LAB 4

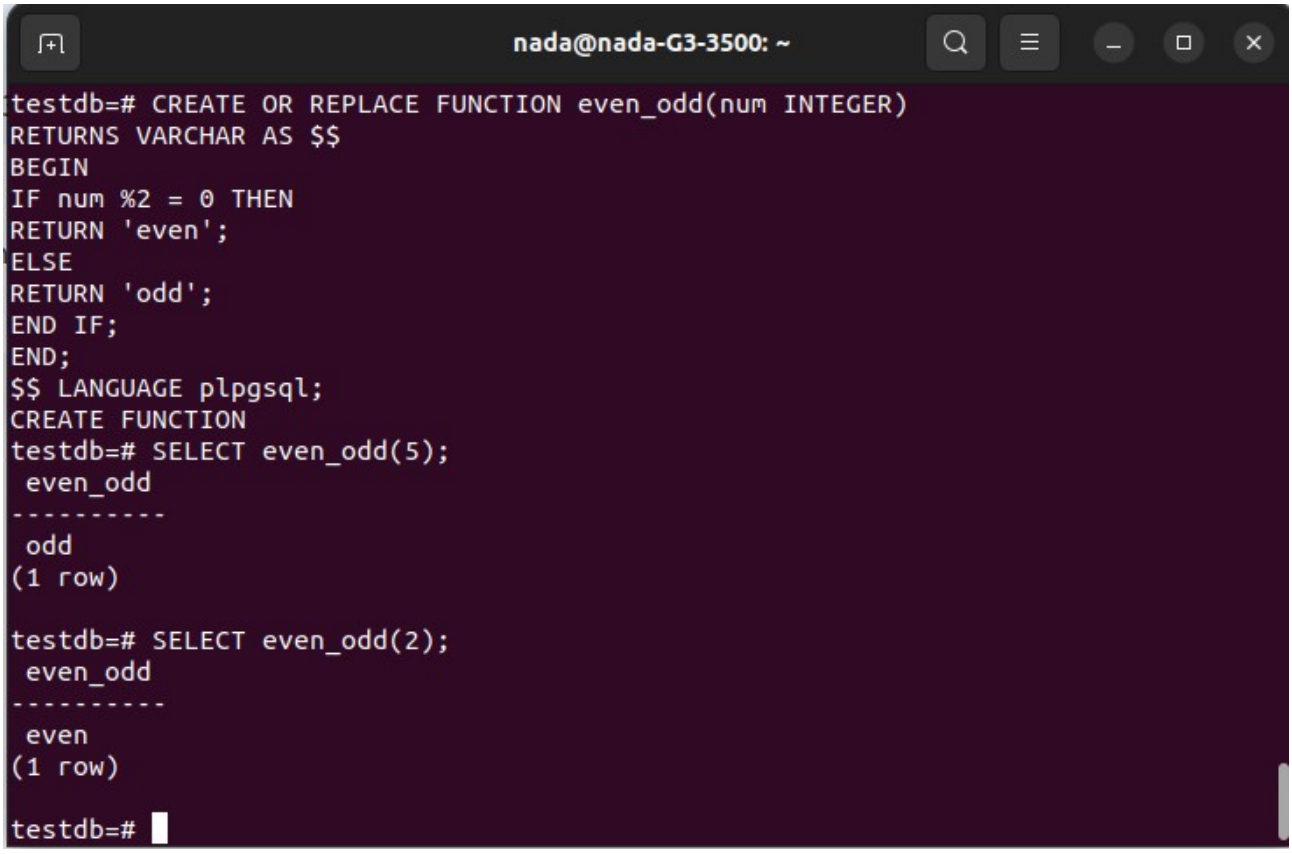
1. Create a multiply function that accepts two numbers and returns their product.

```
testdb=# CREATE FUNCTION multiply_nums(num1 integer, num2 integer)
RETURNS integer AS $$ BEGIN
RETURN num1 * num2; END;
$$ LANGUAGE plpgsql;
CREATE FUNCTION
testdb=# SELECT multiply_nums(10,3);
 multiply_nums
---------------
            30
(1 row)

testdb=#
```

