1. Create a table with eid, ename, salary, hire\_date and address columns.

create table emp

(eid number, ename varchar2(30), salary number(8,2), hire\_date date, address varchar2(50));

1. Set primary key on eid.

alter table emp add constraint emp \_ eid \_ pk primary key(eid)

1. Insert five record using substitution variables.

insert into emp (eid, ename, salary, hire\_date, address)

values(&eid, '&ename', &salary, '&hire\_date', '&address');

1. Add two columns named as phone, job\_title.

alter table emp add(phone varchar2(11), job\_title varchar2(30));

1. Make a query with ename, job\_title, salary and salary with 30% bonus.

select ename, job\_title, salary, salary+(salary\*.30) as review\_salary

From emp;

1. Make a query with ename, salary where salary is equal to minimum salary of employees.

select ename, salary

From emp

Where salary =(select MIN(salary) from emp);

1. Update all job\_title to manager.

update emp set job\_title='Manager'

1. Create a view.

create view v\_emp as select ename, job\_title, salary, salary+(salary\*.30) as review\_salary

from emp;

1. Create a sequence.

create sequence eid\_seq

increment by 1

start with 100 maxvalue 1000000

nocache nocycle;

1. Create an index.

create index emp\_id\_idx On emp(eid)

1. Create a trigger

create table noman

(id number,

name varchar2(30),

salary number (8,2));

-----------------------------------------

create table audit\_noman

(id number,

name varchar2(30),

old\_salary number (8,2),

new\_salary number (8,2), change\_date date);

-----------------------------------------

desc noman

desc audit\_noman

----------------------------------------

insert into noman

values(4, 'C', 5000.00);

---------------------------------------

create or replace trigger update\_audit\_noman

before update on noman

for each row

begin

insert into audit\_noman values(:old.id, :old.name, :old.salary, :new.salary, sysdate);

end;

-----------------------------------------

update noman set salary=6000 where id=12

-----------------------------------------

select \*from audit\_noman

Procedure:

create or replace procedure sqNum(x in out number) as

begin

x :=x\*x;

end;

-------------------

set serveroutput on;

declare

a number :=6;

begin

seqNum(a);

dbms\_output.put\_line(a);

end;

function with Paramiter

create or replace function findMax(x in number, y in number)

Return number is z number;

begin

if x>y then

z:=x;

else

z:=y;

end if;

return z;

end;

-----------------------------------------

set serveroutput on

begin

dbms\_output.put\_line(findMax(45,44));

end;

-----------------------------------------

function Total Employees

create or replace function totalEmployees

return number is total number (3):=0;

begin

select count (\*) into total

from employees;

return total;

end;

--------------------------------

declare

c number(3);

begin

c:= totalEmployees();

dbms\_output.put\_line('Total Employee is:'||c);

end;

created by: @ Abdullah Al Noman