Appendix

All Queries in Web:

1. Recommend TV shows based on ratings, director name, TvShow Year, or actor name based on user selection Recommend TV shows based on selected show's cast members AND genre

```
WITH Cast AS (
    SELECT CI.tid
    FROM Cast_In CI
        JOIN IMDBPerson P
        ON CI.pid = P.pid
    WHERE P.name LIKE '%${actor}%'
), Genre AS (
    SELECT tid
    FROM Genres
    WHERE Genre LIKE '%${genre}%'
)
SELECT pTitle as TV_Show
FROM Titles
WHERE tid IN
    (SELECT tid FROM Cast)
    AND
    tid IN
    (SELECT tid FROM Genre)
ORDER BY TV_Show ASC
```

Display long running TV shows within specified rating range

```
SELECT t.pTitle AS TV_Show,
    (t.endYear - t.startYear)
    AS runTime
FROM Titles t JOIN Ratings r
    ON t.tid = r.tid
WHERE r.aveRating >=
    ${minInputRating}
    AND r.aveRating <=
    ${maxInputRating}
ORDER BY runTime DESC;
```

Display all directors' work who share the user's birth year

```
WITH TVDirectors AS (
    SELECT t.pTitle, d.pid
    FROM Directors d
        JOIN Titles t ON d.tid = t.tid
    WHERE t.type LIKE '%tvseries%'
)
SELECT td.pTitle AS TV_Show
FROM TVDirectors td
    JOIN IMDBPerson ip
    ON td.pid = ip.pid
WHERE ip.birthYear = ${user_birthyear};
```

Recommend TV shows based on ratings, director name, TvShow Year, or actor name based on user selection

```
SELECT DISTINCT t.tid AS tid,
    pTitle AS TV_Show,
    aveRating AS Rating,
    startYear, endYear,
    "Add to WatchList" AS Watch
FROM Titles t
    JOIN Ratings r
    ON t.tid = r.tid
WHERE r.aveRating >= ${minRating}
    AND r.aveRating <= ${maxRating}
    AND t.startYear >= ${startYear}
    AND t.endYear <= ${endYear}
ORDER BY t.startYear, aveRating DESC;

WITH DirectorTitle AS (
    SELECT DISTINCT d.tid,
        ip.name AS Director
    FROM Directors d
        JOIN IMDBPerson ip
        ON d.pid = ip.pid
    WHERE ip.name LIKE
        "%${director_name}%"
),
    DirectorTVShows AS (
        SELECT d.tid,
            t.pTitle AS TV_Show,
            t.startYear,
            t.endYear, d.Director
        FROM DirectorTitle d
            JOIN Titles t
            ON d.tid = t.tid
    )
    SELECT dtv.tid AS tid,
        TV_Show, Director,
        aveRating AS Rating,
        startYear, endYear,
        "Add to WatchList" AS Watch
    FROM DirectorTVShows dtv
        JOIN Ratings r
        ON dtv.tid = r.tid
    WHERE r.aveRating >= ${minRating}
        AND r.aveRating <= ${maxRating}
        AND dtv.startYear >= ${startYear}
        AND dtv.endYear <= ${endYear}
    ORDER BY dtv.startYear,
        aveRating DESC,
        dtv.Director;

WITH ActorTitle AS (
    SELECT DISTINCT ci.tid,
        ip.name AS Actor
    FROM Cast_In ci
```

```
        JOIN IMDBPerson ip
        ON ci.pid = ip.pid
    WHERE ip.name LIKE "%${actor_name}%"
),
    ActorTVShows AS (
        SELECT a.tid,
            t.pTitle AS TV_Show,
            t.startYear,
            t.endYear, a.Actor
        FROM ActorTitle a
            JOIN Titles t
            ON a.tid = t.tid
    )
    SELECT atv.tid AS tid,
        TV_Show, Actor,
        aveRating AS Rating,
        startYear, endYear,
        "Add to WatchList" AS Watch
    FROM ActorTVShows atv
        JOIN Ratings r
        ON atv.tid = r.tid
    WHERE r.aveRating >= ${minRating}
        AND r.aveRating <= ${maxRating}
        AND atv.startYear >= ${startYear}
        AND atv.endYear <= ${endYear}
    ORDER BY atv.startYear,
        aveRating DESC, atv.Actor;

WITH DirectorTitle AS (
    SELECT DISTINCT d.tid,
        ip.name AS Director
    FROM Directors d
        JOIN IMDBPerson ip
        ON d.pid = ip.pid
    WHERE ip.name LIKE
        "%${director_name}%"
),
    ActorTitle AS (
        SELECT DISTINCT d.tid,
            d.Director,
            ip.name AS Actor
        FROM DirectorTitle d
            JOIN Cast_In ci
            ON d.tid = ci.tid
            JOIN IMDBPerson ip
            ON ci.pid = ip.pid
        WHERE ip.name LIKE
            "%${actor_name}%"
    ),
        DirectorActorTVShows AS (
            SELECT a.tid,
                t.pTitle AS TV_Show,
                Director, Actor,
                t.startYear, t.endYear
            FROM ActorTitle a
                JOIN Titles t
                ON a.tid = t.tid
        )

SELECT datv.tid AS tid,
    TV_Show, Director,
    Actor, aveRating AS Rating,
    startYear, endYear,
    "Add to WatchList" AS Watch
FROM DirectorActorTVShows datv
    JOIN Ratings r
    ON datv.tid = r.tid
WHERE r.aveRating >= ${minRating}
    AND r.aveRating <= ${maxRating}
    AND datv.startYear >= ${startYear}
    AND datv.endYear <= ${endYear}
ORDER BY datv.startYear,
    aveRating DESC,
    datv.Director, datv.Actor;
```

