

```

--1
select itemid, description, listprice
from pet..merchandise
where listprice > (select avg(listprice) from pet..merchandise)

--2
select a.itemid
      from (select itemid, avg((quantity*cost)/quantity) as avgcost
            from
              pet..orderitem group by itemid) a
inner join (select itemid, avg((saleprice*quantity)/quantity) as avgsale
from pet..saleitem group by itemid) s on a.itemid=s.itemid
where avgsale> 1.5*avgcost

--3
select e.employeeid, (100*sum(saletotal)/(select sum(quantity*saleprice)
from pet..saleitem)) as total
from (select saleid, employeeid from pet..sale) e
inner join (select saleid, (quantity*saleprice) as saletotal from
pet..saleitem) s on e.saleid=s.saleid
group by e.employeeid

--4
create view PercOfOrdTotal as
select ponumber, round(sum(quantity*cost),2) as 'PercOfOrdTotal'
from pet..orderitem
group by ponumber

create view avg_shipping_cost_q as
select supplierid, avg(m.shippingcost/pct.PercOfOrdTotal)*100 as [avg_ship_cost]
from pet..merchandiseorder m
      inner join PercOfOrdTotal pct on m.ponumber = pct.PercOfOrdTotal
group by supplierid

select top 1 ascq.supplierid, s.name, avg_ship_cost
from avg_shipping_cost_q ascq
inner join pet..supplier s on s.supplierid = ascq.supplierid
order by avg_ship_cost desc

--5
create view animal_money as
select customerid, sum(saleprice) as [total purchases]
from pet..sale s
inner join pet..saleanimal sa on s.saleid = sa.saleid
group by customerid

create view merchandise_money as
select customerid, sum(quantity*saleprice) as [total purchases]
from pet..sale s
inner join pet..saleitem si on s.saleid = si.saleid
group by customerid

select am.customerid, sum(am.[total purchases] + mm.[total purchases]) as [grand
total] into cust_purchases
from animal_money am

```

```
inner join merchandise_money mm on am.customerid = mm.customerid
group by am.customerid
```

```
select top 1 c.customerid, lastname, firstname, [grand total]
from temp_cust_purchases [tcp]
inner join pet..customer c on c.customerid = tcp.customerid
order by [grand total] desc
```

--6

```
create view may_purchases as
select s.customerid, sum(si.saleprice*si.quantity) as [purchases]
from pet..saleitem si
inner join pet..sale s on si.saleid = s.saleid
where month(s.saledate) = 5
group by s.customerid
```

```
create view oct_purchases as
select s.customerid, sum(si.saleprice*si.quantity) as [purchases]
from pet..saleitem si
inner join pet..sale s on si.saleid = s.saleid
where month(s.saledate) = 10
group by s.customerid
```

```
select op.customerid, c.firstname, c.lastname
from oct_purchases op
inner join may_purchases mp on op.customerid = mp.customerid
inner join pet..customer c on c.customerid = mp.customerid
where mp.purchases > 100 and
op.purchases > 50
```

--7

```
select sum(oi.quantity)-(select sum(si.quantity)
from pet..saleitem si inner join pet..sale s on s.saleid = si.saleid
where itemid in (select itemid from pet..merchandise where
description = 'dog food-can-premium') and
s.saledate >= '2004-1-1' and
s.saledate <= '2004-7-1') as [diff on-hand]
from pet..orderitem oi
inner join pet..merchandiseorder mo on mo.ponumber = oi.ponumber
where itemid in (select itemid from pet..merchandise where description = 'dog food-can-
premium') and
receivedate >= '2004-1-1' and receivedate <= '2004-7-1'
```

--8

```
select m.itemid, m.description, m.listprice
from pet..merchandise m
where m.listprice > 50
order by m.itemid
```

```
create view merchandise_items as
select m.itemid, m.description, m.listprice
from pet..merchandise m inner join pet..saleitem si on m.itemid = si.itemid
```

```

inner join pet..sale s on si.saleid = s.saleid
where m.listprice > 50 and month(s.saledate) = 7

select m.itemid, m.description, m.listprice
from pet..merchandise m
where m.listprice > 50 and not exists
(select * from merchandise_items mi where m.itemid = mi.itemid)
order by m.itemid

--9
select distinct m.itemid, m.description, m.quantityonhand, o.itemid
from PET..merchandise m left outer join PET..orderitem o on m.itemid = o.itemid
where m.QuantityOnHand > 100 and O.ItemID is null

--10
select distinct m.itemid, m.description, m.quantityonhand
from PET..merchandise m
where m.QuantityOnHand > 100 and m.itemid not in
(select distinct o.itemid from PET..merchandise m inner join PET..orderitem o on
m.itemid = o.itemid
inner join
    PET..MerchandiseOrder mo on o.ponumber = mo.ponumber
    where year(mo.orderdate) = 2004)

--11
create table category
(
category varchar(10) not null unique,
low int not null,
high int not null,
primary key (category)
);
insert into category
values
('weak', 0, 200),
('good', 200, 800),
('best', 800, 10000)

select * from category

create view as sample_output
select customerid, sum(salecost) as totalspent from
(select a.saleid as saleid, (animalsaleprice+itemsaleprice) as
salecost from
    (select saleid, sum(saleprice) as animalsaleprice
    from pet..saleanimal group by saleid) a
inner join
    (select saleid, sum(quantity*saleprice) as itemsaleprice
    from pet..saleitem group by saleid) b on a.saleid=b.saleid) c
inner join
    (select customerid, saleid from pet..sale) d on c.saleid=d.saleid
group by customerid;

--12
create view as animals_and_merchandise
select s.supplierid, s.name, 'animal' as [ordertype]
from pet..supplier s inner join pet..animalorder ao on s.supplierid = ao.supplierid

```

```
where month(ao.orderdate) = 6
union all
select s.supplierid, s.name, 'merchandise' as [ordertype]
from pet..supplier s inner join pet..merchandiseorder mo on s.supplierid = mo.supplierid
where month(mo.orderdate) = 6
```

--13

```
drop table category
```

```
create table category
```

```
(
category char(4) not null unique,
low int not null,
high int not null,
primary key (category)
);
```

```
insert into category
```

```
values ('weak', 0, 200), ('good', 200, 800), ('best', 800, 10000)
```

--14

```
insert into sample_output (customerid, lastname, firstname, grandtotal, category)
```

```
values ('1', 'Walkin', 'Walkin', '$2,261.51', 'Best')
```

--15

```
update category
```

```
set high = '400'
```

```
where category = 'weak'
```

--17

```
delete from category
```

```
where category = 'weak'
```

--18

```
select * into evan
```

```
from employee
```

```
select *
```

```
from evan
```

```
delete from evan
```

```
select * into newtable
```

```
from employee
```