**Views**

**Admin view:**

create view v\_admin as

select a.admin\_id,a.admin\_name,a.admin\_password

from admin a;

**v\_employee10:**

create view v\_employee10 as

select e.emp\_id,e.emp\_name,e.emp\_number,d.depart\_id

from employee e, department d

where depart\_id=10;

**v\_employee20:**

create view v\_employee20

as select e.emp\_id,e.emp\_name,e.emp\_number,d.depart\_id

from employee e, department d

where depart\_id=20;

**v\_department:**

create view v\_department as

select e.emp\_id,d.depart\_id,e.emp\_name,d.depart\_name from

employee e, department d;

**v\_location:**

create view v\_location as

select location\_id,city,state\_province

from location;

**v\_jobs:**

create view v\_jobs as

select job\_id,job\_title,min\_salary,max\_salary

from jobs;

**v\_jobhistory:**

create view v\_jobhistory as

select e.emp\_id,jh.start\_date,jh.end\_date,j.job\_id,d.depart\_id

from employee e, jobs j, job\_history jh,department d;

**v\_training:**

create view v\_training as

select t.training\_id,t.training\_title,d.depart\_id

from training t,department d;

**v\_users:**

create view v\_users as

select user\_id, user\_password

from users;

**v\_salary:**

create view v\_salary as

select e.emp\_id,s.salary,s.bonus,s.loan

from employee e,salary s;

**v\_country:**

create view v\_country as

select c.country\_id,c.country\_name,r.region\_id

from country c, region r;

**v\_region:**

create view v\_region as

select region\_id,region\_name

from region;

**v\_attendance:**

create view v\_attendance as

select e.emp\_id, a.attendance\_id,a.attendance\_employee,a.attendance\_date

from employee e, attendance a;

**Procedures:**

**insert\_employee:**

create or replace procedure insert\_employee(

e\_id employee.emp\_id%type,

e\_name employee.emp\_name%type,

e\_address employee.emp\_address%type,

e\_number employee.emp\_number%type,

e\_email employee.emp\_email%type)

is

begin

insert into employee(emp\_id,emp\_name,emp\_address,emp\_number,emp\_email)

values(e\_id,e\_name,e\_address,e\_number,e\_email);

commit;

end;

/

**Insert\_admin:**

create or replace procedure insert\_admin(

a\_id admin.admin\_id%type,

a\_name admin.admin\_name%type,

a\_email admin.admin\_email%type,

a\_password admin.admin\_password%type)

is

begin

insert into admin (admin\_id,admin\_name,admin\_email,admin\_password)

values(a\_id,a\_name,a\_email,a\_password);

commit;

end;

/

**Insert\_department:**

create or replace procedure insert\_department(

d\_id department.depart\_id%type,

d\_name department.depart\_name%type,

e\_id department.emp\_id%type,

l\_id department.location\_id%type)

is

begin

insert into department(depart\_id,depart\_name,emp\_id,location\_id)

values(d\_id,d\_name,e\_id,l\_id);

commit;

end;

/

**Insert\_location:**

create or replace procedure insert\_location(

l\_id location.location\_id%type,

s\_address location.street\_address%type,

p\_code location.postal\_code%type,

city location.city%type,

state\_province location.state\_province%type)

is

begin

insert into location(location\_id,street\_address,postal\_code,city,state\_province)

values(l\_id,s\_address,p\_code,city,state\_province);

commit;

end;

/

**Insert\_job:**

create or replace procedure insert\_job(

j\_id jobs.job\_id%type,

j\_title jobs.job\_title%type,

min\_sal jobs.min\_salary%type,

max\_sal jobs.max\_salary%type)

is

begin

insert into jobs(job\_id,job\_title,min\_salary,max\_salary)

values(j\_id,j\_title,min\_sal,max\_sal);

commit;

end;

/

**Insert\_job\_history:**

create or replace procedure insert\_job\_history(

e\_id job\_history.emp\_id%type,

start\_date job\_history.start\_date%type,

end\_date job\_history.end\_date%type,

j\_id job\_history.job\_id%type,

d\_id job\_history.depart\_id%type)

is

begin

insert into job\_history(emp\_id,start\_date,end\_date,job\_id,depart\_id)

values(e\_id,start\_date,end\_date,j\_id,d\_id);

commit;

end;

