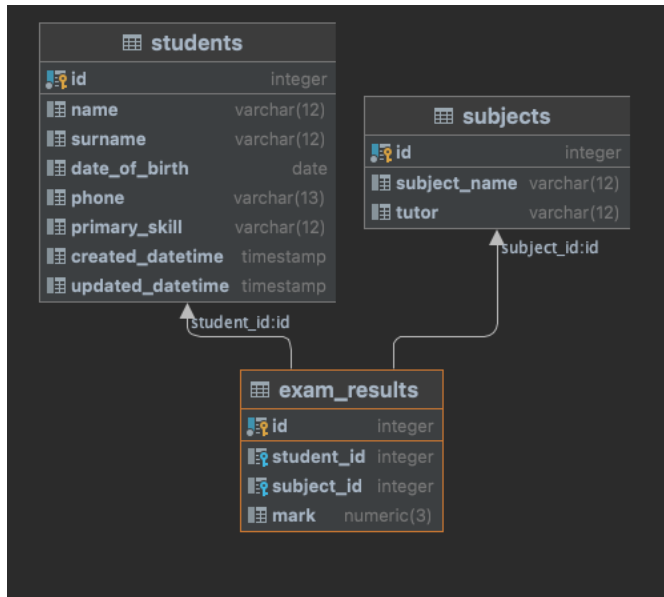# Java Global Mentoring Program A2 [Q4 2021]
Databases module

Before index


```
explain analyse INSERT INTO students (name, surname, date_of_birth, phone,
primary_skill)
SELECT
        substr(md5(random()::text), 1, 10),
        substr(md5(random()::text), 1, 10),
        (DATE '2010-01-01' - (random() * 10000)::integer),
        (random()*999999999)::integer,
        substr(md5(random()::text), 1, 5)
FROM generate_series(1, 100000);

explain analyse INSERT INTO subjects (subject_name, tutor)
SELECT
        substr(md5(random()::text), 1, 10),
        substr(md5(random()::text), 1, 10)
FROM generate_series(1, 1000);

explain analyse INSERT INTO exam_results (student_id, subject_id, mark)
SELECT
        (random()*100000)::integer,
        (random()*1000)::integer,
        (random()*100)::integer
FROM generate_series(1, 1000000);
```

```
          ▐▊ QUERY PLAN
1  Insert on students  (cost=0.00..10500.00 rows=0 width=0) (actual time=4549.333..4549.368 rows=0 loops=1)
2    ->  Subquery Scan on "*SELECT*"  (cost=0.00..10500.00 rows=100000 width=194) (actual time=10.876..3173.621 rows=100000 loops=1)
3          ->  Function Scan on generate_series  (cost=0.00..6500.00 rows=100000 width=104) (actual time=10.694..1317.800 rows=100000 loops=1)
4  Planning Time: 0.143 ms
5  Execution Time: 4551.333 ms
```
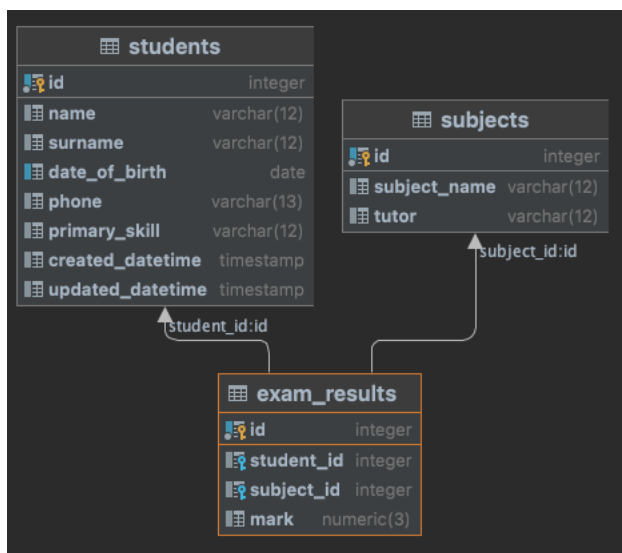
```
          ▐▊ QUERY PLAN
1  Insert on subjects  (cost=0.00..55.00 rows=0 width=0) (actual time=46.324..46.359 rows=0 loops=1)
2    ->  Subquery Scan on "*SELECT*"  (cost=0.00..55.00 rows=1000 width=88) (actual time=0.173..31.737 rows=1000 loops=1)
3          ->  Function Scan on generate_series  (cost=0.00..35.00 rows=1000 width=64) (actual time=0.087..12.861 rows=1000 loops=1)
4  Planning Time: 0.097 ms
5  Execution Time: 46.441 ms
```

```
          ▐▊ QUERY PLAN
1  Insert on exam_results  (cost=0.00..52500.00 rows=0 width=0) (actual time=38603.714..38603.749 rows=0 loops=1)
2    ->  Subquery Scan on "*SELECT*"  (cost=0.00..52500.00 rows=1000000 width=24) (actual time=74.682..26012.082 rows=1000000 loops=1)
3          ->  Function Scan on generate_series  (cost=0.00..32500.00 rows=1000000 width=12) (actual time=74.570..8664.472 rows=1000000 loops=1)
4  Planning Time: 0.077 ms
5  Trigger for constraint student_fk: time=25957.300 calls=1000000
6  Trigger for constraint subject_fk: time=23724.052 calls=1000000
7  Execution Time: 103786.037 ms
```

After indexing

```
create index date_of_birth_index on students (date_of_birth);
```



```
          ▐▊ QUERY PLAN
1  Insert on students  (cost=0.00..10500.00 rows=0 width=0) (actual time=4836.150..4836.185 rows=0 loops=1)
2    ->  Subquery Scan on "*SELECT*"  (cost=0.00..10500.00 rows=100000 width=194) (actual time=9.225..3188.650 rows=100000 loops=1)
3          ->  Function Scan on generate_series  (cost=0.00..6500.00 rows=100000 width=104) (actual time=9.077..1313.615 rows=100000 loops=1)
4  Planning Time: 0.286 ms
5  Execution Time: 4837.425 ms
```

```
drop index date_of_birth_index
```