```sql
-- 103
SELECT MOVIE_TITLE, MOVIE_YEAR, MOVIE_GENRE
FROM   MOVIE

-- 104
SELECT        MOVIE_YEAR, MOVIE_TITLE, MOVIE_COST
FROM          MOVIE
ORDER BY      MOVIE_YEAR DESC

-- 105
SELECT        MOVIE_TITLE, MOVIE_YEAR, MOVIE_GENRE
FROM          MOVIE
ORDER BY      MOVIE_GENRE ASC, MOVIE_YEAR DESC

-- 106
SELECT MOVIE_NUM, MOVIE_TITLE, PRICE_CODE
FROM   MOVIE
WHERE  MOVIE_TITLE LIKE 'R%'

-- 107
SELECT MOVIE_TITLE, MOVIE_YEAR, MOVIE_COST
FROM   MOVIE
WHERE  MOVIE_TITLE LIKE '%HOPE%'

-- 108
SELECT MOVIE_TITLE, MOVIE_YEAR, MOVIE_GENRE
FROM   MOVIE
WHERE  MOVIE_GENRE = 'ACTION'

-- 109
SELECT MOVIE_NUM, MOVIE_TITLE, MOVIE_COST
FROM   MOVIE
WHERE  MOVIE_COST > 40

-- 110
SELECT        MOVIE_NUM, MOVIE_TITLE, MOVIE_COST, MOVIE_GENRE
FROM          MOVIE
WHERE         MOVIE_GENRE IN ('ACTION','COMEDY') AND MOVIE_COST < 50
ORDER BY      MOVIE_GENRE

-- 111
SELECT MEM_NUM, MEM_FNAME, MEM_LNAME, MEM_STREET, MEM_STATE, MEM_BALANCE
FROM   MEMBERSHIP
WHERE  MEM_STATE = 'TN' AND MEM_BALANCE < 5 AND MEM_STREET LIKE '%AVENUE'

-- 112
SELECT        MOVIE_GENRE, COUNT(MOVIE_GENRE) AS "NUMBER OF MOVIES"
FROM          MOVIE
GROUP BY      MOVIE_GENRE

-- 113
SELECT AVG(MOVIE_COST) AS "AVERAGE MOVIE COST"
FROM   MOVIE

-- 114
SELECT        MOVIE_GENRE, AVG(MOVIE_COST) AS "AVERAGE COST"
FROM          MOVIE
GROUP BY      MOVIE_GENRE
```

```sql
-- 115
SELECT MOVIE_TITLE, MOVIE_GENRE, PRICE_DESCRIPTION, PRICE_RENTFEE
FROM   MOVIE M INNER JOIN PRICE P ON M.PRICE_CODE = P.PRICE_CODE

-- 116
SELECT MOVIE_GENRE, AVG(PRICE_RENTFEE) AS "AVERAGE RENTAL FEE"
FROM   MOVIE M INNER JOIN PRICE P ON M.PRICE_CODE = P.PRICE_CODE

-- 117
SELECT MOVIE_TITLE, (MOVIE_COST / PRICE_RENTFEE) AS "BREAKEVEN RENTALS"
FROM   MOVIE M INNER JOIN PRICE P ON M.PRICE_CODE = P.PRICE_CODE

-- 118
SELECT MOVIE_TITLE, MOVIE_YEAR
FROM   MOVIE
WHERE  PRICE_CODE IS NOT NULL

-- 119
SELECT MOVIE_TITLE, MOVIE_GENRE, MOVIE_COST
FROM   MOVIE
WHERE  MOVIE_COST >= 44.99 AND MOVIE_COST <= 49.99

-- 120
SELECT MOVIE_TITLE, PRICE_DESCRIPTION, PRICE_RENTFEE, MOVIE_GENRE
FROM   MOVIE M INNER JOIN PRICE P ON M.PRICE_CODE = P.PRICE_CODE
WHERE  MOVIE_GENRE = 'FAMILY' OR MOVIE_GENRE = 'COMEDY' OR MOVIE_GENRE = 'DRAMA'

-- 121
SELECT M.MEM_NUM, MEM_FNAME, MEM_LNAME, MEM_BALANCE
FROM   MEMBERSHIP M INNER JOIN RENTAL R ON M.MEM_NUM = R.MEM_NUM
WHERE  R.MEM_NUM IS NOT NULL

-- 122
SELECT MIN(MEM_BALANCE) AS "MINIMUM BALANCE", MAX(MEM_BALANCE) AS "MAXIMUM BALANCE",
            AVG(MEM_BALANCE) AS "AVERAGE BALANCE"
FROM   MEMBERSHIP M INNER JOIN RENTAL R ON M.MEM_NUM = R.MEM_NUM
WHERE  R.MEM_NUM IS NOT NULL

-- 123
SELECT        R.RENT_NUM, R.RENT_DATE, V.VID_NUM, M.MOVIE_TITLE, D.DETAIL_DUEDATE,
D.DETAIL_RETURNDATE
FROM          RENTAL R INNER JOIN DETAILRENTAL D ON R.RENT_NUM = D.RENT_NUM
                     INNER JOIN VIDEO V ON D.VID_NUM = V.VID_NUM
                     INNER JOIN MOVIE M ON V.MOVIE_NUM = M.MOVIE_NUM
WHERE         D.DETAIL_RETURNDATE > D.DETAIL_DUEDATE
ORDER BY      R.RENT_NUM, M.MOVIE_TITLE

-- 124
SELECT        R.RENT_NUM, R.RENT_DATE, V.VID_NUM, M.MOVIE_TITLE, D.DETAIL_FEE
FROM          RENTAL R INNER JOIN DETAILRENTAL D ON R.RENT_NUM = D.RENT_NUM
                     INNER JOIN VIDEO V ON D.VID_NUM = V.VID_NUM
                     INNER JOIN MOVIE M ON V.MOVIE_NUM = M.MOVIE_NUM
WHERE         D.DETAIL_RETURNDATE <= D.DETAIL_DUEDATE

-- 125
CREATE VIEW TEMP_VIEW AS
SELECT MOVIE_GENRE,
```

```
            (SELECT       AVG(MOVIE_COST)
            FROM   MOVIE
            WHERE  M.MOVIE_GENRE = MOVIE_GENRE
            ) AS "AVERAGE COST"
FROM   MOVIE M

SELECT M.MOVIE_NUM, M.MOVIE_GENRE, T.[AVERAGE COST], M.MOVIE_COST, (M.MOVIE_COST -
T.[AVERAGE COST])
            / T.[AVERAGE COST] * 100 AS "PERCENT DIFFERENCE"
FROM   MOVIE M INNER JOIN TEMP_VIEW T ON M.MOVIE_GENRE = T.MOVIE_GENRE

DROP VIEW TEMP_VIEW
```