

WORKSHEET 3 SQL

1. Write SQL query to create table Customers.

Ans: mysql> create table customers (

-> customerNumber int,

-> customersName varchar(20),

-> contactLastName varchar(10),

-> contactFirstName varchar(10),

-> phone int,

-> addressLine1 varchar(30),

-> addressLine2 varchar(20),

-> city varchar (10),

-> state varchar(15),

-> postalCode int,

-> country varchar(15),

-> salesRepEmployeeNumber int,

-> creditLimit int); Query OK, 0 rows affected (1.80 sec)

mysql> show tables;

+-----+

| Tables_in_sqlsheet |

+-----+

| customers |

+-----+

1 row in set (0.00 sec)

2. Write SQL query to create table Orders.

Ans: mysql> create table orders (

-> orderNumber int,

-> orderDate Date,

-> requiredDate Date,

-> shippedDate Date,

-> status varchar (15),

-> comments varchar (25),

-> customerNumber int);

Query OK, 0 rows affected (0.56 sec)

mysql> show tables;

```
+-----+
| Tables_in_sqlsheet |
+-----+
| customers | | orders |
+-----+
```

2 rows in set (0.00 sec)

3. Write SQL query to show all the columns data from the Orders Table.

Ans:

mysql> desc customers;

```
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| customerNumber | int | YES | | NULL | |
| customersName | varchar(20) | YES | | NULL | |
| contactLastName | varchar(10) | YES | | NULL | |
| contactFirstName | varchar(10) | YES | | NULL | |
| phone | int | YES | | NULL | |
| addressLine1 | varchar(30) | YES | | NULL | |
| addressLine2 | varchar(20) | YES | | NULL | |
| city | varchar(10) | YES | | NULL | |
| state | varchar(15) | YES | | NULL | |
| postalCode | int | YES | | NULL | |
| country | varchar(15) | YES | | NULL | |
| salesRepEmployeeNumber | int | YES | | NULL | |
| creditLimit | int | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
```

13 rows in set (0.05 sec)

4. Write SQL query to show all the comments from the OrdersTable.

Ans: mysql> select comments from orders;

5. Write a SQL query to show orderDate and Total number of orders placed on that date, from Orderstable.

Ans: mysql> select orderDate,sum(orderDate) from orders;

6. Write a SQL query to show employeeNumber, lastName, firstName of all the employees from employees table.

Ans: mysql> select employeeNumber, lastName, firstName from employees;

7. Write a SQL query to show all orderNumber, customerName of the person who placed the respective order.

Ans: -mysql> select orderNumber from orders UNION select customersName from customers;

8. Write a SQL query to show name of all the customers in one column and salerepemployee name in another column.

Ans: mysql> select customersName, salesRepEmployeeNumber from employees;

9. Write a SQL query to show Date in one column and total payment amount of the payments made on that date from the payments table.

Ans: mysql> select paymentDate, sum(paymentDate) from payments;

10. Write a SQL query to show all the products productName, MSRP, productDescription from the products table.

Ans: mysql> select productName, MSRP, productDescription from products;

11. Write a SQL query to print the productName, productDescription of the most ordered product.

Ans: mysql> SELECT productName, productDescription, COUNT(*) FROM products GROUP BY productName ORDER BY productName;

12. Write a SQL query to print the city name where maximum number of orders were placed.

Ans: -mysql> select city from orders inner join customer on city.customerNumber=customers.customerNumber group by city Order by count(orderNumber) desc limit3;

13. Write a SQL query to get the name of the state having maximum number of customers.

Ans: select state from customers group by state Order by Count(customerNumber) desc limit 3;

14. Write a SQL query to print the employee number in one column and Full name of the employee in the second column for all the employees.

Ans: select employeeNumber,concat(FirstName,LastName) as 'Full Name' from Employees;

15. Write a SQL query to print the orderNumber, customer Name and total amount paid by the customer for that order (quantityOrdered × priceEach).

Ans: SELECT orderNumber, customerName, SUM(priceEach * quantityOrdered) total FROM
orderDetails INNER JOIN customers USING (productCode) GROUP BY productCode ORDER BY
total;