# 5η Εργασία: Βελτιστοποίηση Ερωτημάτων και Εναύσματα

## Ιωάννα Καλαντζή p3210057

## Οδυσσέας Τζαμάκος p3210196

### Ερώτημα Α

Να δοθεί screenshot του πλάνου εκτέλεσης του ερωτήματος:

A screenshot of a computer program

Description automatically generated with low confidence

### Ερώτημα Β

1. Να φτιάξετε **2 ευρετήρια** τα οποία επιταχύνουν την εκτέλεση του ερωτήματος.

CREATE INDEX idx\_movie\_title ON movie(id);

CREATE INDEX idx\_movie\_cast\_movie\_id ON movie\_cast(movie\_id);

1. Να δοθεί το αντίστοιχο πλάνο εκτέλεσης του ερωτήματος αφού έχουν φτιαχτεί τα ευρετήρια.

A screenshot of a computer program

Description automatically generated with low confidence

1. Ποιές είναι οι αλλαγές που έχουν γίνει στο πλάνο εκτέλεσης;

Δεν παρατηρούμε κάποια αξιοσημείωτη αλλαγή στην εκτέλεση, απλώς μειώνεται ο χρόνος εκτέλεσης καθώς δεν γίνεται σάρωση της σχέσης αλλά επιλεκτική προσπέλαση στοιχείων.

### Ερώτημα C

Χρησιμοποιώντας τις εντολές GROUP BY και HAVING στον πίνακα Ratings:

1. Ποιό είναι το ερώτημα το οποίο επιστρέφει τον μέσο όρο βαθμολογίας ανά ταινία για ταινίες με *μέση βαθμολογία πάνω από 4;*

