```
/*


Neba Nfonsang
SQL Project 1: Single table queries and joins
between multiple tables
 */


USE TSQLV4
-- Download this database from
```
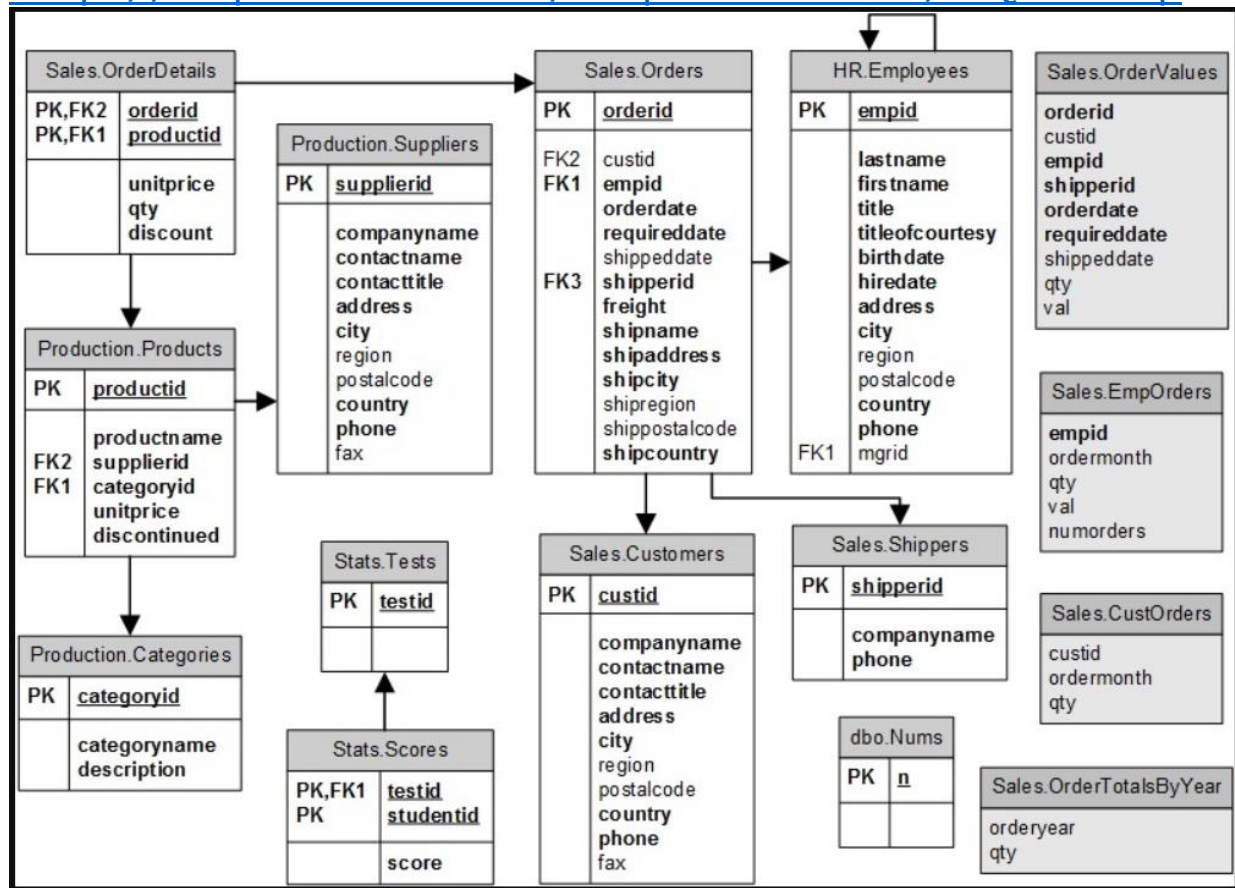http://tsql.lucient.com/SampleDatabases/TSQLV4.zip

```
/*
Query # 1: Select all records on all columns from
the HR.Employee table
*/

SELECT *
FROM HR.Employees;

/*
Query # 2: Select records from the HR.Employee table
where
the title is "Sales Manager"
*/

SELECT *
FROM HR.Employees
WHERE title='Sales Representative';

/*
Query # 3: Select records from the HR.Employee table
where employid is greater than 5
*/

SELECT *
FROM HR.Employees
WHERE empid>5;


/*
Query # 4: select records showing year and employees
where customer id=85. The records are grouped by
employee id and year
*/
```

```sql
SELECT empid,
    YEAR(orderdate) AS orderyear
FROM Sales.Orders
WHERE custid=85
GROUP BY empid, YEAR(orderdate);

/*
Query # 5: select records showing year and employees
where customer id=85.
