- Ponoviti upit. Rezultat?
 - U execution plan- u se vidi da je upotrebljen indeks
 - Sort aggregate je opet ispisan, ali su prebrojani zapravo cijeli slogovi iz indeksa

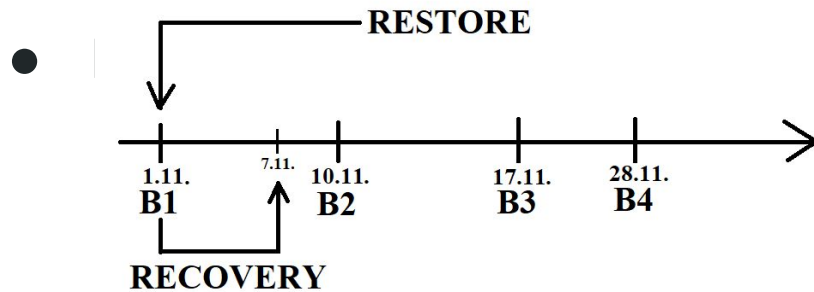
Execution plan

- Autotrace ostaje uključen dok traje sesija, ako ga je potrebno isključiti ranije treba izvršiti
 - `set autotrace off;`
- Da bi obični korisnici baze podataka mogli da prate execution plan morali bi imati odgovarajuće privilegije
- Zadatak – prebrojati osobe ženskog pola iz tabele korisnik.radnik
 - Ako je postojao indeks nad kolonom pol u tabeli korisnik.radnik obrisati ga
 - Analizirati execution plan naredbe
 - Dodati bitmap indeks
 - Ponovo prebrojati osobe ženskog pola
 - Prikazati sve osobe iz tabele korisnik.radnik koje su ženskog pola.
 - Efekti?

Backup and Recovery

Backup and Recovery

- Administratori bi trebali obezbjediti što ređe pojavljivanje grešaka
- Nakon pojavljivanja greške ili otkaza sistema administratori treba da obezbjede što brži



- Da bi se postigao što lakši, brži i kvalitetniji oporavak potrebno je:
 - redovno raditi backup
 - praviti kopije kontrolnih fajlova (na različitim lokacijama/diskovima)
 - praviti kopije log fajlova (na različitim lokacijama/diskovima)
 - arhivirati redo log fajlove

Dodavanje novog control fajla

- Logovati se kao sys u CDB putem SQL Developer- a
 - Pronaći putanju do control file- ova i otvoriti taj direktorijum
 - `select *`
`from v$controlfile;`

Dodavanje novog control fajla

- Logovati se kao sys u CDB putem sqlplus- a
 - ugasiti instancu da ostane u konzistentnom stanju
 - `shutdown immediate;`
 - kreirati direktorijum CF_COPY u C:\ i kopirati jedan od control file- ova u ovaj direktorijum, ali ga preimenovati (npr. CF1_COPY.CTL)
 - `startup nomount;`
 - trebamo startovati instancu u ovom režimu rada jer control files još uvijek nisu pokrenuti
 - `select value`
`from v$parameter`
`where name='control_files';`
 - sa ovom naredbom ćemo vidjeti putanje do trenutni aktivnih control file- ova

Dodavanje novog control fajla

- Logovati se kao sys u CDB putem sqlplus- a – nastavak
 - `shutdown immediate;`
 - gasimo ponovo da bismo dodali još jedan control file
 - `create pfile from spfile;`
 - kreiramo parametarski tekstualni fajl od parametarskog binarnog fajla (kog održava oracle)
 - novi pfile bi trebao da ima naziv initorcl.ora (init[SID].ora) i trebao bi se nalaziti na lokaciji `C:\oracle21c_setup\app\oracle\database\`
 - otvoriti pfile putem notepad- a i pronaći sa find liniju u kojoj piše lokacija do trenutnog control file- a (ako nije vidljivo)
 - dodati lokaciju do kopiranog control fajla
 - u liniji u kojoj stoji lokacija do trenutnog control file- a dodati
 - `;'C:\CF_COPY\CF1_COPY.CTL'`

