1. შექმენი ტერმინალიდან დირექტორია exam\_btu\_გვარი. შექმენი ცხრილ სივრცე გვარი\_tablesp ორი დატა ფაილებთან გვარი\_data1.dbf და გვარი\_data2.dbf ერთად ლოკაცია იყოს exam\_btu\_გვარი დირექტორია ზომები 20 და 30 მეგაბაიტი.

სკრიპტი:

create tablespace chakhvadze\_tablesp

DATAFILE '/home/oracle/exam\_btu\_chakhvadze/chakhvadze\_data1.dbf' size 20m,

'/home/oracle/exam\_btu\_chakhvadze/chakhvadze\_data2.dbf' size 30m ;

2. შექმენი მომხმარებელი სახელით შენი სახელი პაროლი შენი გვარი დეფაულტ ცხრილსივრცე გვარი\_tablesp. მიანიჭე როლური პრივილეგია dba. მიანიჭე სისტემური პრივილეგიები ცხრილის შექმნა, სინონიმის შექმნა, ხედის და მიმდევრობის შექმნა. მიეცით hr.employees ცხრილზე მოსელექტების და დააპდეითების ობიექტ პრივილეგია.

სკრიპტი:

create user aleksandre IDENTIFIED by chakhvadze

default TABLESPACE chakhvadze\_tablesp QUOTA UNLIMITED on chakhvadze\_tablesp;

grant dba to aleksandre;

grant create table, create synonym , create view, create sequence to aleksandre;

grant select, update on hr.employees to aleksandre;

3. მიქონექთი შენი სახელის უზერით. ქონექშენის სახელი იყოს შენი სახელი\_გვარი.

შექმენი ცხრილი exam რომელსაც ექნება 2 სვეტი exam\_id რიცხვითი 4 თანრიგიანი პირველადი გასღები და exam\_name სტრქიონული 20 სიმბოლო მაქსიმუმ უნიკალური.

შექმენი ცხრილი students რომელსაც ექნება 3 სვეტი id რიცხვითი 4 თანრიგიანი პირველადი გასაღები ავტომატურად ზრდადი, last\_name სტრქიონული 20 სიმბოლო მაქსიმუმ არანალიზებული, exam\_id რიცხვითი 4 თანრიგიანი მეორადი გასაღები exam ცხრილის exam\_id-ით შეზღუდული.

სკრიპტი:

create table exam (exam\_id number(4) primary key, exam\_name varchar2(20) unique);

create table students (id number(4) GENERATED BY DEFAULT ON NULL AS IDENTITY primary key,

last\_name varchar2(20) not null, exam\_id number(4) REFERENCES exam(exam\_id));

4. შექმენი მიმდევრობა გვარი\_seq სტარტით 10 ბიჯი 2 მაქიმალური 400. შექმენი ტრიგერი გვარი\_trig exam ცხრილის exam\_id პირველადი გასაღები სვეტისათვის.

შეიტანე 2 ჩანაწერი exam ცხრილში:

Excel

Python

შეიტანე 2 ჩანაწერი students ცხრილში:

Donadze, 10

Lebanidze, 12

სკრიპტი:

create sequence chakhvadze\_seq start with 10 increment by 2 maxvalue 400;

create or replace trigger chakhvadze\_trig

before insert on exam

for each row

begin

select chakhvadze\_seq.nextval into :new.exam\_id from dual;

end;

/

insert into exam(exam\_name) values('Excel');

insert into exam(exam\_name) values('Python');

insert into students(last\_name,exam\_id) values('Donadze',10);

insert into students(last\_name, exam\_id) values('Lebanidze',12);

commit;

5. შექმენი ცხრილი სახელად შენი გვარი\_tab, რომელსაც ექნება 1 სვეტი lastname სტრიქონული მაქსიმუმ 30 სიმბოლო. შეიტანე შენი გვარი. მოახდინე ამ ცხრილის ისე წაშლა რომ ჩავარდეს ნაგვის ყუთში. მოახდინე ნაგვის ყუთიდან მისი აღდგენა (აქ წარმოდგენილია სამივე ვარიანტი გამოცდაზე შეგხვდება მხოლოდ ერთ-ერთი) 1. ორიგინალი სახელის მეშვეობით, 2. ობიექტ ნეიმის მესვეობით, 3. რომ სახელი ერქვას examtab\_შენი გვარი.

სკრიპტი:

create table chakhvadze\_tab (lastname varchar2(30));

insert into chakhvadze\_tab values('Chakhvadze');

commit;

drop table chakhvadze\_tab;

select \* from user\_recyclebin;

--1

flashback table chakhvadze\_tab to before drop;

--2

flashback table "BIN$/nZUG/iWKOTgUwEAAH8g/g==$0" to before drop;

--3

flashback table "BIN$/nZUG/iXKOTgUwEAAH8g/g==$0" to before drop RENAME to examtab\_chakhvadze;

6. შექმენი ცხრილი სახელად შენი გვარი, რომელსაც ექნება 1 სვეტი lastname სტრიქონული მაქსიმუმ 30 სიმბოლო. შეიტანე ჩანაწერები

Burkadze

Gagua

1. დააფიქსირე ამჟამინდელი სისტემური ცვლილების ნომერი (ან აქ ამ სისტემური ცვლილების ნომერის მაგივრად იქნება შექმენი რესპორ პოინტი შენი გვარი\_point). 2. შექმენი რესტორ პოინტი შენი გვარი\_point.