*/

SELECT empid,
    YEAR(orderdate) AS orderyear
FROM Sales.Orders
WHERE custid=85;


/*
Query # 6: select records showing year and employees
where customer id=71. Use distinct to make sure
there are no duplicates
*/

SELECT DISTINCT empid,
    YEAR(orderdate) AS orderyear
FROM Sales.Orders
WHERE custid=71;
```

```
/*
Query # 7: select employee ids and order year
records without including duplicates (use GROUP BY)
*/

SELECT empid,
    YEAR(orderdate) AS orderyear
FROM Sales.Orders
WHERE custid=71
GROUP BY empid, YEAR(orderdate);

/*
Query # 8: count the number of orders for each
employee each year where customer id=71
*/

SELECT empid,
    YEAR(orderdate) AS orderyear,
    COUNT(*) AS numorder
FROM Sales.Orders
WHERE custid=71
GROUP BY empid, YEAR(orderdate);
```

```
/*
Query # 9: count duplicate records or number of
orders for each employee for each year where
customer with id=71 has more than 1 order.
*/

SELECT empid,
    YEAR(orderdate) AS orderyear,
    COUNT(*) AS numorder
FROM Sales.Orders
WHERE custid=71
GROUP BY empid, YEAR(orderdate)
HAVING COUNT(*)>1;

/*
Query # 10: order selected record by hiredate
*/

SELECT empid, firstname, lastname, country
FROM HR.Employees
ORDER BY hiredate;

/*
Query # 11: order selected record by hiredate
Most recent hires should be at the top of the
results (Descending order)
*/

SELECT empid, firstname, lastname, country
FROM HR.Employees
ORDER BY hiredate DESC;
```

```
/*
Query # 12: Select the top recent 15% of the hired
employees
*/

SELECT TOP (15) PERCENT empid, firstname, lastname,
country
FROM HR.Employees
ORDER BY hiredate DESC;


/*
Query # 13: If recordes have the same date,
then, use orderid to sort which record comes first
*/

SELECT TOP(10) orderid, custid, orderdate, empid
FROM Sales.Orders
ORDER BY orderdate DESC, orderid DESC;


/*
Query # 14: Skip the first 5 rows and fetch the next
4 rows
*/

SELECT orderid, custid, orderdate, empid
FROM Sales.Orders
ORDER BY orderdate DESC, orderid DESC
OFFSET 5 ROWS FETCH NEXT 4 ROWS ONLY;
```

```sql
/*
Query # 15: Include a logical condition - BETWEEN
*/

SELECT empid, lastname, firstname,
    YEAR(birthdate) as birthyear
FROM HR.Employees
WHERE YEAR(birthdate) BETWEEN 1970 AND 1980;


/*
Query # 16: Select based on a list of years of birth
*/

SELECT empid, lastname, firstname,
    YEAR(birthdate) as birthyear
FROM HR.Employees
WHERE YEAR(birthdate) IN (1970, 1972, 1980);

/*
Query # 17: Using multiple operators
*/

SELECT orderid, custid, empid, orderdate
FROM Sales.Orders
WHERE
    (custid=2
    AND empid IN(1, 2, 3, 4))
    OR
    (custid=70
    And empid in (5, 6, 7));
```

```sql
/*
Query # 18: Select full names where
last name starts with D
*/

SELECT empid, firstname + N' ' + lastname AS
fullname FROM HR.Employees
WHERE lastname LIKE N'D%';

/*
Query # 19: Select full names where
last name starts with A, B, C or D
*/

SELECT empid, firstname + N' ' + lastname AS
fullname
FROM HR.Employees
WHERE lastname LIKE N'[ABCD]%';

/*
Query # 20: left outer join (one to many)
*/

SELECT TOP(10) PERCENT p.productid, p.productname,
p.categoryid, p.unitprice,
        c.categoryname, c.description
FROM Production.Products as p
    LEFT OUTER JOIN Production.Categories as c
        ON p.categoryid=c.categoryid;
```

```
/*
Query # 21: use IS NULL
*/

SELECT TOP(10) PERCENT *
FROM Sales.Orders
WHERE shipregion is NULL
    AND (shipcountry='France')
ORDER BY orderdate;


/*
Query # 22: use IS NOT NULL
*/

SELECT TOP(1) PERCENT *
FROM Sales.Orders
WHERE shipregion is NOT NULL
ORDER BY orderdate;
```