/

**Insert\_training:**

create or replace procedure insert\_training(

t\_id training.training\_id%type,

t\_title training.training\_title%type,

t\_desc training.training\_description%type,

d\_id training.depart\_id%type)

is

begin

insert into training(training\_id,training\_title,training\_description,depart\_id)

values(t\_id,t\_title,t\_desc,d\_id);

commit;

end;

/

**Insert\_user:**

create or replace procedure insert\_user(

u\_id users.user\_id%type,

u\_pass users.user\_password%type)

is

begin

insert into users(user\_id,user\_password)

values(u\_id,u\_pass);

commit;

end;

/

**Insert\_salary:**

create or replace procedure insert\_salary(

sal\_id salary.salary\_id%type,

sal salary.salary%type,

bonus salary.bonus%type,

loan salary.loan%type,

last\_updated salary.last\_updated%type,

e\_id salary.emp\_id%type)

is

begin

insert into salary(salary\_id,salary,bonus,loan,last\_updated,emp\_id)

values(sal\_id,sal,bonus,loan,last\_updated,e\_id);

commit;

end;

/

**Insert\_country:**

create or replace procedure insert\_country(

c\_id country.country\_id%type,

c\_name country.country\_name%type,

r\_id country.region\_id%type)

is

begin

insert into country(country\_id,country\_name,region\_id)

values(c\_id,c\_name,r\_id);

commit;

end;

/

**Insert\_region:**

create or replace procedure insert\_region(

r\_id region.region\_id%type,

r\_name region.region\_name%type)

is

begin

insert into region(region\_id,region\_name)

values(r\_id,r\_name);

commit;

end;

/

**Insert\_attendance:**

create or replace procedure insert\_attendance(

atten\_id attendance.attendance\_id%type,

atten\_emp attendance.attendance\_employee%type,

atten\_date attendance.attendance\_date%type)

is

begin

insert into attendance(attendance\_id,attendance\_employee,attendance\_date)

values(atten\_id,atten\_emp,atten\_date);

commit;

end;

/

**Edit\_employee:**

create or replace procedure edit\_employee(

e\_id in employee.emp\_id%type,

e\_name in employee.emp\_name%type,

e\_address in employee.emp\_address%type,

e\_number in employee.emp\_number%type,

e\_email in employee.emp\_email%type)

is

begin

update employee set emp\_id=e\_id,emp\_name=e\_name,emp\_address=e\_address,emp\_email=e\_email where emp\_id=e\_id;

end;

/

**Edit\_admin:**

create or replace procedure edit\_admin(

a\_id in admin.admin\_id%type,

a\_name in admin.admin\_name%type,

a\_email in admin.admin\_email%type,

a\_pass in admin.admin\_password%type)

is

begin

update admin set admin\_name=a\_name,admin\_email=a\_email,admin\_password=a\_pass

where admin\_id=a\_id;

end;

/

**Edit\_department:**

create or replace procedure edit\_department(

d\_id in department.depart\_id%type,

d\_name in department.depart\_name%type,

e\_id in department.emp\_id%type,

l\_id in department.location\_id%type)

is

begin

update department set depart\_name=d\_name,emp\_id=e\_id,location\_id=l\_id

where depart\_id=d\_id;

end;

/

**Edit\_location:**

create or replace procedure edit\_location(

l\_id in location.location\_id%type,

s\_address in location.street\_address%type,

p\_code in location.postal\_code%type,

city in location.city%type,

s\_prov in location.state\_province%type)

is

begin

update location set street\_address=s\_address,postal\_code=p\_code,city=city,state\_province=s\_prov

where location\_id=l\_id;

end;

/

**Edit\_job:**

create or replace procedure edit\_job(

j\_id in jobs.job\_id%type,

j\_title in jobs.job\_title%type,

min\_sal in jobs.min\_salary%type,

max\_sal in jobs.max\_salary%type)

is

begin

update jobs set job\_title=j\_title,min\_salary=min\_sal,max\_salary=max\_sal

where job\_id=j\_id;

end;

/

**Edit\_job\_history:**

create or replace procedure edit\_job\_history(

e\_id in job\_history.emp\_id%type,

s\_date in job\_history.start\_date%type,

e\_date in job\_history.end\_date%type,

j\_id in job\_history.job\_id%type,

d\_id in job\_history.depart\_id%type)

is

begin

update job\_history set start\_date=s\_date,end\_date=e\_date,job\_id=j\_id,depart\_id=d\_id where emp\_id=e\_id;

end;

