

Databases Assignment 1

Description

For this assignment, you must analyze and improve the database performance (considering only the performance for READ operations) for three queries that are frequently executed in your system. You can only create indexes, so you should write DDL instructions that will be executed on the database by the DBA of your company.

The database management system to be used is a recent version of PostgreSQL with the DVD Rental database provided.

Query 1 (auditing staff):

```
EXPLAIN ANALYZE
SELECT r1.staff_id, p1.payment_date
FROM rental r1, payment p1
WHERE r1.rental_id = p1.rental_id AND
NOT EXISTS (SELECT 1 FROM rental r2, customer c WHERE r2.customer_id =
c.customer_id and active = 1 and r2.last_update > r1.last_update);
```

Query 2 (popular movies year by year):

```
EXPLAIN ANALYZE
SELECT title, release_year
FROM film f1
WHERE f1.rental_rate > (SELECT AVG(f2.rental_rate) FROM film f2 WHERE
f1.release_year = f2.release_year);
```

Query 3 (how much movies [which have not been rented by teenagers who rented the movie because there are actors with the same name as them] earn for the store):

```
EXPLAIN ANALYZE
SELECT f.title, f.release_year, (SELECT SUM(p.amount) FROM payment p, rental
r1, inventory i1 WHERE p.rental_id = r1.rental_id AND r1.inventory_id =
i1.inventory_id AND i1.film_id = f.film_id)
FROM film f
WHERE NOT EXISTS (SELECT c.first_name, count(*) FROM customer c, rental r2,
inventory i1, film f1, film_actor fa, actor a
```

```
WHERE c.customer_id = r2.customer_id AND r2.inventory_id = i1.inventory_id AND  
i1.film_id = f1.film_id and f1.rating in ('PG-13','NC-17') AND f1.film_id =  
fa.film_id AND f1.film_id = f.film_id AND fa.actor_id = a.actor_id and  
a.first_name = c.first_name GROUP BY c.first_name HAVING count(*) >2);
```

Deliverables

One single file:

- YOUR_NAME_SURNAME.sql containing all the DDL instructions

Failing to attend the specification above will incur in -50%. If you create tables, columns, temporary tables or views, your grade will be ZERO. **You can only create indexes.**

Evaluation / grading

For each query, the performance will be measured before and after your indexes were created. For each query, there will be some percentage of improvement in the runtime: X_1 , X_2 and X_3 . Your metric as a student is $X = X_1 + X_2 + X_3$.

Next, students will be sorted according to the metric X , in decreasing order. Then, the grade is as follows:

- top 30% of the students get 100/100 for this assignment
- next 40% of students get 85/100 for this assignment
- next 25% get 60/100
- last 5% of students get 50/100

Plagiarism

The assignment is individual. In the case of signs of plagiarism (at the discretion of the instructors), all the involved students will have grade 0 (zero).

Deadline

Work submitted after 25 st of April will receive **ZERO**, no matter the quality or how much effort you put into it. No appeal.