```
create table emp_34 (empno number(4) not null, ename varchar2(10), job varchar2(9), mgr
number(4),hiredate date, sal number(7,2), comm number(7,2), deptno number(2) not null);

insert into emp_34 values(7839, 'king', 'president', null, '17-nov-81', 5000, null, 10);
insert into emp_34 values(7698, 'blake', 'manager', 7839, '01-may-81', 2850, null, 30);
insert into emp_34 values(7782, 'clark', 'manager', 7839, '09-jun-81', 2450, null, 20);
insert into emp_34 values(7566, 'jones', 'manager', 7839, '02-apr-81', 2975, null, 210);
insert into emp_34 values(7654, 'martin', 'salesman', 7698, '28-sep-81', 1250, 1400, 30);
insert into emp_34 values(5499, 'allen', 'salesman', 7698, '20-feb-81', 1600, 300, 30);
insert into emp_34 values(7844, 'turner', 'salesman', 7698, '08-sep-81', 1500, 0, 30);
insert into emp_34 values(7900, 'james', 'clerk', 7698, '03-dec-81', 950, null, 30);
insert into emp_34 values(7521, 'ward', 'salesman', 7698, '22-feb-81', 1250, 500, 30);
insert into emp_34 values(7902, 'ford', 'analyst', 7566, '03-dec-81', 3000, null, 20);
insert into emp_34 values(7369, 'smith', 'clerk', 7902, '17-dec-81', 800, null, 20);
insert into emp_34 values(7788, 'scott', 'analyst', 7566, '09-dec-81', 3000, null, 20);
insert into emp_34 values(7876, 'adams', 'clerk', 7788, '12-jan-83', 1100, null, 20);
insert into emp_34 values(7934, 'miller', 'clerk', 7782, '23-jan-82', 1300, null, 10);

select * from emp_34;
select ename, sal, deptno from emp_34;
select (sal+300) as new_sal from emp_34;
select * from emp_34;
select ename, ((12*sal) + 100) as annual from emp_34;
select ename || ' as a ' || job as employee_details from emp_34;
select distinct deptno from emp_34;
select ename, job, deptno from emp_34;
select ename, sal, comm from emp_34 where sal<=comm;
select ename, sal from emp_34 where sal>=1000 and sal<=2000;
select ename, sal, comm, mgr from emp_34 where mgr=7902 or mgr=7566 or mgr=7788;
select ename from emp_34 where ename like 's%';
select ename from emp_34 where ename like '%r';
```

```
create table client_master(client_no varchar2(6), name varchar2(20), address1 varchar2(30),
address2 varchar2(30), city varchar2(15), state varchar2(15), pincode number(6), bal_due
number(10,2));
create table product_master(product_no varchar2(20),description varchar2(20), profit_percent
number(20), unit_measure varchar2(20), qty_on_hand number(20), reorder_lvl number(20),
sell_price number(20), cost_price number(20));

insert into client_master values(0001, 'Ivan', null, null, 'Bombay', 'Maharashtra', 400054, 15000);
insert into client_master values(0002, 'Vandana', null, null, 'Madras', 'Tamilnadu', 780001, 0);
insert into client_master values(0003, 'Pramada', null, null, 'Bombay', 'Maharashtra', 780001, 5000);
insert into client_master values(0004, 'Basu', null, null, 'Bombay', 'Maharashtra', 780001, 0);
insert into client_master values(0005, 'Ravi', null, null, 'Delhi', null, 780001, 2000);
insert into client_master values(0006, 'Rukmini', null, null, 'Bombay', 'Maharashtra', 780001, 0);

insert into product_master values('P00001', '1.44 floppies', 5, 'piece', 100, 20, 525, 500);
insert into product_master values('P03453', 'Monitors', 6, 'piece', 10, 20, 12000, 11200);
insert into product_master values('P06734', 'Mouse', 5, 'piece', 20, 20, 1050, 500);
insert into product_master values('P07865', '1.22 floppies', 5, 'piece', 100, 20, 525, 500);
insert into product_master values('P07868', 'Keyboards', 2, 'piece', 10, 20, 3150, 3050);
insert into product_master values('P07885', 'CD Drive', 2.5, 'piece', 10, 20, 5250, 5100);
insert into product_master values('P07965', '540 HDD', 4, 'piece', 10, 20, 8400, 8000);
insert into product_master values('P07975', '1.44 Drive', 5, 'piece', 10, 20, 1050, 1000);
insert into product_master values('P08865', '1.22 Drive', 5, 'piece', 2, 20, 1050, 1000);

select name from client_master;
select name, city from client_master;
select description from product_master;
select name from client_master where city = 'Bombay';
select * from client_master where client_no=0001 or client_no=0002;
select * from product_master where description='1.44 Drive' or description='1.22 Drive';
select description from product_master where sell_price>5000;
select name from client_master where city = 'Bombay' or city = 'Delhi' or city = 'Madras';
select description from product_master where sell_price>2000 and sell_price<=5000;
select name, city, state from client_master where state!='Maharashtra';

update product_master set sell_price=1150 where description='1.44 floppies';
delete from client_master where client_no=0001;
update client_master set city='Bombay' where client_no=0005;
update client_master set bal_due=1000 where client_no=0001;
select description, (sell_price*15) as new_selling_price from product_master where sell_price>1500;
select name from client_master where city like '_a%';
select name from client_master where name like '_a%';
select description from product_master order by description;
select count(qty_on_hand) from product_master;
select avg(sell_price) from product_master;
select min(sell_price) from product_master;
select max(sell_price) as max_sell_price, min(sell_price) as min_sell_price from product_master;
select count(description) from product_master where sell_price>=1500;
```

```
create table sales_master(salesman_no varchar2(6) check (salesman_no like 's%'),
sal_name varchar2(20) not null,
address varchar2(20) not null,
city  varchar2(20),
state varchar2(20),
pincode number(6),
sal_amt number(8,2) check(sal_amt!=0) not null,
tgt_to_get number(6,2) check(tgt_to_get!=0) not null,
ytd_sales number(6,2) check(ytd_sales!=0) not null,
remarks varchar2(30),
primary key (salesman_no));

create table data(name varchar2(20) check(name='honey' or name='bunny'),
phone number(5));
insert into data values('honey', 11111);
insert into data values('som', 11111);

create table data(name varchar2(20) check(name='honey' or name='bunny'),
address varchar2(20) default 'home',
phone number(5));
insert into data (name, phone) values ('honey', 11111);
insert into data values('bunny', 'india', 11111);

create table Department( d_id varchar2(5),
d_name varchar2(20),
s_id varchar2(5),
primary key (d_id),
foreign key (s_id) references Student(s_id));

create table data(name varchar2(20) check(name='honey' or name='bunny'),
address varchar2(20) default 'home',
phone number(5),
phone2 number(5) check(phone2 >= phone));

create table Department( d_id varchar2(5),
d_name varchar2(20),
s_id varchar2(5),
primary key (d_id));
alter table Department rename to new_Department;
alter table new_Department rename column s_id to student_id;
alter table new_Department add phone number(10);
alter table new_Department modify d_name varchar2(20) not null;
alter table new_Department modify d_name varchar2(30);
drop table new_Department;

create table Department( d_id varchar2(5),
d_name varchar2(20),
s_id varchar2(5),
primary key (d_id));
alter table Department add primary key(s_id);
…….TABLE CAN HAVE ONLY ONE PRIMARY KEY……..
```

```
create table Department( d_id varchar2(5),
d_name varchar2(20),
s_id varchar2(5));
alter table Department add primary key(s_id);

create table Department( d_id varchar2(5),
d_name varchar2(20),
s_id varchar2(5));
alter table Department add constraint pk2 primary key(d_id, s_id);

create table Department( d_id varchar2(5),
d_name varchar2(20),
address varchar2(25));
alter table Department add primary key(d_id);
create table Student( s_id varchar2(5),
s_name varchar2(20));
alter table Student add foreign key (s_id) references Department(d_id);

create table Department( d_id varchar2(5),
d_name varchar2(20),
address varchar2(25));
alter table Department add constraint pk2 primary key(d_id, d_name);
create table Student( s_id varchar2(5),
s_name varchar2(20));
alter table Student add constraint fk2 foreign key (s_id, s_name) references Department(d_id,
d_name);
alter table Student drop constraint fk2;  // first foreign key has to be deleted, then primary key
alter table Department drop constraint pk2;

create table Department( d_id varchar2(5),
d_name varchar2(20),
address varchar2(25));
alter table Department add primary key(d_id);
create table Student( s_id varchar2(5),
s_name varchar2(20));
alter table Student add constraint fk1 foreign key (s_id) references Department(d_id);
alter table Student drop constraint fk1;  // to drop foreign key, constraint is always needed ig
alter table Department drop primary key;
```