შეიტანე ჩანაწერი

Kurtanidze

1. გამოიტანე მონაცემები ბაზაში სისტემური ცვლილების ნომრის მდგომარეობით. 2. აღადგინე ბაზაში რესტორ პოინტის მდგომარეობა.

სკრიპტი:

create table chakhvadze (lastname varchar2(30));

insert into chakhvadze values ('Burkadze');

insert into chakhvadze values ('Gagua');

commit;

--1

select current\_scn from v$database; --5368508

--2

create RESTORE POINT chakhvadze\_point;

insert into chakhvadze values ('kurtanidze');

commit;

--1

select \* from chakhvadze as of scn 5368508;

--2

alter table chakhvadze enable row movement;

flashback table chakhvadze to restore point chakhvadze\_point;

7. მოასელექტე hr.employees ცხრილი. GUI-გარემოდან მოახდინე ჩანაწერების ექსპორტი csv (dsv დელიმიტატორი იყოს tab, text) ფაილად დექსტოპზე დაარქვი შენი გვარი\_exam. მოახდინე csv (dsv, text) შენი გვარი\_exam ფაილიდან მონაცემების იმპორტი ბაზაში ცხრილს დაერქვას სახელი t#შენი გვარი.

სკრიპტი: select \* from hr.employees;

8. შექმენი ცხრილი t$შენი გვარი hr.employees ცხრილზე დაყრდნობით. ტერმინალური expdp უტილიტას დახმარებით გამოაექსპორტეთ შენი მომხმარებლით და პასვორდით t$შენი გვარი ცხრილი დირექტორიაში data\_dump\_dir, დამპ ფაილი სახელი იყოს exp\_table\_გვარი.dmp, log იყოს exp\_გვარი\_log.log. ტერმინალური Impdp უტილიტას დახმარებით შეაიმპორტეთ შენი მომხმარებლით და პასვორდით t$შენი გვარი ცხრილი შეცვლილი სახელით გვარი\_emp დირექტორიიდან data\_dump\_dir, დამპ ფაილი სახელი exp\_table\_გვარი.dmp, log იყოს imp\_გვარი\_log.log.

სკრიპტი:

expdp aleksandre/chakhvadze tables=t\$chakhvadze directory=data\_pump\_dir dumpfile=exp\_table\_chakhvadze.dmp logfile=exp\_chakhvadze\_log.log

აქ ყურადღება მიაქციეთ t\$chakhvadze-ში \ ნისანს რადგან ტერმინალში დოლარის ნიშანი გამოიყენება ცვლადებთან სამუშაოდ ამიტომ მას უნდა დაეკარგოს პირველადი შინაარსი.

impdp aleksandre/chakhvadze tables=t\$chakhvadze directory=data\_pump\_dir dumpfile=exp\_table\_chakhvadze.dmp logfile=imp\_chakhvadze\_log.log remap\_table=t\$chakhvadze:chakhvadze\_emp

9. შექმენი ცხრილი sales, რომელსაც ექნება სვეტები sales\_id რიცხვითი პირველადი გასაღები, sales\_amt რიცხვითი, region\_id რიცხვითი და sales\_dtt თარიღის ტიპის. შეიტანეთ მასში 1 სქრიქონი და დაარეზერვეთ. სქრიქონები: 1,101,10,sysdate-10; შექმენით დარეფრეშებადი მატერიალიზებული ხედი სახელით sales\_daily\_mv შექმნილ sales ცხრილზე დაყრდნობით. შეიტანეთ 1 დამატებითი ჩანაწერი sales ცხილში. 4,99,200,sysdate; დაარეფრეშეთ მატერიალიზებული ხედი complete რეჯიმით dbms\_mview pl/sql პეკიჯის გამოყენებით. გამოიტანეთ ჩანაწერები მატერიალიზებული ხედიდან;

სკრიპტი:

create table sales(

sales\_id number primary key ,sales\_amt number ,region\_id number ,sales\_dtt date );

insert into sales values(1,101,10,sysdate-10);

commit;

create materialized view sales\_daily\_mv

segment creation immediate refresh

complete on demand as

select sum(sales\_amt) sales\_amt ,trunc(sales\_dtt) sales\_dtt

from sales

group by trunc(sales\_dtt);

insert into sales values(4,99,200,sysdate);

commit;

exec dbms\_mview.refresh('SALES\_DAILY\_MV','C');

select \* from sales\_daily\_mv;

10. მოახდინეთ მომხმარებლისა და ქონექშენის წაშლა, მომხმარებელი თავისი ყველა ობიექტიანად იქნეს მოცილებული. წაშალე ცხრილ სივრცე გვარი\_tablesp. წაშალე დირექტორია exam\_btu\_გვარი.

სკრიპტი:

drop user aleksandre cascade;

drop TABLESPACE chakhvadze\_tablesp INCLUDING CONTENTS and DATAFILES CASCADE CONSTRAINTS;

rm -rf exam\_btu\_chakhvadze/