SELECT m.title, AVG(r.rating) AS AVG\_Rating

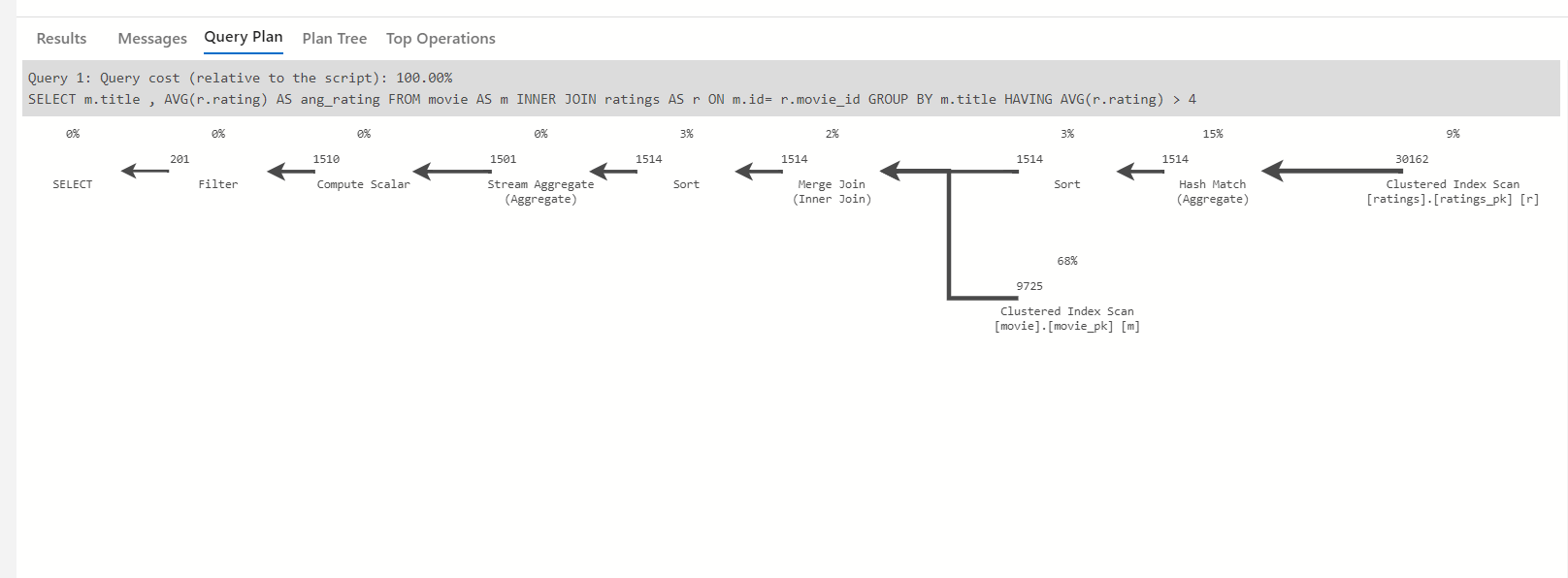
FROM movie AS m

INNER JOIN ratings AS r ON m.id = r.movie\_id

GROUP BY m.title

HAVING AVG(r.rating) > 4;

1. Να τρέξετε το ερώτημα και να επιστρέψετε το αντίστοιχο πλάνο εκτέλεσης.



1. Να δημιουργηθεί ένα *ευρετήριο επικάλυψης* το οποίο μπορεί να χρησιμοποιηθεί για την απάντηση στο συγκεκριμένο ερώτημα.

CREATE INDEX idx\_ratings\_movie\_rating ON ratings(user\_id, movie\_id , rating);

1. Καταγράψτε το πλάνο εκτέλεσης με το πλάνο επικάλυψης,

A screen shot of a computer

Description automatically generated with medium confidence

1. τι παρατηρείτε σε αυτό ως προς το διάβασμα του πίνακα Ratings και γιατί συμβαίνει αυτό;

Το αποτέλεσμα δεν αλλάζει! Αλλάζει όμως ο τρόπος και η ταχύτητα της προσπέλασης και βελτιστοποιείται η διαδικασία.

### Ερώτημα D

1. Να δοθεί η εντολή Alter Table για την δημιουργία του γνωρίσματος AVG\_Rating στον πίνακα Movie.

ALTER TABLE movie

ADD AVG\_Rating float;

1. Χρησιμοποιώντας την εντολή UPDATE, αρχικοποιήστε το συγκεκριμένο γνώρισμα με την μέση βαθμολογία ανά ταινία σε περίπτωση που αυτή υπάρχει και Null διαφορετικά.

UPDATE movie

SET AVG\_Rating = (

    SELECT AVG(rating)

    FROM ratings

    WHERE ratings.movie\_id = movie\_id

)

WHERE id IN (

  SELECT movie\_id

    FROM ratings

)OR AVG\_Rating IS NULL;

1. Γράψτε ένα *έναυσμα* (trigger) το οποίο για κάθε εισαγωγή στον πίνακα Ratings ενημερώνει το γνώρισμα AVG\_Rating του πίνακα Movie με την μέση βαθμολογία μόνο των ταινιών των οποίων η βαθμολογία άλλαξε.

CREATE TRIGGER UpdateAvgRating

ON Ratings

AFTER INSERT, UPDATE, DELETE

AS

BEGIN

    UPDATE movie

    SET AVG\_Rating = (

        SELECT AVG(rating)

        FROM ratings

        WHERE ratings.movie\_id = movie\_id

    )

    WHERE id IN (

        SELECT DISTINCT movie\_id

        FROM inserted

    );

END;

1. Ποιό από τα ευρετήρια που φτιάξατε βοηθάει στην εκτέλεση του συγκεκριμένου trigger.

Από τα ευρετήρια που κατασκευάσαμε το πιο σημαντικό για την υλοποίηση του trigger αυτού είναι το idx\_ratings\_movie\_rating που βασίζεται στον πίνακα ratings και τις στήλες movie\_id και rating, καθώς μπορούμε πιο εύκολα να προσπελάσουμε τις στήλες αυτές σε μικρότερο χρόνο και να τις αξιοποιήσουμε στην νέα στήλη AVG\_Rating.

### Ερώτημα Ε

Περιγράψτε κάποιον εναλλακτικό τρόπο με τον οποίο θα μπορούσατε να υλοποιήσετε το trigger σας, χωρίς να χρειαστεί να τρέχετε κάποια group by συνάρτηση στα υπάρχοντα δεδομένα.