1. (2 ქულა) დაუკავშირდით ორაკლ სერვერს როგორც პრივილეგირებული მომხმარებელი system და შექმენით როლი სკრიპტულად თქვენი გვარი\_role და მიანიჭეთ შემდეგი სისტემური პრივილეგიები create session, create any table, create any view, create any index, create any sequence, create any synonym, alter any table, alter any index, alter any sequence, drop any table, drop any view, drop any index, drop any sequence, drop any synonym. მიანიჭეთ ობიექტ პრივილეგია მოსელექტება hr სქემის employees ცხრილზე. მიანიჭეთ როლი dba შექმნილ როლს.

სკრიპტი:

create role chakhvadze\_role;

grant create session, create any table, create any view, create any index, create any sequence,

create any synonym, alter any table, alter any index, alter any sequence, drop any table,

drop any view, drop any index, drop any sequence, drop any synonym

to chakhvadze\_role;

grant select on hr.employees to chakhvadze\_role;

grant dba to chakhvadze\_role;

2. (2 ქულა) სკრიპტულად შექმენი მომხმარებელი btu\_გვარი პაროლი btu, დეფაულტ ცხრილსივრცე users ულიმიტოდ, დროებით ცხრილსივრცედ temp, პროფაილი default. გადაამოწმეთ იუზერის არსებობა. მიანიჭეთ შექმნილი როლი ამ მომხმარებელს.

სკრიპტი:

create user btu\_chakhvadze identified by btu

default TABLESPACE users QUOTA UNLIMITED on users

TEMPORARY TABLESPACE temp

profile default;

select \* from dba\_users where username ='BTU\_CHAKHVADZE';

grant chakhvadze\_role to btu\_chakhvadze;

3. (1 ქულა) მიურთდით ბაზას როგორც მომხმარებელი btu\_გვარი. გადაამოწმეთ ვინაობა. გადაამოწმეთ თუ ფლობს რაიმე ობიექტს.

სკრიპტი:

show user

select \* from dba\_objects where owner='BTU\_CHAKHVADZ';

4. (2 ქულა) შექმენი ცხრილი test\_გვარი რომელსაც ექნება 4 სვეტი: პირველი id რიცხვითი, salary რიცხვითი (5,2) და მე-3 ვირტუალური სვეტი status, რომელიც შეყვანილი ხელფაისის მიხედვით დააგენერირებს სვეტის მნიშვნელობას თუ შეტანილი იქნება salary-ში 500 ან მასზე ნაკლები low, თუ 500-ზე მეტი good, მე-4 სვეტი comm სიმბოლური მაქსიმუმ 30 სიმბოლო უჩინარი. შექმენი sal\_idx ინდექსი salary სვეტზე.

სკრიპტი:

create table test\_chakhvadze (id number, salary number(5,2),

status generated always as (

case when salary <= 500 then 'low'

when salary > 500 then 'good' end), comm varchar2(30) invisible);

create index sal\_idx on test\_chakhvadze(salary);

5. (1 ქულა) შექმენით მიმდევრობა seq\_გვარი სახელით რომლის სტარტის წერტილია 10, ბიჯი 5, არ არის ციკლური, არის მოწესრიგებული, არ იყოს ქეშირებული, მაქსიმალური მნიშვნელობა 1000 და მინიმალური მნიშვნელობა 10.

სკრიპტი:

create sequence seq\_chakhvadze start with 10 INCREMENT BY 5

NOCACHE ORDER MAXVALUE 1000 MINVALUE 10;

6. (2 ქულა) შექმენი ტრიგერი test\_tr რომელიც ავტომატურ ზრდად id სვეტის მნიშვნელობებს დააგენერირებს ახალი ჩანაწერის შეტანისას test ცხრილში.

სკრიპტი:

create or replace trigger test\_tr

before insert on test\_chakhvadze

for each row

begin

select seq\_chakhvadze.nextval into :new.id from dual;

end;

/

7. (2 ქულა) შეიტანეთ test\_გვარი ცხრილში შემდეგი ჩანაწერები salary და comm სვეტებში

450, 'btu is the best'

120, 'cool exam'

150, 'this day'

580, 'perfect thing'

890, 'good idea'

900, 'great option'

სკრიპტი:

insert into test\_chakhvadze(salary, comm) values (450, 'btu is the best');

insert into test\_chakhvadze(salary, comm) values (120, 'cool exam');

insert into test\_chakhvadze(salary, comm) values (150, 'this day');

insert into test\_chakhvadze(salary, comm) values (580, 'perfect thing');

insert into test\_chakhvadze(salary, comm) values (890, 'good idea');

insert into test\_chakhvadze(salary, comm) values (900, 'great option');

commit;

8. (2 ქულა) შექმენით ქვემოთხოვნით ცხრილი emp სქემა hr-ის employees-ზე დაყრდნობით, საიდანაც აიღება მხოლოდ ის ჩანაწერები სადაც გვარის სიგრზე არის 7-სიმბოლოზე მეტი. შექმენით ქვემოთხოვნით ცხრილი emp2 ცხრილ emp-ზე დაყრდნობით მხოლოდ კონსტრუქცია და არა ჩანაწერებიც.

სკრიპტი:

create table emp as

select \* from hr.employees

where length(last\_name)>7;

create table emp2 as

select \* from emp

where 2=1;

select \* from emp;

select \* from emp2;

9. (2 ქულა) შექმენი my\_exam\_view ხედი emp ცხრილის პირველი 5 ჩანაწერით.