2. Show based on range of runtimeMinutes; sorted

```
SELECT DISTINCT t.tid,
    t.pTitle as TV_Show,
    t.runtimeMinutes AS Runtime,
    "Add to WatchList" AS Watch
FROM Titles t JOIN Ratings r
    ON t.tid = r.tid
WHERE t.runtimeMinutes
    >=${minRangeRunTime}
    AND t.runtimeMinutes
    <= ${maxRangeRunTime}
ORDER BY t.runtimeMinutes DESC;
```

3. Display long running TV shows within specified rating range

```
SELECT DISTINCT t.tid,
    t.pTitle AS TV_Show,
    (MAX(t.endYear)-MIN(t.startYear))
    AS Years,
    "Add to WatchList" AS Watch
FROM Titles t JOIN Ratings r
    ON t.tid = r.tid
GROUP BY t.tid
HAVING Years>=${minYears}
    AND Years<=${maxYears}
ORDER BY Years DESC;
```

4. Display all directors' work who share the user's birth year

```
SELECT DISTINCT  t.tid,
    t.pTitle AS TV_Show,
    IP.name as Director,
```

```
    "Add to WatchList" AS Watch
FROM Titles t JOIN Directors d
    ON d.tid = t.tid
    JOIN IMDBPerson IP
    on IP.pid = d.pid
    JOIN AdditionalTtles A
    ON t.tid = A.tid
WHERE t.type LIKE '%tvseries%'
    AND IP.birthYear=${user_birthyear}
    AND A.language="en";

SELECT DISTINCT  t.tid as tid,
    t.pTitle AS TV_Show,
    IP.name as Director,
    "Add to WatchList" AS Watch
FROM Titles t
    JOIN Directors d
    ON d.tid = t.tid
    JOIN IMDBPerson IP
    on IP.pid = d.pid
WHERE t.type LIKE '%tvseries%'
    AND IP.birthYear=${user_birthyear};
```

5. Display all writers' work who share the user's birth year

```
SELECT DISTINCT  t.tid as tid,
    t.pTitle AS TV_Show,
    IP.name as Writer,
    "Add to WatchList" AS Watch
FROM Titles t
    JOIN Writers w
    ON w.tid = t.tid
    JOIN IMDBPerson IP
    on IP.pid = w.pid
    JOIN AdditionalTitles A
    on t.tid = A.tid
WHERE t.type LIKE '%tvseries%'
    AND IP.birthYear=${user_birthyear}
    AND A.language="en";

SELECT DISTINCT  t.tid as tid,
    t.pTitle AS TV_Show,
    IP.name as Writer,
    "Add to WatchList" AS Watch
FROM Titles t
    JOIN Writers w
    ON w.tid = t.tid
    JOIN IMDBPerson IP
    on IP.pid = w.pid
WHERE t.type LIKE '%tvseries%'
    AND IP.birthYear=${user_birthyear};
```

6. Display all actors' shows who share the user's birth year

```
WITH MIN AS (
    SELECT DISTINCT  t.tid
    FROM Titles t
    WHERE t.type LIKE '%tvseries%'
        AND t.startYear<=${birthYear}
)
    SELECT M.tid as tid,
        t.pTitle AS TV_Show,
        t.startYear, t.endYear,
        "Add to WatchList" AS Watch
    FROM Titles t
        JOIN MIN M
        ON t.tid=M.tid
        JOIN AdditionalTitles A
        on t.tid = A.tid
    WHERE t.type LIKE '%tvseries%'
        AND t.endYear>=${birthYear}
        AND A.language="en";

WITH MIN AS (
    SELECT DISTINCT  t.tid
    FROM Titles t
    WHERE t.type LIKE '%tvseries%'
        AND t.startYear<=${birthYear}
)
    SELECT M.tid as tid,
        t.pTitle AS TV_Show,
        t.startYear, t.endYear,
        "Add to WatchList" AS Watch
    FROM Titles t
        JOIN MIN M
        ON t.tid=M.tid
    WHERE t.type LIKE '%tvseries%'
        AND t.endYear>=${birthYear};
```

