**Logical AND**

Select supplier\_id, supplier\_name from supplier

Where supplier\_id between

And supplier\_name like ‘%a’;

**Logical OR**

Select supplier\_id, supplier\_name from supplier

Where supplier\_id between

or supplier\_name like ‘%a’;

**Order by**

Select supplier\_id, supplier\_name,city from supplier

order by supplier\_id;

Select supplier\_id, supplier\_name,city from supplier

order by supplier\_id desc;

**Sorting**

Select order\_id, order\_bill, order\_date

From orders

Order by order\_id, order\_date desc;

**Substitution**

Select order\_id, order\_bill, order\_date

From orders

Where order\_id = &order\_num;

**Case Manipulation- Lower, Upper, Initcap**

Select lower(city) from supplier;

Select upper(supplier\_name) from supplier;

Select lower(supplier\_name), upper(city) from supplier;

Select lower(supplier\_name), initcap(Supplier\_name) from supplier;

**Substr**

Select supplier\_name, substr(supplier\_name,3) from supplier;

**Length**

Select city, length(city) from supplier;

**Instr**

Select supplier\_name, instr(supplier\_name,’a’) from supplier;

Select city, instr(city,’P’) from supplier;

**L-Pad**

Select product\_id, lpad(product\_id,3,’#’) from product;

**R-Pad**

Select product\_id, rpad(product\_id,10,’#’) from product;

**Replace**

Select replace(supplier\_name,’P’,’B’)

AND replace (supplier\_name,’i’,’a’) from supplier;

**Trim**

Select trim (‘#’ from ‘###Priyanka##’) from supplier;

**Number Funct- Round, Trunc, Mod**

Select round (order\_id/900,2) from orders;

Select trunc (order\_id/9000,1) from orders;

Select mod (order\_id,13) from orders;

**To Char**

Select order\_id, to\_char(‘order\_date’,’mm-yy’) from orders;

**Nesting**

Select (concat(Supplier\_name,city)), upper((concat(Supplier\_name,city))) from supplier;

**Average**

Select avg(order\_bill) from orders;

**Count**

Select count(order\_id) from orders;

Select count(product\_id) from orders;

**Min**

select min(order\_bill) as Min\_order from orders;

**Max**

select max(order\_bill) as Max\_order from orders;

**Sum**

select sum(order\_bill) as Total\_order from orders;

**GroupBy**

select product\_id, sum(order\_bill) from orders

group by product\_id;

**Having – we cant use where clause on GroupBy hence HAVING**

Select product\_id, max(order\_bill) from orders

Group by product\_id

Having max(order\_bill) >8000;

**Join**

Select supplier\_name, product\_id, city

From supplier

Natural join product;

**USING + Join +where**

Select p.Product\_name, product\_id, s.supplier\_name

From product p join supplier s

Using (product\_id)

Where product\_id >10;

**On Clause**

Select p.product\_id, s.supplier\_name, s.product\_id

From product p join supplier s

On(p.product\_id = s.product\_id);

**3 way join**

Select p.product\_id, s.supplier\_name,s.supplier\_id, s.product\_id, o.supplier\_id,

From product p join supplier s

On(p.product\_id = s.product\_id)

Join orders o

On(s.supplier\_id = o.supplier\_id);

**SubQuery**

Select supplier\_name

From supplier

Where supplier\_id >

( select avg(supplier\_id) from supplier);

**IN operator**

Select supplier\_id, supplier\_name,city

From supplier

Where supplier\_id in (40,50,54);

**Union**

Select product\_id

From product

Union

Select Product\_id

From supplier;

**Union All**

Select product\_id

From product

Union all

Select Product\_id

From supplier;

**Intersect**

Select product\_id

From product

Instersect

Select Product\_id

From supplier;

**Minus**

Select product\_id

From product

minus

Select Product\_id

From supplier;

Select product\_id

From supplier

minus

Select Product\_id

From product;