Dodavanje novog control fajla

- Logovati se kao sys u CDB putem sqlplus- a – nastavak
 - sačuvati fajl
 - `create spfile from pfile;`
 - kreiranje parametarskog fajla kog održava oracle preko parametarskog fajla koji je ljudski čitljiv
 - `startup open;`
 - provjeriti da li postoji lokacija do novog control file- a
 - `select *
from v$controlfile;`

Redo log fajlovi

- Redo log fajlovi pamte promjene nastale izvršavanjem transakcija i internih serverskih aktivnosti u bazi podataka
- Moraju se čuvati u grupama kako bi se osiguralo njihovo postojanje uslijed otkaza
- Svaka grupa ima jedan redo log file i nekoliko njegovih kopija
- Svaka instanca mora da ima bar dva redo log file- a da bi operacija pisanja uvijek bila dostupna (u jedan redo log se piše, a drugi se arhivira)

Redo log fajlovi

- Proces LogWriter puni sve fajlove u jednoj grupi dok ih ne popuni ili dok se ne desi log switch nakon čega prelazi na sledeću grupu i tako ciklično se vraća na prvu grupu i prepisuje stari sadržaj
- Logovati se kao sys u CDB putem SQL Developer- a
 - `select *`
`from v$log;`
 - pregled koje redo log grupe postoje
 - `select *`
`from v$logfile;`
 - pregled koji redo log fajlovi postoje;

Redo log fajlovi

- Logovati se kao sys u CDB putem SQL Developer- a – nastavak
 - kreirati direktorijum RedoLogs u C:\
 - dodati novi redo log fajl u grupu koja trenutno nije aktivna grupa
 - alter database
add logfile member 'C:\RedoLogs\RL_F_06.LOG' to group 2;
 - dodati novu redo log grupu sa dva fajla
 - alter database
add logfile group 4 ('C:\RedoLogs\RL_F_04.LOG',
'C:\RedoLogs\RL_F_08.LOG')
size 200M
blocksize 512;

Redo log fajlovi

- Logovati se kao sys u CDB putem SQL Developer- a – nastavak
 - brisanje fajla iz redo log grupe (samo ako nije aktivna)
 - alter database
drop logfile member 'C:\RedoLogs\RL_F_06.LOG';
 - brisanje redo log grupe sa pripadajućim fajlovima (samo ako nije aktivna)
 - alter database
drop logfile group 4;
 - eksplicitno prebacivanje na narednu grupu
 - alter system switch logfile;

Redo log fajlovi

- Voditi računa o imenovanju redo log fajlova, ako započinju sa prefiksima poput O1_MF_2_% vjerovatno će naredba biti odbijena jer fajlovi koje imaju takav prefiks su OMF (oracle managed fajlovi)
- Prilikom dodavanja fajla u grupu voditi računa da veličina fajla bude ista kao kod ostalih fajlova iz grupe (podrazumjevano ponašanje)

Arhivski režim

- Archiver je pozadinski proces koji služi da sačuva automatski redo log fajlove prije nego što njihov sadržaj bude pregažen
- Da bi ovako nešto bilo moguće ARCHIVELOG režim mora biti uključen

Arhivski režim

- Logovati se kao sys user u CDB pomoću sqlplus- a
 - `archive log list;`
 - dobijamo status archivelog režima
 - Log sequence je kao ID redo log grupe
 - `shutdown immediate;`
 - Moramo ugasiti bazu podataka da bismo mogli da uspostavimo novi režim
 - `startup mount;`
 - archivelog režim se podešava u MOUNT režimu rada
 - `alter database archivelog;`
 - prebacivanje u archive log režim
 - ako je potrebno da se ovaj režim ugasi piše se `alter database noarchivelog;`
 - `alter database open;`
 - `archive log list;`
 - provjera da li je postavljen dobar režim