/

**Edit\_training:**

create or replace procedure edit\_training(

t\_id in training.training\_id%type,

t\_title in training.training\_title%type,

t\_des in training.training\_description%type,

d\_id in training.depart\_id%type)

is

begin

update training set training\_title=t\_title,training\_description=t\_des,

depart\_id=d\_id where training\_id=t\_id;

end;

/

**Edit\_users:**

create or replace procedure edit\_users(

u\_id in users.user\_id%type,

u\_pass in users.user\_password%type)

is

begin

update users set user\_password=u\_pass where user\_id=u\_id;

end;

/

**Edit\_salary:**

create or replace procedure edit\_salary(

s\_id in salary.salary\_id%type,

sal in salary.salary%type,

bonus in salary.bonus%type,

loan in salary.loan%type,

last\_updated in salary.last\_updated%type,

e\_id in salary.emp\_id%type)

is

begin

update salary set salary=sal,bonus=bonus,loan=loan,last\_updated=last\_updated, emp\_id=e\_id where salary\_id=s\_id;

end;

/

**Edit country:**

create or replace procedure edit\_country(

c\_id in country.country\_id%type,

c\_name in country.country\_name%type,

r\_id in country.region\_id%type)

is

begin

update country set country\_name=c\_name,region\_id=r\_id where country\_id=c\_id;

end;

/

**Edit\_region:**

create or replace procedure edit\_region(

r\_id in region.region\_id%type,

r\_name in region.region\_name%type)

is

begin

update region set region\_name=r\_name where region\_id=r\_id;

end;

/

**Edit\_attendance:**

create or replace procedure edit\_attendance(

atten\_id in attendance.attendance\_id%type,

atten\_emp in attendance.attendance\_employee%type,

atten\_date in attendance.attendance\_date%type)

is

begin

update attendance set attendance\_employee=atten\_emp,attendance\_date=atten\_date where attendance\_id=atten\_id;

end;

/

**Delete\_employee:**

create or replace procedure delete\_employee(e\_id in employee.emp\_id%type)

is

begin

delete from employee where emp\_id=e\_id;

dbms\_output.put\_line('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

dbms\_output.put\_line('employee ID'||e\_id||'deleted');

dbms\_output.put\_line('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

end;

/

**Delete admin:**

create or replace procedure delete\_admin(a\_id in admin.admin\_id%type)

is

begin

delete from admin where admin\_id=a\_id;

dbms\_output.put\_line('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

dbms\_output.put\_line('admin\_id'||a\_id||'deleted');

dbms\_output.put\_line('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

end;

/

**Delete\_location:**

create or replace procedure delete\_location(loc\_id in location.location\_id%type)

is

begin

delete from location where location\_id=loc\_id;

dbms\_output.put\_line('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

dbms\_output.put\_line('location\_id'||loc\_id||'deleted');

dbms\_output.put\_line('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

end;

/

**Delete\_department:**

create or replace procedure delete\_department(d\_id in department.depart\_id%type)

is

begin

delete from department where depart\_id=d\_id;

dbms\_output.put\_line('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

dbms\_output.put\_line('depart\_id'||d\_id||'deleted');

dbms\_output.put\_line('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

end;

/

**Delete\_job:**

create or replace procedure delete\_job(j\_id in jobs.job\_id%type)

is

begin

delete from jobs where job\_id=j\_id;

dbms\_output.put\_line('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

dbms\_output.put\_line('job\_id'||j\_id||'deleted');

dbms\_output.put\_line('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

end;

/

**Delete\_job\_history:**

create or replace procedure delete\_job\_history(j\_id in job\_history.job\_id%type)

is

begin

delete from job\_history where job\_id=j\_id;

dbms\_output.put\_line('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

dbms\_output.put\_line('job\_id'||j\_id||'deleted');

dbms\_output.put\_line('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

end;

/

**Delete\_training:**

create or replace procedure delete\_training(t\_id in training.training\_id%type)

is

begin

delete from training where training\_id=t\_id;

dbms\_output.put\_line('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

dbms\_output.put\_line('training\_id'||t\_id||'deleted');

dbms\_output.put\_line('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

end;

/

**Delete\_user:**

create or replace procedure delete\_user(u\_id in users.user\_id%type)

is

begin

dbms\_output.put\_line('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

dbms\_output.put\_line('user\_id'||u\_id||'deleted');

dbms\_output.put\_line('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

end;

