

Q1-

Ans: create table customers (  
customerNumber int NOT NULL PRIMARY KEY,  
customerName varchar(200),  
contactLastName varchar(100),  
contactFirstName varchar(100),  
phone varchar(50),  
addressLine1 varchar(300)  
addressLine2 varchar(300)  
city varchar(200)  
state varchar(200)  
postalCode varchar(100)  
country varchar(200)  
saleRepEmployeeNumber int  
creditLimit float(100, 2)  
)

Q2.

Ans: create table orders (  
orderNumber int NOT NULL PRIMARY KEY,  
orderDate Date,  
requiredDate Date,  
shippedDate Date,  
status varchar(100),  
comments tinytext,  
customerNumber int  
FOREIGN KEY (customerNumber) REFERENCE customers (customerNumber)  
)

Q3.

Ans: SELECT \* FROM orders

Q4.

Ans: SELECT comments FROM orders

Q5.

Ans: SELECT orderDate, count(orderNumber) FROM orders GROUP BY orderDate

Q6.

Ans: SELECT employeeNumber, lastName, firstName FROM employees

Q7.

Ans: SELECT OD.orderNumber, CN.customerName FROM orders as OD INNER JOIN customers as CN ON OD.customerNumber = CN.customerNumber

Q8.

Ans: SELECT CN.customerName, CONCAT( EM.firstName, ' ', EM.lastName) as 'salerepemployee'  
FROM customers as CN INNER JOIN employees as EM ON CN.salesRepEmployeeNumber = EM.employeeNumber

Q9.

Ans: SELECT paymentDate, SUM(amount) FROM payments GROUP BY paymentDate

Q10.

Ans: SELECT productName, MSRP, productDescription FROM products

Q11.Ans: SELECT productName, productDescription FROM products WHERE productCode = (SELECT TOP(1) productCode FROM orderDetails ORDER BY quantityOrdered)

Q12.

Ans: SELECT city FROM customers WHERE customerNumber IN (SELECT customerNumber FROM orders HAVING COUNT(orderNumber) IN (SELECT MAX(order) FROM (SELECT COUNT(orderNumber) order FROM orders GROUP BY orderNumber) A))

Q13.

Ans: SELECT state FROM customers GROUP BY state HAVING COUNT(customerName) IN (SELECT MAX(customer) FROM (SELECT COUNT(customerName) customer FROM customers GROUP BY state) A)

Q14.

Ans: SELECT employeeNumber, CANCAT(firstName, ' ', lastName) as 'FullName' FROM employees

Q15.

Ans: SELECT cu.orderNumber, cu.CustomerNamer, SUM(od.quantityOrdered\*od.priceEach) as 'Total amount Paid' FROM customers as cu INNER JOIN orders as O ON cu.customerNumber = O.customerNumber INNER JION orderDetails as od ON od.orderNumber=od.orderNumber