Sample SQL Queries

1) Retrieve the address and last order date of the customer named Karen Smith.

```
SELECT      ADDRESS, LAST_ORDER_DATE
FROM        CUSTOMER
WHERE       NAME = 'KAREN SMITH'
```

2) Retrieve the name and price of all toys made by Fischer-Price.

```
SELECT      NAME, MSRP
FROM        TOY, MANUFACTURER
WHERE       MAN_NAME = 'FISCHER PRICE'  AND
            TOY.MAN_ID=MANUFACTURER.MAN_ID
```

3) For every undelivered order, list the toy name, the manufacturer name and the customer name

```
SELECT      TOY.NAME, MAN_NAME, CUSTOMER.NAME
FROM        ORDER, TOY, MANUFACTURER, CUSTOMER
WHERE       DELIV IS NULL AND
            ORDER.CUST_NUM = CUSTOMER.CUST_NUM AND
             ORDER.TOY_NUM = TOY.TOY_NUM AND
            TOY.MAN_ID = MANUFACTURER.MAN_ID
```

4) Retrieve the name of every toy in the toy relation.

```
SELECT      NAME
FROM        TOY
```

5) Retrieve the name of every toy and the name of every manufacturer.

```
SELECT      NAME, MAN_NAME
FROM        TOY, MANUFACTURER
```

6) Retrieve all attributes of the TOY relation for which the manufacturer is FP

```
SELECT      *
FROM        TOY
WHERE       MAN_ID = 'FP'
```

7) List the prices of all toys in the TOY relation.

```
SELECT      MSRP
FROM        TOY
```

```
SELECT      DISTINCT MSRP
FROM        TOY
```

8) (Using the Company db from the textbook - because our example does not have recursion)

Make a list of all project names for projects that involve an employee whose last name is SMITH as a worker or as a manager of the dept that controls the project.

```
(SELECT     PNAME
FROM        EMPLOYEE, WORKS_ON, PROJECT
WHERE       LNAME=SMITH AND SSN=ESSN AND
             PNO=PNUMBER)
UNION
( SELECT    PNAME
FROM        EMPLOYEE, DEPARTMENT, PROJECT
WHERE       LNAME=SMITH AND SSN=MGRSSN AND
            DNUMBER=DNUM)
```

9) Reformulate the above query as a nested query

```
SELECT      DISTINCT PNUMBER
FROM        PROJECT
WHERE       PNUMBER IN      (SELECT     PNUMBER
                             FROM       PROJECT, DEPARTMENT,
                                        EMPLOYEE
                             WHERE      DNUM=DNUMBER AND
                                        MGRSSN=SSN AND
                                        LNAME=SMITH)
            OR
            PNUMBER IN      (SELECT     PNO
                             FROM       WORKS_ON, EMPLOYEE
                             WHERE      ESSN=SSN AND
                                        LNAME=SMITH)
```

10) Select the toy numbers of all toys that have the same price and age group as the Farm House.

```
SELECT      DISTINCT TOY_NUM
FROM        TOY
WHERE       (MSRP, AGE_GROUP) IN    (SELECT MSRP, AGE_GROUP
                                     FROM    TOY
                                     WHERE   NAME = FARM HOUSE)
```

11) Select the toy names of all toys that cost more than the Farm House.

```
SELECT      NAME
FROM        TOY
```

```
WHERE     MSRP > ALL (SELECT    MSRP
                      FROM     TOY
                      WHERE    NAME=FARM HOUSE)
```

12) (from Company db) Retrieve the name of each employee who has a dependent with the same first name and sex as the employee.

```
SELECT    E.FNAME, E.LNAME
FROM      EMPLOYEE E
WHERE     E.SSN IN        (SELECT    ESSN
                           FROM       DEPENDENT
                           WHERE      ESSN=E.SSN AND
                           E.FNAME=DEPENDENT_NAME
                                      AND SEX=E.SEX)
```

