use cts1

select \* from employee

-- order by : used to sort records

select \* from employee order by salary

select \* from employee order by salary desc

select \* from employee order by salary

, managerid desc

create table salesdata (salesman varchar(20),

item varchar(20), region varchar(20), qty int)

insert into salesdata values

('Ajay','Mouse', 'South', 90),

('Ajay','Plotter', 'North', 12),

('Ajay','Printer', 'South', 23),

('Deepak','Mouse', 'West', 34),

('Deepak','Mouse', 'East', 83),

('Vijay','Plotter', 'South', 12),

('Ajay','Printer', 'North', 86),

('Jaideep','Mouse', 'South', 89),

('Sagar','Mouse', 'East', 23),

('Vijay','Plotter', 'East', 70)

select \* from salesdata

select count(\*) from salesdata

select sum(qty) from salesdata

-- How much sales is done by Ajay

select sum(qty) from salesdata where salesman='Ajay'

-- How much sales is done by every salesman

select qty from salesdata group by salesman

-- Group by clause : It is used to create subgroups within the table

-- data and then we can perform some functions on that data

-- Syntax of group by clause

-- select {column name} , AggregateFunction

-- from tablename group by columname

select sum(qty) from salesdata group by salesman

select salesman, sum(qty) from salesdata group by salesman

-- How much sales is done in every region

Select region , sum (qty) from salesdata group by region

-- How much sales is done by every salesman in every region

Select salesman, region , sum (qty) As "Total Sales" from salesdata group by

salesman, region

Select \* from salesdata

-- Get total sales done in every region , but sales done should

-- be more than 50

select region , sum(qty) from

salesdata

where qty > 50

group by region

-- having clause

-- Get total sales done in every region ,

-- but total sales done should be more than 150

select region , sum(qty) from

salesdata

group by region

having sum(qty)>150

select \* from salesdata

alter table salesdata add price float

update salesdata set price = 9000 where item='Printer'

update salesdata set price = 12000 where item='Plotter'

select salesman , item , qty , price, qty \* price ,

case when qty \* price > 12000 then 'You did best'

when qty \* price > 10000 then 'You did better'

when qty \* price > 12000 then 'Work hard'

end from salesdata

select qty , qty + price from salesdata

select qty , price, (qty + isnull(price,0)) from salesdata

select \* from salesdata

alter table salesdata add price float

update salesdata set price = 9000 where item='Printer'

update salesdata set price = 12000 where item='Plotter'

select salesman, qty, case

when qty > 90 then 'You did extremely well'

when qty > 70 then 'You did well'

when qty >50 then 'Work hard'

else 'Work more hard' End

AS "Performance Status"

from salesdata

select count(price) from salesdata

select count(\*) from salesdata

-- Handle NULL values

select item, qty , price , qty \* price AS "Amount" from salesdata

select item, qty , price , qty \* isnull(price,0) AS "Amount" from salesdata

<https://www.w3schools.com/sql/sql_isnull.asp>