Arhivski režim

- Logovati se kao sys u CDB putem SQL Developer- a
 - `select *`
`from v$archive_dest;`
 - pregled svih predefinisanih lokacija na kojima se arhivirani redo log fajlovi trebaju skladištiti
 - napraviti direktorijum ArchiveLogs u C:\
 - `alter system set`
`log_archive_dest_2='location=C:\ArchiveLogs';`
 - postavljanje dodatne lokacije za skladištenje arhiviranih redo log fajlova
 - `alter system set`
`log_archive_dest_1='location=C:\oracle21c_setup\app\oracle\homes\OraDB21Home1\rdbms';`
 - ukoliko se izgubi prethodna lokacija

Arhivski režim

- Dodati redove u tabelu radnik u PDB bazi pa odraditi commit ponoviti ovo dva puta
- Prebaciti se u CDB i napisati naredbu
 - `alter system switch logfile;`
- Pregledati sadržaj direktorijuma u kom se skladište arhivirani redo log fajlovi
- Tokom instalacije je bila ponuđena opcija da se ovaj režim uključi

Fast Recovery Area

- Fast Recovery Area predstavlja lokaciju na disku kojom upravlja instanca baze podataka
- Instanca obezbeđuje automatsko čuvanje i upravljanje bekapima i fajlovima koji su predviđeni za oporavak kao što su
 - archived redo logovi, bekap setovi, flashback logovi
- U ranijim verzijama Fast Recovery Area je imao naziv Flash Recovery Area
- Tokom instalacije je bila ponuđena opcija da se specificira lokacija za fast recovery area

Fast Recovery Area

- Da bi se naknadno omogućila upotreba fast recovery area potrebno je specificirati dva parametra
 - `DB_RECOVERY_FILE_DEST_SIZE` i
 - `DB_RECOVERY_FILE_DEST`
- Pregled vrijednosti parametara za recovery
 - `show parameter db_recovery_file;`
- Pregled informacija o destinaciji za fast recovery area
 - `select *
from v$recovery_file_dest;`

Fast Recovery Area

- Da bi se Fast Recovery Area mogla upotrebljavati potrebno je prvo zadati veličinu destinacije
 - `alter system set db_recovery_file_dest_size=10G scope=both;`
- Nakon zadavanja veličine, potrebno je zadati lokaciju koja će se koristiti za skladištenje
 - `alter system set
db_recovery_file_dest='C:\oracle21c_setup\app\oracle\fast_recovery_area'
scope=both;`
- Provjeriti vrijednosti parametara

Oporavak nakon gubitka control file- a

- Logovati se kao sys user u CDB pomoću sqlplus- a
 - prikazati listu controlfile- ova
 - ukoliko postoji samo jedan controlfile neophodno je napraviti dodatni control file
 - `select *`
`from v$controlfile;`
 - obrisati control file sa lokacije `$ORACLE_BASE/oradata/ORCL/CONTROLFILE`
 - prikazati status u kom se nalazi instanca baze podataka
 - `select status`
`from v$instance;`
 - očekivano ponašanje nakon ove naredbe je da će SUBP poslati poruku da ne može da pronađe controlfile

Oporavak nakon gubitka control file- a

- Logovati se kao sys user u CDB pomoću sqlplus- a
 - pogledati u alert log fajlu (na dnu) poruke koje je poslao SUBP
\$ORACLE_BASE/diag/rdbms/orcl/orcl/alert/log.xml
 - ukoliko se ne javljaju poruke da controlfile nedostaje izvršiti naredbu za gašenje instance baze podataka
 - shutdown immediate;
 - da bi se riješio problem neophodno je prekopirati controlfile sa backup lokacije u direktorijum u kom nedostaje i potrebno je uraditi preimenovanje u originalni naziv koji može da se pročita iz alert log fajla
 - probati sa gašenjem instance, a nakon toga i sa paljenjem
 - shutdown immediate;
 - startup open;