13) Query 12 can be rewritten using the EXISTS clause

```
SELECT    E.FNAME, E.LNAME
FROM      EMPLOYEE E
WHERE     EXISTS    (SELECT     *
                     FROM        DEPENDENT
                     WHERE       E.SSN=ESSN AND SEX=E.SEX
                                 AND
                                 E.FNAME=DEPENDENT_NAME)
```

14) List the names of customers who have no outstanding orders.

```
SELECT    NAME
FROM      CUSTOMER C
WHERE     NOT EXISTS      (SELECT    *
                           FROM       ORDER
                           WHERE
                           C.CUST_NUM=ORDER.CUST_NUM
                                 AND DELIV IS NULL)
```

15) Retrieve the names of all toys manufactured by FP or FY.

```
SELECT    TOY_NAME
FROM      TOY
WHERE     MAN_ID IN (FP, FY)
```

- can explicitly specify a set of values using an IN clause.

16) Retrieve the names of customers who have never ordered a toy from the catalog.

```
SELECT    NAME
FROM      CUSTOMER
WHERE     LAST_ORDER_DATE IS NULL
```

17) Retrieve the toy names and the customer names for every outstanding order for toys whose names fall in the first half of the alphabet .

```
SELECT    T.NAME AS TOY_NAME, C.NAME AS CUSTOMER_NAME
FROM      CUSTOMER AS C, TOY AS T, ORDER AS O
WHERE     (C.CUST_NUM=O.CUST_NUM) AND (DELIV IS NULL)
           AND (T.TOY_NUM=ORDER.TOY_NUM) AND
           (TOY_NAME < 'N')
```

18) Retrieve the toy number of every toy ordered by KAREN SMITH.

```
SELECT    TOY_NUM
FROM      (ORDER JOIN CUSTOMER ON
                    ORDER.CUST_NUM=CUSTOMER.CUST_NUM)
WHERE     NAME='KAREN SMITH'
```

19) Find the average price of all toys in the TOY relation.

```
SELECT    AVG(MSRP)
FROM      TOY
```

- built-in aggregate functions SUM, MAX, MIN, AVG, COUNT

20) Find the total number of toys orderd by and the total amount of money spent by customer GEORGE GRANT.

```
SELECT    SUM(MSRP*QUANTITY), SUM(QUANTITY)
FROM      TOY AS T, CUSTOMER AS C, ORDER AS O
WHERE     O.TOY_NUM=T.TOY_NUM AND
           O.CUST_NUM=C.CUST_NUM AND
          C.NAME='GEORGE GRANT'
```

21) Find the total number of toys order by and the total amount of money spent by each customer.

```
SELECT    CUST_NUM, SUM(MSRP*QUANTITY), SUM(QUANTITY)
FROM      TOY AS T, ORDER AS O
WHERE     O.TOY_NUM=T.TOY_NUM
GROUP BY  CUST_NUM
```

22) Find the total number of toys order by and the total amount of money spent by each customer who made at least three orders.

```
SELECT    CUST_NUM, SUM(MSRP*QUANTITY), SUM(QUANTITY)
FROM      TOY AS T, ORDER AS O
WHERE     O.TOY_NUM=T.TOY_NUM
GROUP BY  CUST_NUM
HAVING    COUNT(*) > 3
```

- HAVING clause allows you to put a condition on the groups that end up in the result

23) Retrieve all customers who live in New York state.

```
SELECT      NAME
FROM        CUSTOMER
WHERE       ADDRESS LIKE '%NY%
```

24) Show the new prices if Fischer Price raised their MSRPs by 10%.

```
SELECT      NAME, 1.1*MSRP
FROM        TOY
WHERE       MAN_ID=FP
```

25) Retrieve all toys with fewer than 50 in inventory sorted by manufacturer and by price within each manufacturer.

```
SELECT      MAN_ID, NAME, MSRP
FROM        TOY
WHERE       NUM_IN_STOCK<50
ORDER BY    MAN_ID, MSRP
```