**Assignment**

use classicmodels database

**1. How many products in each product line?**

select count(\*), productLine from products

group by productLine;

**2. What is the minimum payment received?**

select min(amount) from payments;

**3. List all payments greater than twice the average payment.**

select amount from payments

where amount > (select (avg(amount)\*2) from payments);

4. **What is the average percentage markup of the MSRP on buyPrice?**

select concat(round(avg( buyPrice/MSRP \* 100 ),2),'%') AS avg\_percentage from products;

**5. How many distinct products does ClassicModels sell?**

select count(distinct(productName) )from products;

**6. Report the name and city of customers who don't have sales representatives?**

select contactFirstName, contactLastName, city from customers

where salesRepEmployeeNumber is null;

**7. What are the names of executives with VP or Manager in their title? Use the CONCAT function to combine the employee's first name and last name into a single field for reporting.**

select firstName, lastName, jobTitle from employees

where jobTitle like "%Manager%" or jobTitle like "%VP%";

**8. Which orders have a value greater than $5,000?**

select orderNumber, sum(quantityOrdered\*priceEach) as value\_p from orderdetails

group by orderNumber

having sum(quantityOrdered\*priceEach) > 5000;

**9. List the value of 'On Hold' orders.**

select sum(od.quantityOrdered\*od.priceEach) as value\_p from orders o

join orderdetails od on od.orderNumber = o.orderNumber

where o.status = "On Hold";

**10. Report the number of orders 'On Hold' for each customer.**

select customerNumber, count(status) from orders

where status = "On Hold"

group by customerNumber;

**11. List products sold by order date.**

select p.productName, o.orderDate from orders o

join orderdetails od on od.orderNumber = o.orderNumber

join products p on p.productCode = od.productCode

order by o.orderDate;

**12. List the order dates in descending order for orders for the 1940 Ford Pickup Truck.**

select p.productName, o.orderDate from orders o

join orderdetails od on od.orderNumber = o.orderNumber

join products p on p.productCode = od.productCode

where p.productName = "1940 Ford Pickup Truck"

order by o.orderDate desc;

**13. List the names of customers and their corresponding order number where a particular order from that customer has a value greater than $25,000?**

select c.customerName, o.orderNumber from customers c

join orders o on o.customerNumber = c.customerNumber

join orderdetails od on od.orderNumber = o.orderNumber

group by o.orderNumber

having sum(od.quantityOrdered\*od.priceEach) > 25000;

**14. Are there any products that appear on all orders?**

select productCode, count(orderNumber) from orderDetails

group by productCode

having count(orderNumber) = (select count(distinct(orderNumber)) from orderDetails);

**15. List the names of products sold at less than 80% of the MSRP**

select productName from products where buyPrice <(0.8\*MSRP);

**16.Compute the commission for each sales representative, assuming the commission is 5% of the value of an order. Sort by employee last name and first name.**

select e.firstName, e.lastName, c.salesRepEmployeeNumber, sum(((od.quantityOrdered\*od.priceEach)\*5)/100) from employees e

join customers c on c.salesRepEmployeeNumber = e.employeeNumber

join orders o on o.customerNumber = c.customerNumber

join orderdetails od on od.orderNumber = o.orderNumber

group by c.salesRepEmployeeNumber;

**17. What is the difference in days between the most recent and oldest order date in the Orders file?**

select datediff(max(orderDate), min(orderDate)) from orders;

**18. Compute the ratio of payments for each customer for 2003 versus 2004.**

**select c.customerName, sum(p.amount) amount2003, sum(p2.amount) amount2004,**

round((sum(p.amount)/sum(p2.amount)),3) ratio03to04 from payments p

join payments p2 on p.customerNumber = p2.customerNumber

join customers c on p.customerNumber = c.customerNumber where year(p.paymentDate) = 2003 and year(p2.paymentDate) = 2004

group by customerName;

**19. Find the products sold in 2003 but not 2004.**

**20. Find the customers without payments in 2003.**

select c.customerName, p.paymentDate, amount from payments p

join customers c on c.customerNumber = p.customerNumber

where year(paymentDate)=2003 and amount is null;