Oporavak nakon gubitka redo log file- a

- Logovati se kao sys user u CDB pomoću SQL Developer- a
 - potrebno je obezbjediti da trenutno bude aktivna grupa koja sadrži više redo log fajlova
 - obrisati jedan redo log file
 - instanca baze podataka bi trebala nastaviti normalno sa radom
 - pogledati u alert logu da li postoji poruka da jedan redo log file nedostaje
 - izvršiti promjenu trenutne redo log grupe dok god se ponovo ne dođe do iste
 - pogledati sada alert logove
 - prekopirati alert log fajl koji nije obrisani i uraditi preimenovanje u originalni naziv obrisanog fajla

Oporavak nakon gubitka datafile- a

- Ako je instanca baze podataka u NOARCHIVELOG režimu onda bi trebalo odraditi restore na poslednji backup i onda korisnik treba da unese sve izmjene ručno da bismo doveli bazu podataka u stanje koje je bliže sadašnjem vremenskom trenutku
- Ako je instanca baze podataka u ARCHIVELOG režimu i izgubimo fajl koji nije u SYSTEM ili UNDO tablespace- u, onda možemo da uradimo restore na poslednji backup i recovery
- Više o oporavku datafile- ova u narednim lekcijama

Flashback

- Recovery (oporavak) može biti veoma sport
- Flashback mehanizam je brži i jednostavniji
- Radi na nivou slogova i transakcija
- Najčešće služi da razriješi greške nastale ljudskim faktorom, pogrešnim unosima od strane korisnika
- Flashback može da vrati cijelu bazu podataka u neki prethodni trenutak u vremenu tako što uradi undo svih promjena, može da vrati tabelu u neko prethodno stanje u vremenu ili da vrati obrisanu tabelu

Flashback table

- Logovati se kao sys user u PDB pomoću SQL Developer- a
 - da bi se odradio flashback tabele, mora biti uključena opcija Row Movement za tabelu
 - `alter table korisnik.radnik enable row movement;`
 - dodati tri nova reda u tabelu radnik, zapamtiti vrijeme izmjene i prikazati broj radnika u tabeli
 - Uraditi flashback tabele u vremenski trenutak prije dodavanja redova (prilagoditi tačno vrijeme u naredbi ispod)
 - `flashback table korisnik.radnik to timestamp to_timestamp('2023-11-28 9:09:00', 'YYYY-MM-DD HH:MI:SS');`
 - efekti?
 - umjesto timestamp- a može da se iskoristi i SCN (System Change Number)

Flashback table

- Logovati se kao sys user u PDB pomoću SQL Developer- a
 - pregledati stanje u recycle bin- u
 - `select *`
`from recyclebin;`
 - dropovati tabelu radnik
 - `drop table korisnik.radnik;`
 - pregledati stanje u recycle bin- u
 - prebrojati radnike u tabeli - javiće da tabela ne postoji
 - da bi se vratila tabela potrebno je pokrenuti naredbu
 - `flashback table korisnik.radnik to before drop;`
 - efekti?
 - pregledati recycle bin, prebrojati radnike
- Samostalno pogledati kako da se egzaktan objekat iz recycle bin- a vrati (po nazivu koji je dobio u recycle bin- u)

Flashback Query

- Možemo vidjeti kakve podatke je tabela imala u nekom trenutku ranije
- Logovati se kao sys user u PDB pomoću SQL Developer- a
 - dodati tri nova reda u tabelu radnik i zapamtiti vrijeme izmjene
 - prikazati sve radnike u tabeli
 - `select * from korisnik.radnik;`
 - prikazati sve radnike, ali neposredno prije izmjene podataka u tabeli
 - `select * from korisnik.radnik as of timestamp to_timestamp('2023-11-28 9:39:00', 'YYYY-MM-DD HH:MI_SS');`

Kraj!

Hvala na pažnji!