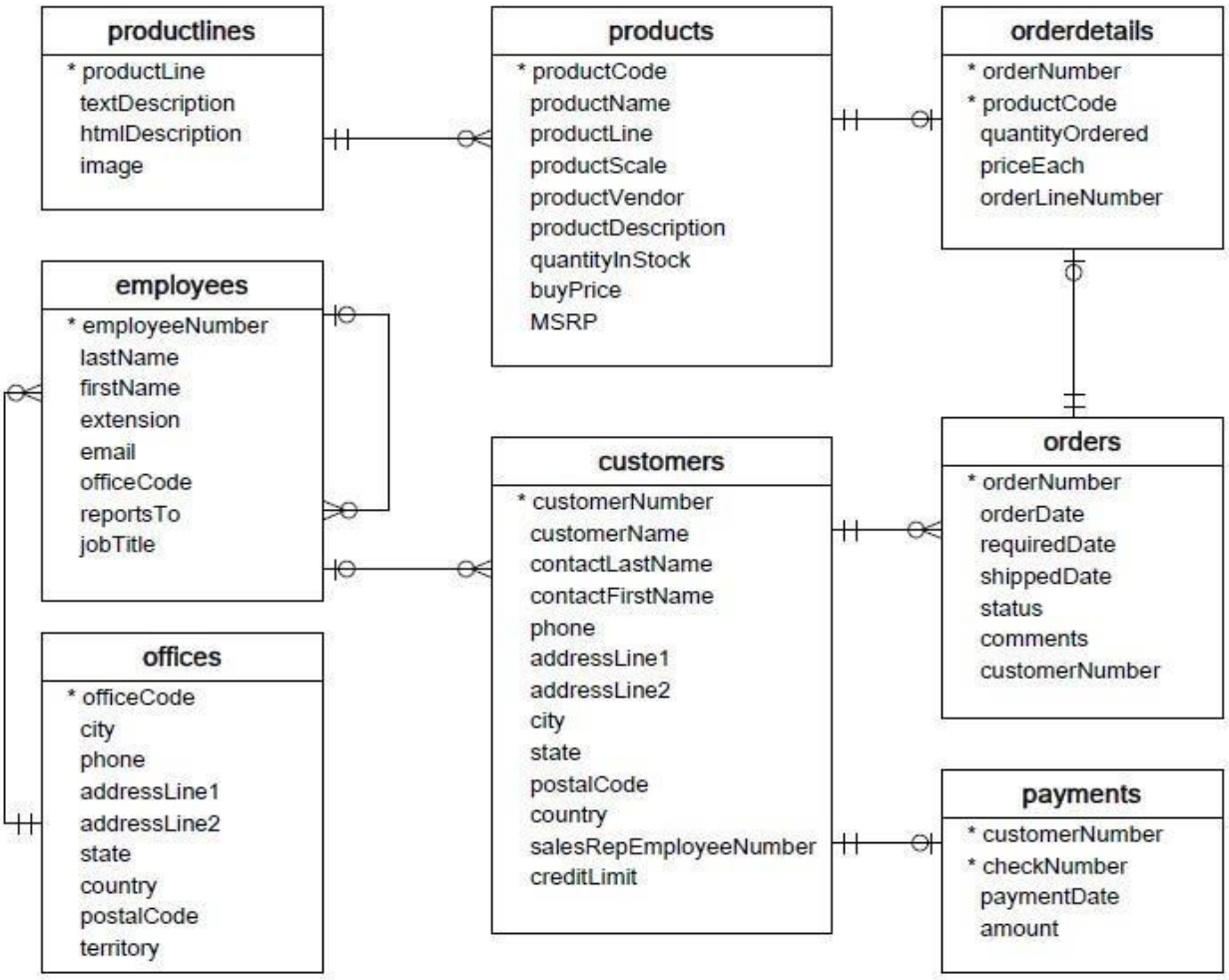
WORKSHEET 3 SQL SOLUTION



- **Customers**: stores customer's data.
- **Products**: stores a list of scale model cars.
- **ProductLines**: stores a list of product line categories.
- **Orders**: stores sales orders placed by customers.
- **OrderDetails**: stores sales order line items for each sales order.
- **Payments**: stores payments made by customers based on their accounts.
- **Employees**: stores all employee information as well as the organization structure such as who reports to whom.
- **Offices**: stores sales office data.

1. Write SQL query to create table **Customers.**
   Ans.
   The SQL CREATE TABLE statement for the customers table is: CREATE TABLE customers (customer_id int NOT NULL, customer_name char(50) NOT NULL, address char(50), city char(50), state char(25), zip_code char(10), CONSTRAINT customers_pk PRIMARY KEY (customer_id));
2. Write SQL query to create table **Orders.**
   Ans.

```
CREATE TABLE table_name (
    column1 datatype,
    column2 datatype,
    column3 datatype,
   ....
);
```

3. Write SQL query to show all the columns data from the **Orders** Table.
   Ans.
```
SELECT ord_date, salesman_id, ord_no, purch_amt
FROM orders;
```
4. Write SQL query to show all the comments from the **Orders** Table.
   Ans.
```
SELECT *
FROM orders
WHERE salesman_id IN
      (SELECT salesman_id
          FROM salesman
              WHERE city='London');
```
5. Write a SQL query to show orderDate and Total number of orders placed on that date, from **Orders** table.
   Ans.
```
SELECT COUNT(*)
FROM orders
WHERE ord_date='2012-08-17';
```
6. Write a SQL query to show employeNumber, lastName, firstName of all the employees from **employees** table.
   Ans.
```
 SELECT employee_id, first_name, last_name,
(SELECT department_name FROM departments d
 WHERE e.department_id = d.department_id) department
 FROM employees e ORDER BY department;
```

7. Write a SQL query to show all orderNumber, customerName of the person who placed the respective order.
   Ans.
```
SELECT o.orderNumber
FROM orders o
LEFT JOIN customers c
ON c.customerNumber = o.customerNumber
WHERE c.customerName LIKE 'N%'
```

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

   Ans.

```
SELECT salesman_id,name
FROM salesman a
WHERE 1 <
      (SELECT COUNT(*)
       FROM customer
            WHERE salesman_id=a.salesman_id);
```


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.

```
SELECT COUNT(*)
 FROM payment
WHERE(TO_CHAR(payment_date, 'Day')) = 'Monday'
```

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

```
SELECT item_mast.pro_name, pro_price, company_mast.com_name
FROM item_mast
INNER JOIN company_mast
ON item_mast.pro_com = company_mast.com_id;
```

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

```
SELECT p.`product_id`, p.`name`, SUM(o.`quantity`) AS quantity
FROM `Order_Detail` AS o
INNER JOIN `Product` AS p
ON o.`product_id` = p.`product_id`
GROUP BY o.`product_id`
ORDER BY SUM(o.`quantity`) DESC, p.`name` ASC
LIMIT 3
```

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

```
SELECT customer_id, COUNT(DISTINCT ord_no),
MAX(purch_amt)
FROM orders
GROUP BY customer_id
ORDER BY 2 DESC;
```

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

```
SELECT city,MAX(grade)
FROM customer
GROUP BY city;
```

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 city,MAX(grade)
FROM customer
GROUP BY city;
```

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 c.CustomerID,o.OrderID,(ord.Quantity*p.Price) as
Total_Amount
from Customers c inner join Orders o
inner join Products p
inner join OrderDetails ord
on c.CustomerID = o.CustomerID
and o.OrderID = ord.OrderID
and ord.ProductID = p.ProductID;
```

FLIP ROBO