BHARATIYA VIDYA BHAVAN'S

SARDAR PATEL INSTITUTE OF TECHNOLOGY

MUNSHI NAGAR, ANDHERI (WEST), MUMBAI – 400 058, India
(Autonomous College Affiliated to University of Mumbai)

Executed

Name: Kalyani Chintaman Bharsat

Duration: 1 hour
Marks: 25 Marks
UCID: 2023510005

General Instructions:

Viva will be taken at the time of practical as well as after the practical if required.
The figures to the right indicate full marks.
If you are using any additional information, state it clearly.
Once you finish with the code show it to the examiner for testing. Write your answer in Word file and upload it on Moodle.

Q 1 A)	**Consider the bank database given below. The primary keys are underlined and the data types are specified:** **BRANCH**(branch_name:varchar2(30),branch_city:varchar2(30),assets:number(10)) **CUSTOMER**(customer_name:varchar2(30),customer_street:varchar2(30), customer_city:varchar2(30)) **ACCOUNT**(account_number:number(10),customer_name:varchar2(30), branch_name:varchar2(30), balance:number(10)) a) Create the above tables by properly specifying the primary keys and the foreign keys and named constraints. b) Enter atleast five tuples for each relation. c) Write SQL query to display details of customer whose living city is same as branch city. d) Write SQL query to find average balance of each branch. e) Write SQL query to truncate all the relations.	**10**
B)	Write PL/SQL code to give grant and revoke all the privileges to RMAN user on Account relation.	**10**
C)	List customer details, account details and branch details using join.	**5**

a:- Create table branch (branch_name varchar(30) primary key, branch_city varchar(30), assets int);
- Create table customer (customer_name varchar(30) primary key, customer_street varchar(30), customer_city varchar(30));
- Create table account (account_number int, customer_name varchar(30), branch_name varchar(30), balance int);
- alter table account add constraint fk1 foreign key (customer_name) references customer(customer_name);
- alter table account add constraint fk2 foreign key (branch_name) references branch(branch_name);

b. insert into branch values('Virar', 'Vasai', 100000), ('Andheri', 'Mumbai', 200000), ('GK', 'Palhar', 250000), ('Manvelpada', 'Borivali',

800000), ('DN nagar', 'kandiwali', 400000);
-insert into customer values('Kalyani', 'Ak road', 'Vasai'),
('Samruddhi', 'Sk road', 'Palhar'), ('Atharv', 'Mira road', 'Mumbai'),
('Swati', 'Dk road', 'vasal'), ('Sanika', 'SV road', 'Kandiwali');
-insert into ~~customer~~ account values (101, 'Kalyani', 'Virar', 10000), (203,
'Samruddhi', 'D N nagar', 20000), (303, 'Atharv', 'Gk', 200000),
(102, 'Swati', 'Andheri', 30000), (204, 'Sanika', 'Manvelpada',
400000);

C. Select * from customer as c join account as a on ((c.customer
_name = a.customer_name) join branch as b on (a.branch_name =
b.branch_name) where c.customer_city = b.branch_city;

d. select branch_name, avg(assets) as average_balance
from branch group by branch_name;

e. truncate table branch;
truncate table customer;
truncate table account;

B. Create definer ='root'@'localhost' procedure 'Grant_Revoke'
(permission varchar(20))
Begin
case permission
when 'grant' then
grant all privileges on bank_05.account to 'RMAn'@'localhost';
when 'revoke' then
revoke privileges on bank_05.account from 'RMAn'@'localhost';
end case;
End

→ Create user 'RMAn'@'localhost' identified by 'pass';
Call grant_revoke('grant');
Call grant_revoke ('revoke');
Rollback

C. Select * from customer as c left join account as a on
(c.customer_name} = a.customer_name) left join
branch as b on (a.branch_name = b.branch_name);