join : used to combine rows from two/more tables, based on a related column between those tables

```
select * from one cross join two; // Cartesian join; all possible combinations
select * from one inner join two on one.countryId = two.countryId;
select * from one left join two on one.countryId = two.countryId;
select * from one right join two on one.countryId = two.countryId;
select * from one full outer join two on one.countryId = two.countryId;
select one.countryName, two.stateId, two.stateName from one full outer join two on one.countryId
= two.countryId;
select one.countryName, two.stateName from one, two where one.countryId = two.countryId and
one.countryName='India';  // states in India
```

pl sql : procedural sql
By default, sql is declarative. We specify ‘what-to-do’, not ‘how-to-do’
After declare, begin : no ;
|| means +
dbms_output.put_line

Sum of two numbers

```
declare
a number;
b number;
c number;
begin
a := 5;
b := 10;
c := a + b;
dbms_output.put_line('sum of '|| a || ' and ' || b || ' is = ' || c);
end;
```

Check if number is even or odd

```
declare
num number;
rem number;
begin
num := 11;
rem := mod(num,2);
if rem=0
then dbms_output.put_line('Even');
else
dbms_output.put_line('Odd');
end if;
end;
```

Sum of 1 to 10

```
declare
sumVal number;
i number;
begin
sumVal := 0;
for i in 1..10 loop
sumVal := sumVal + i;
end loop;
dbms_output.put_line('Sum is = ' || sumVal);
end;
```

Reverse a number

```
declare
a number;
d number;
rev number;
begin
a := 2345;
rev := 0;
while a>0 loop
d := mod(a, 10);
rev := (rev * 10) + d;
a := floor(a/10);
end loop;
dbms_output.put_line('Reverse is '|| rev);
end;
```

Function to add two numbers

```
create or replace function adder(n1 in number, n2 in number)
return number
is
n3 number;
begin
n3 := n1+n2;
return n3;
end;
/
declare
n3 number;
begin
n3 := adder(11,22);
dbms_output.put_line('Addition is: ' || n3);
end;
```