Portfolio Opdracht 3

Dataset

De dataset waarmee we gaan werken voor deze portfolio opdracht is een overzicht van zo'n ~10 000 films, hun Rotten Tomatoes score, en op welke streaming platformen deze beschikbaar zijn. Deze dataset komt van Kaggle (link).

Aanpassingen

Origineel had de dataset 11 kolommen:

Row ID	Een uniek ID beginnend van 0
ID	Een uniek ID beginnend van 1
Title	Naam van de film
Year	Jaar van Uitgave
Age	Leeftijd rating van de film.
Rotten Tomatoes	De score van Rotten Tomatoes. Tekst met als format: ##/100
# Netflix	Is de film beschikbaar op dit platform? Waardes: 0 of 1
# Hulu	Is de film beschikbaar op dit platform? Waardes: 0 of 1
# Prime Video	Is de film beschikbaar op dit platform? Waardes: 0 of 1
# Disney+	Is de film beschikbaar op dit platform? Waardes: 0 of 1
Type	Een lege kolom

De onderdelen dat we gaan aanpassen om de data bruikbaarder te maken zijn:

- We gaan de twee eerste kolommen laten vallen en de naam gebruiken als ID
- De leeftijd rating is maar bij de helft van de records ingevuld, deze gaan we dus ook weglaten
- De Rotten Tomatoes scores heeft momenteel een onhandige formatering (##/100), dit gaan we veranderen naar een nummer van 0 tot 100
- De kolom Type wordt niet gebruikt, dus deze gaan we ook verwijderen.
- Sommige filmnamen beginnen met een #, dus verwijderen we het comment symbool in de read & write options van CSV edit.

Onze uiteindelijke dataset heeft dus de volgende kolommen:

Title	Naam van de film
Year	Jaar van Uitgave
Rotten Tomatoes	De score van Rotten Tomatoes als geheel getal 0 - 100
# Netflix	Is de film beschikbaar op dit platform? Waardes: 0 of 1
# Hulu	Is de film beschikbaar op dit platform? Waardes: 0 of 1
# Prime Video	Is de film beschikbaar op dit platform? Waardes: 0 of 1
# Disney+	Is de film beschikbaar op dit platform? Waardes: 0 of 1

Tabel importcsv

Onze tabel gaat dus ook 7 kolommen hebben:

Title varchar(150) NOT NULL
De langste filmnaam in de dataset is 104 karakters lang, een varchar(150) geeft wat extra speling. We gebruiken de naam als ID, dus de waarde mag nooit NULL zijn.

Year smallint NOT NULL
We hebben geen volledige datums alleen maar de jaartallen, een smallint is daar ruim groot genoeg voor, en laat ons nog steeds toe om de jaartallen met elkaar te vergelijken. In deze dataset is het jaar ook altijd ingevuld.

Score smallint
De Rotten Tomatoes score is een waarde van 0 tot 100, smallint is dus meer dan groot genoeg. Op sommige rijen in de dataset is er geen score ingevuld, dus we laten NULL toe als waarde.

Netflix boolean NOT NULL
Hulu boolean NOT NULL
Prime Video boolean NOT NULL
Disney+ boolean NOT NULL
Voor deze 4 kolommen gebruiken we een boolean: Of ze hebben de film, of ze hebben ze niet. Het datatype boolean aanvaard 1 en 0 als een alternatief voor true en false. Deze waardes zijn ook altijd ingevuld in de dataset.

Queries

Create

```
CREATE TABLE r1055651.importcsv (
        name            varchar(150) NOT NULL,
        year            smallint NOT NULL,
        score           smallint,
        netflix         boolean NOT NULL,
        hulu            boolean NOT NULL,
        prime           boolean NOT NULL,
        “disney+”       boolean NOT NULL,
        CONSTRAINT pk_movies PRIMARY KEY ( name )
);
```

Select

Een overzicht met het aantal films per streaming service:

```
SELECT SUM(CASE WHEN netflix THEN 1 ELSE 0 END) AS "Movies on Netflix",
SUM(CASE WHEN hulu THEN 1 ELSE 0 END) AS "Movies on Hulu",
SUM(CASE WHEN prime THEN 1 ELSE 0 END) AS "Movies on Prime",
SUM(CASE WHEN "disney+" THEN 1 ELSE 0 END) AS "Movies on Disney+"
FROM r1055651.importcsv;
```

Een overzicht van de 10 beste films:

```
SELECT name, score
FROM r1055651.importcsv
WHERE score IS NOT NULL
ORDER BY score DESC
LIMIT 10;
```

Een overzicht van de jaren waar de gemiddelde scores van al de uitgekomen films groter of gelijk is aan 50:

```
SELECT year, ROUND(AVG(score), 2) as "Average score"
FROM r1055651.importcsv
GROUP BY year
HAVING ROUND(AVG(score), 2) >= 50
ORDER BY AVG(score) DESC;
```

Het aantal slechte films (score lager dan 50):

```
SELECT count(*) as "Amount of bad movies"
FROM r1055651.importcsv
WHERE score < 50;
```

Een overzicht van de gemiddelde scores per streaming service:

```
SELECT
(SELECT ROUND(AVG(score), 2)
FROM r1055651.importcsv
WHERE netflix IS true
GROUP BY netflix) as "Average Netflix score",
(SELECT ROUND(AVG(score), 2)
FROM r1055651.importcsv
WHERE hulu IS true
GROUP BY hulu) as "Average Hulu score",
```

```
(SELECT ROUND(AVG(score), 2)
FROM r1055651.importcsv
WHERE prime IS true
GROUP BY prime) as "Average Prime score",
(SELECT ROUND(AVG(score), 2)
FROM r1055651.importcsv
WHERE "disney+" IS true
GROUP BY "disney+") as "Average Disney+ score";
```