7. Display all shows within the range of the user's birth year

```
SELECT DISTINCT  t.tid,
    t.pTitle AS TV_Show,
    IP.name as Actor,
    "Add to WatchList" AS Watch
FROM Titles t JOIN Cast_In CI
    ON CI.tid = t.tid
    JOIN IMDBPerson IP
    ON IP.pid = CI.pid
    JOIN AdditionalTitles A
    on t.tid = A.tid
WHERE t.type LIKE '%tvseries%'
    AND IP.birthYear=${user_birthyear}
    AND A.language="en";

SELECT DISTINCT  t.tid as tid,
    t.pTitle AS TV_Show,
    IP.name as Actor,
```

```
    "Add to WatchList" AS Watch
FROM Titles t
    JOIN Cast_In CI
    ON CI.tid = t.tid
    JOIN IMDBPerson IP
    on IP.pid = CI.pid
WHERE t.type LIKE '%tvseries%'
    AND IP.birthYear=${user_birthyear};
```

8. Display TV shows and writers of a particular genre

```
WITH TVTitles AS (
    SELECT tid, pTitle
    FROM Titles
    WHERE type LIKE '%tvseries%'
), GenresList AS (
    SELECT T.tid
    FROM Titles T JOIN Genres G
        ON T.tid = G.tid
    WHERE type LIKE '%tvseries%'
        AND G.Genre LIKE '%${genre}%'
), WritersList AS (
    SELECT w.pid,  t.tid, t.pTitle
    FROM TVTitles t JOIN Writers w
        ON t.tid = w.tid
    WHERE t.tid IN
        (SELECT tid FROM GenresList)
    GROUP BY w.pid
)
SELECT DISTINCT WL.tid,
    WL.pTitle as TV_Show,
    IP.name as Writer
FROM WritersList WL JOIN IMDBPerson IP
    ON WL.pid = IP.pid
ORDER BY TV_Show;
```

9. Display TV shows that have cast sizes of at least minSize

```
WITH SizeCount AS (
    SELECT tid , COUNT(*) AS size
    FROM Cast_In
    GROUP BY tid
)
SELECT T.tid, T.pTitle as TV_Show,
    SC.size
FROM SizeCount SC JOIN
    Titles T ON SC.tid = T.tid
WHERE SC.size >= ${minSize}
    AND SC.size <=${maxSize}
ORDER BY SC.size DESC;
```

10. Get tv show directed by a director involved in at least x-number (user input) of different TV shows within a specified rating range

```
WITH DirectorList AS (
    SELECT pid, COUNT(DISTINCT tid)
    AS numShows
    FROM Directors
    GROUP BY pid
    HAVING numShows >= ${minSize}
  ), DirectorsName AS (
      SELECT IP.name, DL.numShows, D.tid
      FROM IMDBPerson IP JOIN
        DirectorList DL ON IP.pid = DL.pid
        JOIN Directors D on IP.pid = D.pid
  )
    SELECT DISTINCT T.tid,
        DN.name AS Director,
        T.pTitle as TV_Show, DN.numShows,
        "Add to WatchList" AS Watch
    FROM Titles T JOIN DirectorsName DN
        ON T.tid=DN.tid
    WHERE DN.name LIKE "%${director}%"
    ORDER BY numShows DESC , name ASC;
```

11. Display all TV shows

```
SELECT DISTINCT T.tid as tid,
    T.pTitle as TV_Show,
    "Add to WatchList" AS Watch
FROM Titles T
ORDER BY TV_Show ASC;
```

12. Find all highest rated TV shows

```
SELECT DISTINCT T.tid, T.pTitle as TV_Show,
    R.aveRating as Rating,
    "Add to WatchList" AS Watch
FROM Titles T join Ratings R
    on T.tid = R.tid
WHERE type LIKE '%tvseries%'
    AND R.aveRating>=${minRating}
    AND R.aveRating <=${maxRating}
ORDER BY Rating DESC;
```

13. Populate Watch List

```
SELECT DISTINCT T.tid as tid,
    pTitle AS TV_Show,
    "Remove from WatchList" AS Remove
FROM Titles T
    JOIN WatchList l
    ON T.tid = l.tid
ORDER BY pTitle;
```

14. Add to Watch List

```
INSERT INTO WatchList(tid)
VALUES ("${req.query.tid}");
```

15. Remove show from Watch List

```
DELETE FROM WatchList
WHERE tid = "${req.query.tid}";
```

16. Clear Watch List

```
DELETE FROM WatchList
```