**Anime Recommendation database 2020**

Impala

1. แต่ละปีมีอนิเมะใหม่กี่เรื่อง

select count() as count\_of\_anime, sub.Year

from

(SELECT substring(premiered, -1,4) AS Year

FROM anime

where premiered <> 'Unknown' )sub

group by sub.Year

1. อนิเมะคะแนน TOP 10 (1961-2021)

SELECT name,score FROM animelist

Where score > 1

ORDER BY score DESC

LIMIT 10;

1. อนิเมะคะแนน Top ของแต่ละซีซั่น

select animelist.name, sub.premiered, max\_score as score FROM (

select animelist.premiered, substring(animelist.premiered, -4, 4) as year, MAX(score) AS max\_score from animelist

where animelist.premiered is not NULL and score is not NULL and animelist.premiered != "Unknown"

group by animelist.premiered

) sub

join animelist on animelist.score = sub.max\_score and animelist.premiered = sub.premiered

order by sub.year

1. ตรวจหา Anime ที่ถูก Review bomb

SELECT

h.name AS name,

h.n/COUNT(animelist.rating)\*100 AS "Percent of review bomb suspect / all review",

h.n AS "Review bomb suspect count",

COUNT(animelist.rating) AS "Review count",

h.episodes AS "Episode(s)",

AVG(animelist.watched\_episodes) AS "Average episode watch",

AVG((animelist.watched\_episodes/h.episodes)\*100) AS "Average watched progress"

FROM (

SELECT COUNT(r.rating) AS n, anime.name, anime.mal\_id, anime.episodes

FROM animelist r

JOIN anime ON anime.MAL\_ID = r.anime\_id

WHERE

(r.watched\_episodes BETWEEN 0 AND 1) AND (r.rating BETWEEN 1 AND 2) AND anime.episodes > 1

GROUP BY anime.name, anime.mal\_id, anime.episodes

) h

JOIN reviewlist ON h.MAL\_ID = reviewlist.anime\_id

GROUP BY h.name, animelist.anime\_id, h.n, h.episodes

ORDER BY h.n/COUNT(animelist.rating) DESC

1. ตรวจหา ประเภท anime ที่คนดองมากที่สุด

SELECT genre, dongCount as "Number of on hold and watch later", total as "Number of all reviews", dongCount\*100/total as "Percent of on hold and watch later" FROM (

SELECT SUM(sub.c) AS dongCount, anime\_genres.genre

FROM (

SELECT animelist.name, sub.c

FROM (

SELECT COUNT(\*) AS c, sub2.anime\_id

FROM (

SELECT watching\_status, reviewList.anime\_id

FROM reviewList

) sub2

WHERE sub2.watching\_status = 6 OR sub2.watching\_status = 3

GROUP BY sub2.anime\_id

) sub

INNER JOIN animelist ON sub.anime\_id = animelist.mal\_id

) sub

INNER JOIN anime\_genres ON anime\_genres.name = sub.name

GROUP BY anime\_genres.genre

) Dong

INNER JOIN (

SELECT SUM(sub.c) AS total, anime\_genres.genre as g

FROM (

SELECT animelist.name, sub.c

FROM (

SELECT COUNT(\*) AS c, sub2.anime\_id

FROM (

SELECT watching\_status, reviewList.anime\_id

FROM reviewList

) sub2

GROUP BY sub2.anime\_id

) sub

INNER JOIN animelist ON sub.anime\_id = animelist.mal\_id

) sub

INNER JOIN anime\_genres ON anime\_genres.name = sub.name

GROUP BY anime\_genres.genre

) AS allReview ON allReview.g = Dong.genre

ORDER BY dongCount\*100/total DESC

Spark

1. เรียงอันดับ 10 อนิเมะที่มีตอนเยอะสุด

SELECT name, episodes

FROM animelist

ORDER BY episodes DESC

LIMIT 10;

1. ช่วงเวลาที่เมื่อ anime ออกแล้วมีคนดูมากสุด 10 อันดับ

SELECT count(reviewlist.user\_id) as no\_user,

animelist.premiered

FROM reviewlist

LEFT JOIN animelist

ON reviewlist.anime\_id = animelist.mal\_id

WHERE reviewlist.watching\_status = 1

OR reviewlist.watching\_status = 2

AND animelist.premiered <> 'Unknown'

GROUP BY animelist.premiered

ORDER BY count(reviewlist.user\_id) DESC

LIMIT 10;

1. ตรวจหา Review ที่มีคุณภาพ

SELECT ep.user AS user, ep.name AS aniname, ep.rating, ROUND((ep.watched\_episodes\*100)/ep.epi, 2) AS watchComp

FROM (SELECT reviewlist.user\_id as user,

reviewlist.rating as rating,

reviewlist.watched\_episodes as watched\_episodes,

animelist.mal\_id as id,

animelist.name as name,

animelist.episodes as epi

FROM animelist

INNER JOIN reviewlist ON reviewlist.anime\_id = animelist.mal\_id) AS ep

WHERE ep.watched\_episodes > 1

AND ROUND((ep.watched\_episodes\*100)/ep.epi, 2) < 101

AND ROUND((ep.watched\_episodes\*100)/ep.epi, 2) > 85

ORDER BY ep.user ASC