TP 1 :

# 3 :

CREATE TABLE name\_basics (

nconst VARCHAR(20) PRIMARY KEY,

primaryName VARCHAR(1024),

birthYear INTEGER,

deathYear INTEGER,

primaryProfession TEXT,

knownForTitles TEXT

);

CREATE TABLE title\_basics (

tconst VARCHAR(20) PRIMARY KEY,

titleType VARCHAR(50),

primaryTitle VARCHAR(1024),

originalTitle VARCHAR(1024),

isAdult BOOLEAN,

startYear INTEGER,

endYear INTEGER,

runtimeMinutes INTEGER,

genres TEXT

);

CREATE TABLE title\_akas (

titleId VARCHAR(20),

ordering INTEGER,

title VARCHAR(1024),

region VARCHAR(10),

language VARCHAR(20),

types TEXT,

attributes TEXT,

isOriginalTitle BOOLEAN,

PRIMARY KEY (titleId, ordering)

);

CREATE TABLE title\_crew (

tconst VARCHAR(20) PRIMARY KEY,

directors TEXT,

writers TEXT,

FOREIGN KEY (tconst) REFERENCES title\_basics(tconst)

);

CREATE TABLE title\_episode (

tconst VARCHAR(20) PRIMARY KEY,

parentTconst VARCHAR(20),

seasonNumber INTEGER,

episodeNumber INTEGER,

FOREIGN KEY (tconst) REFERENCES title\_basics(tconst),

FOREIGN KEY (parentTconst) REFERENCES title\_basics(tconst)

);

CREATE TABLE title\_ratings (

tconst VARCHAR(20) PRIMARY KEY,

averageRating DECIMAL(3,1),

numVotes INTEGER,

FOREIGN KEY (tconst) REFERENCES title\_basics(tconst)

);

CREATE TABLE title\_principals (

tconst VARCHAR(20),

ordering INTEGER,

nconst VARCHAR(20),

category VARCHAR(100),

job VARCHAR(1024),

characters TEXT,

PRIMARY KEY (tconst, ordering),

FOREIGN KEY (tconst) REFERENCES title\_basics(tconst)

);

# 4 :

COPY title\_basics FROM '/import/title.basics.tsv'

WITH (FORMAT csv, DELIMITER E'\t', HEADER, NULL '\N', QUOTE E'\001');

COPY name\_basics FROM '/import/name.basics.tsv'

WITH (FORMAT csv, DELIMITER E'\t', HEADER, NULL '\N', QUOTE E'\001');

COPY title\_akas FROM '/import/title.akas.tsv'

WITH (FORMAT csv, DELIMITER E'\t', HEADER, NULL '\N', QUOTE E'\001');

COPY title\_crew FROM '/import/title.crew.tsv'

WITH (FORMAT csv, DELIMITER E'\t', HEADER, NULL '\N', QUOTE E'\001');

COPY title\_episode FROM '/import/title.episode.tsv'

WITH (FORMAT csv, DELIMITER E'\t', HEADER, NULL '\N', QUOTE E'\001');

COPY title\_ratings FROM '/import/title.ratings.tsv'

WITH (FORMAT csv, DELIMITER E'\t', HEADER, NULL '\N', QUOTE E'\001');

COPY title\_principals FROM '/import/title.principals.tsv'

WITH (FORMAT csv, DELIMITER E'\t', HEADER, NULL '\N', QUOTE E'\001');

# 5 :

11651827

"tvEpisode" 8966036

"short" 1060458

"movie" 714522

"video" 310547

"tvSeries" 281507

"tvMovie" 150916

"tvMiniSeries" 61681

"tvSpecial" 52539

"videoGame" 42983

"tvShort" 10637

"tvPilot" 1

"Famous Patrick (A Super SpongeBob Fan Movie)" 10.0 12

"Mother Mary" 10.0 633

"P Rell: Crucifix" 10.0 16

"Parrot Kindergarten" 10.0 6

"Kousalya Tanaya Ragava" 10.0 1145

"Through the Pain" 10.0 8

"Suthradaari" 10.0 99

"Regal vs. Rebel" 10.0 5

"Lost" 10.0 5

"Third Generation Successor Sara: Lustful Nightshade" 10.0 5

# 6 :

"Limit (cost=22413.29..22467.82 rows=10 width=27) (actual time=36.407..39.790 rows=10 loops=1)"

" -> Nested Loop (cost=22413.29..570393.42 rows=100499 width=27) (actual time=36.406..39.787 rows=10 loops=1)"

" -> Gather Merge (cost=22412.86..34117.62 rows=100499 width=16) (actual time=36.381..38.443 rows=10 loops=1)"

" Workers Planned: 2"

" Workers Launched: 2"

" -> Sort (cost=21412.83..21517.52 rows=41875 width=16) (actual time=28.640..28.663 rows=943 loops=3)"

" Sort Key: r.averagerating DESC"

" Sort Method: quicksort Memory: 3280kB"

" Worker 0: Sort Method: quicksort Memory: 1753kB"

" Worker 1: Sort Method: quicksort Memory: 1739kB"

" -> Parallel Seq Scan on title\_ratings r (cost=0.00..18198.13 rows=41875 width=16) (actual time=0.020..23.338 rows=32186 loops=3)"

" Filter: (numvotes > 1000)"

" Rows Removed by Filter: 490446"

" -> Index Scan using title\_basics\_pkey on title\_basics b (cost=0.43..5.34 rows=1 width=31) (actual time=0.133..0.133 rows=1 loops=10)"

" Index Cond: ((tconst)::text = (r.tconst)::text)"

"Planning Time: 0.144 ms"

"Execution Time: 40.068 ms"