CSC343 Worksheet 6 Solution

June 21, 2020

1. **Exercise 8.1.1:**

a)
```
1   CREATE VIEW RichExec AS
2       SELECT * FROM MovieExec
3       WHERE netWorth >= 10000000;
4
```

Notes:

- Virtual Views
  - **Syntax:** CREATE VIEW < view-name > AS < view-definition >
  - Contrasts to database that exists in physical storage
  - Exists in RAM
  - Is created using query
  - can be used like a relation

    ### Notes:

    ```
    1       CREATE VIEW ParamountMovies AS
    2           SELECT title, year
    3           FROM Movies
    4           WHERE studioName = 'Paramount';
    5
    ```

b)
```
1   CREATE VIEW StudioPres AS
2       SELECT * FROM Movies
3       INNER JOIN Studio ON cert# = presC#;
4
```

c)
```
1   CREATE VIEW ExecutiveStar AS
2       SELECT * FROM MovieExec
3       NATURAL JOIN MovieStar;
4
```

2. **Exericse 8.1.2:**

a)
```
SELECT name, gender FROM ExecutiveStar;
```

b)
```
SELECT name FROM RichExec WHERE netWorth > 10000000;
```

c)
```
SELECT name FROM StudioPres
NATURAL JOIN ExecutiveStar
WHERE netWorth > 50000000
```

3. **Exericse 8.2.1:**

RichExec is updatable.

Notes:

- Updatable View Conditions
    - The WHERE cluase in CREATE VIEW must not be a subquery
    - The FROM clause has only one occurence of R
    - The SELECT clause must include enough attributes
    - NOT NULL attributes must have default values
        * A solution to this is by including the attribute without default value in CREATE VIEW

    Example:

    ```
    Movies(title, year, length, genre, studioName, producerC#)
    Suppose studioName is NOT NULL but has no default value.
    Then, a fix is:

    CREATE VIEW Paramount AS
        SELECT studioName, title, year
        FROM Movies
        WHERE studioName = 'Paramount';
    ```

4. **Exericse 8.2.2:**

a) No. It is not updatable. Since,

1. studioName attribute in Movies is NOT NULL without default value

b) **Notes:**

- Using Trigger in VIEW
  - Uses INSTEAD OF in place of BEFORE or AFTER
  - When event causes the trigger, the trigger is done instead of the event

    Example:

```sql
CREATE VIEW ParamountMovies AS
    SELECT title, year
    FROM Movies
    WHERE studioName = 'paramount';

CREATE TRIGGER ParamountInsert
INSTEAD OF INSERT ON ParamountMovies
REFERENCING NEW ROW AS NewRow
FOR EACH ROW
INSERT INTO Movies(title, year, studioName)
VALUES(NewRow.title, NewRow.year, 'Paramount');
```