სკრიპტი:

create view my\_exam\_view as

select \* from emp where rownum<=5;

10. (2 ქულა) შექმენი სინონიმები test\_გვარი ცხრილზე t\_chakh (t\_გვარის პირველი 3 ასო) სახელით და my\_exam\_view ხედზე vu სახელით. გადაამაწმეთ თუ რა ობიექტებს ფლობს მომხმარებელი.

სკრიპტი:

create synonym t\_chakh for test\_chakhvadze;

create synonym vu for my\_exam\_view;

11. (2 ქულა) a) შეიტანეთ ახალი ჩანაწერი ცხრილში test\_გვარი 300, 'gtu forever'; b) დაამოდიფიცირეთ ჩანაწერი რომლის აიდია 20 comm სვეტში არსებულს მიუკონკატენირეთ თრანქირებული სისტემური თარიღი; c) წაშალეთ ის სტრიქონები რომლებშიც comm სვეტის სიგრძე არის 2-ის ჯერადი; d) დაამატეთ სვეტი manager რიცხვითი; e) სვეტი manager გახადეთ სიმბოლური მაქსიმალური 50 სიმბოლო; f) გადაარქვით სახელი manager სვეტს man სახელით; g) ამოშალეთ სვეტი man; h) გახადეთ გამოუყენებელი comm; i) გადაარქვით ცხრილს test\_გვარი სახელი დაარქვით გვარის საწყისი 5 ასო\_tab; j) გახადეთ ცხრილი მხოლოდ ამოკითხვადი.

სკრიპტი:

a)

insert into test\_chakhvadze(salary, comm) values (300, 'gtu forever');

commit;

b)

update test\_chakhvadze set comm=comm||trunc(sysdate)

where id = 20;

commit;

c)

delete from test\_chakhvadze where mod(id,2)=0;

commit;

d)

alter table test\_chakhvadze add manager number;

e)

alter table test\_chakhvadze MODIFY manager varchar2(50);

f)

alter table test\_chakhvadze rename column manager to man;

g)

alter table test\_chakhvadze drop column man;

h)

alter table test\_chakhvadze set UNUSED(comm);

i)

alter table test\_chakhvadze RENAME to chakhva\_tab;

j)

alter table chakhva\_tab read only;

12. (2 ქულა) შექმენით ახალი პროფილი exam\_prof\_გვარი რომელსაც ექნება შემდეგი პარამეტრები

COMPOSITE\_LIMIT default

SESSIONS\_PER\_USER 30

CPU\_PER\_SESSION unlimited

CPU\_PER\_CALL unlimited

LOGICAL\_READS\_PER\_SESSION 400000

LOGICAL\_READS\_PER\_CALL 10000

IDLE\_TIME 8

CONNECT\_TIME 200

PRIVATE\_SGA default

FAILED\_LOGIN\_ATTEMPTS 20

PASSWORD\_LIFE\_TIME 100

PASSWORD\_REUSE\_TIME 140

PASSWORD\_REUSE\_MAX 15

PASSWORD\_VERIFY\_FUNCTION null

PASSWORD\_LOCK\_TIME 3

PASSWORD\_GRACE\_TIME 18

სკრიპტი:

CREATE PROFILE exam\_prof\_chakhvadze LIMIT

COMPOSITE\_LIMIT default

SESSIONS\_PER\_USER 30

CPU\_PER\_SESSION unlimited

CPU\_PER\_CALL unlimited

LOGICAL\_READS\_PER\_SESSION 400000

LOGICAL\_READS\_PER\_CALL 10000

IDLE\_TIME 8

CONNECT\_TIME 200

PRIVATE\_SGA default

FAILED\_LOGIN\_ATTEMPTS 20

PASSWORD\_LIFE\_TIME 100

PASSWORD\_REUSE\_TIME 140

PASSWORD\_REUSE\_MAX 15

PASSWORD\_VERIFY\_FUNCTION null

PASSWORD\_LOCK\_TIME 3

PASSWORD\_GRACE\_TIME 18;

13. (1 ქულა) დაამოდიფიცირეთ btu\_გვარი იუზერის პარამეტრები პაროლი გახადეთ btu23 და პროფილი exam\_prof\_გვარი.

სკრიპტი:

alter user btu\_chakhvadze identified by btu23 profile exam\_prof\_chakhvadze;

14. (1 ქულა) გააქტიურე აუდიტი btu\_გვარი იუზერის განხორციელებულ ნებისმიერ ბრძანებაზე. ნახე აუდიტის ლოგი. აუდიტი გააუქმე.

სკრიპტი:

audit all statements by btu\_chakhvadze;

select \* from dba\_audit\_trail;

select \* from DBA\_STMT\_AUDIT\_OPTS;

select \* from SYS.AUD$;

noaudit all statements by btu\_chakhvadze;

15. (1 ქულა) წაშალეთ იუზერი btu\_გვარი. წაშალეთ როლი გვარი\_role. წაშალეთ პროფილი exam\_prof\_გვარი. გადაამოწმეთ არსებობს თუ არა ეს იუზერი.

სკრიპტი:

Drop user btu\_chakhvadze cascade;

Drop role chakhvadze\_role;

Drop profile exam\_prof\_chakhvadze;

select \* from dba\_users where username ='BTU\_CHAKHVADZE';