/

**Delete\_salary:**

create or replace procedure delete\_salary(s\_id in salary.salary\_id%type)

is

begin

dbms\_output.put\_line('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

dbms\_output.put\_line('salary\_id'||s\_id||'deleted');

dbms\_output.put\_line('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

end;

/

**Delete\_country:**

create or replace procedure delete\_country(c\_id in country.country\_id%type)

is

begin

dbms\_output.put\_line('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

dbms\_output.put\_line('country\_id'||c\_id||'deleted');

dbms\_output.put\_line('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

end;

/

**Delete\_region:**

create or replace procedure delete\_region(r\_id in region.region\_id%type)

is

begin

dbms\_output.put\_line('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

dbms\_output.put\_line('region\_id'||r\_id||'deleted');

dbms\_output.put\_line('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

end;

/

**Delete\_attendance:**

create or replace procedure delete\_attendance(atten\_id in attendance.attendance\_id%type)

is

begin

dbms\_output.put\_line('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

dbms\_output.put\_line('attendance\_id'||atten\_id||'deleted');

dbms\_output.put\_line('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

end;

/

**Triggers:**

**Insert\_value\_emp:**

create or replace trigger insert\_value\_emp

after insert on employee

for each row

begin

if inserting then

dbms\_output.put\_line('RECORD INSERTED SUCCESSFULLY'||sysdate());

end if;

end;

/

**Insert\_value\_admin:**

create or replace trigger insert\_value\_admin

after insert on admin

for each row

begin

if inserting then

dbms\_output.put\_line('RECORD INSERTED SUCCESSFULLY'||sysdate());

end if;

end;

/

**Insert\_value\_department:**

create or replace trigger insert\_value\_department

after insert on department

for each row

begin

if inserting then

dbms\_output.put\_line('RECORD INSERTED SUCCESSFULLY'||sysdate());

end if;

end;

/

**Insert\_value\_location:**

create or replace trigger insert\_value\_location

after insert on location

for each row

begin

if inserting then

dbms\_output.put\_line('RECORD INSERTED SUCCESSFULLY'||sysdate());

end if;

end;

/

**Insert\_value\_job:**

create or replace trigger insert\_value\_job

after insert on jobs

for each row

begin

if inserting then

dbms\_output.put\_line('RECORD INSERTED SUCCESSFULLY'||sysdate());

end if;

end;

/

**Insert\_jobhistory:**

create or replace trigger insert\_jobhistory

after insert on job\_history

for each row

begin

if inserting then

dbms\_output.put\_line('RECORD INSERTED SUCCESSFULLY'||sysdate());

end if;

end;

/

**Insert\_training:**

create or replace trigger insert\_training

after insert on training

for each row

begin

if inserting then

dbms\_output.put\_line('RECORD INSERTED SUCCESSFULLY'||sysdate());

end if;

end;

/

**Insert\_users:**

create or replace trigger insert\_users

after insert on users

for each row

begin

if inserting then

dbms\_output.put\_line('RECORD INSERTED SUCCESSFULLY'||sysdate());

end if;

end;

/

**Insert\_salary:**

create or replace trigger insert\_salary

after insert on salary

for each row

begin

if inserting then

dbms\_output.put\_line('RECORD INSERTED SUCCESSFULLY'||sysdate());

end if;

end;

/

**Insert\_country:**

create or replace trigger insert\_country

after insert on country

for each row

begin

if inserting then

dbms\_output.put\_line('RECORD INSERTED SUCCESSFULLY'||sysdate());

end if;

end;

/

**Insert\_region:**

create or replace trigger insert\_region

after insert on region

for each row

begin

if inserting then

dbms\_output.put\_line('RECORD INSERTED SUCCESSFULLY'||sysdate());

end if;

end;

/

**Insert\_attendance:**

create or replace trigger insert\_attendance

after insert on attendance

for each row

begin

if inserting then

dbms\_output.put\_line('RECORD INSERTED SUCCESSFULLY'||sysdate());

end if;

end;

/

**Before insert, update or delete from employee:**

create or replace trigger backup

before insert or update or delete on employee

for each row

begin

insert into employee values

(:old.emp\_id,:old.emp\_name,:old.emp\_address,:old.emp\_number,:old.emp\_email);

end;

/