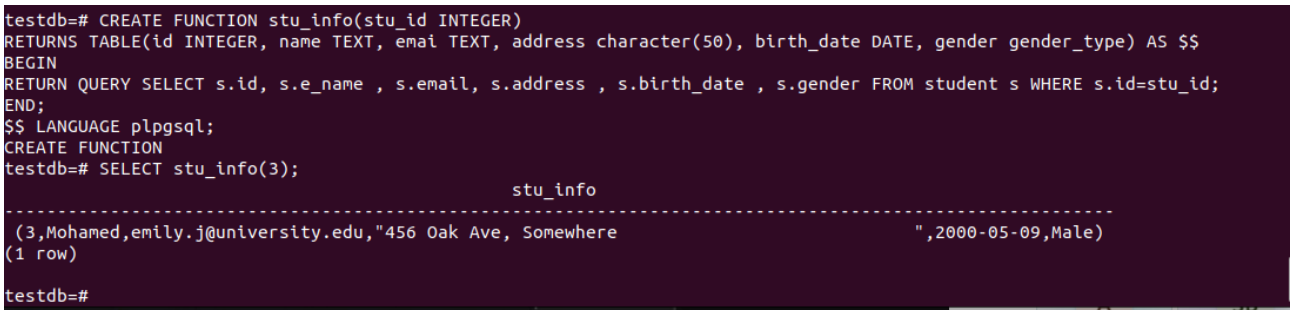
2. Create a hello_world function that takes a name as input and returns a personalized welcome message for that name.

```
testdb=# CREATE FUNCTION hello_world (name VARCHAR(50))
RETURNS VARCHAR(50) AS $$
BEGIN
RETURN CONCAT('Welcome ',  name);
END;
$$ LANGUAGE plpgsql;
CREATE FUNCTION
testdb=# SELECT hello_world(Nada);
ERROR:  column "nada" does not exist
LINE 1: SELECT hello_world(Nada);
                           ^
testdb=# SELECT hello_world('Nada');
 hello_world
--------------
 Welcome Nada
(1 row)

testdb=#
```

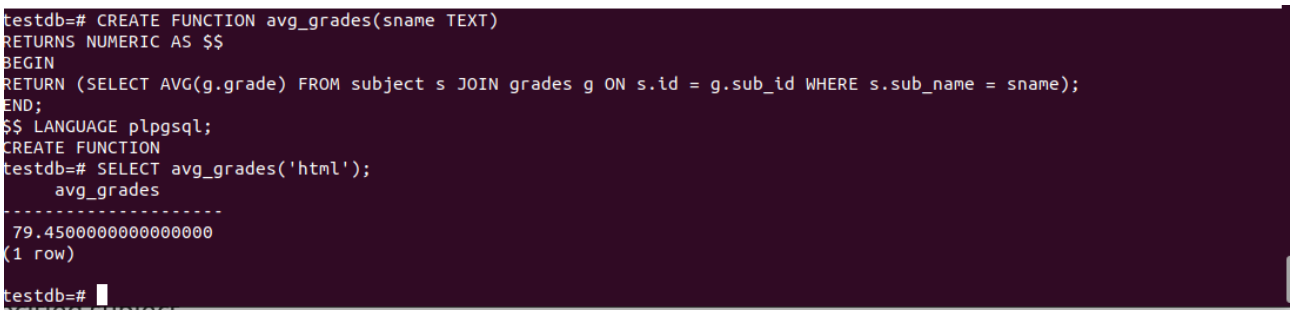
3. Create a function that accepts a number and determines whether it is odd or even.



4. Create a function that takes a Student ID as input and retrieves all information related to that student.



5. Implement a function that takes the name of a subject and calculates the average grades for that subject.



6. Create a trigger to automatically save deleted student records from the Student table to the Deleted_Students table.

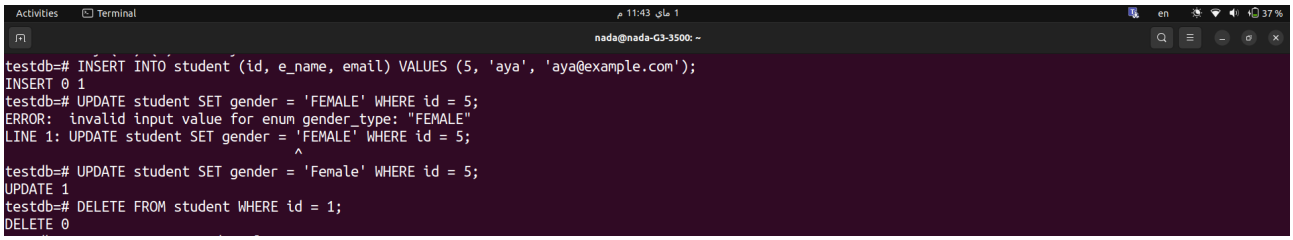
```
testdb=# CREATE TABLE deleted_student (id INTEGER, name TEXT, email TEXT, address character(50), track_id INTEGER, birth_date DATE, gender gender_type, deleted_at TIMESTAMP DEFAULT CURRENT_T
IMESTAMP, deleted_by TEXT DEFAULT current_user);
CREATE TABLE
testdb=# CREATE FUNCTION save_del_studs()
RETURNS TRIGGER AS $$
BEGIN
INSERT INTO deleted_student
SELECT OLD.*, CURRENT_TIMESTAMP, current_user;
DELETE FROM stu_sub WHERE stu_id = OLD.id;
RETURN OLD;
END;
$$ LANGUAGE plpgsql;
CREATE FUNCTION
testdb=# CREATE TRIGGER trigg_deleted_students
BEFORE DELETE ON student
FOR EACH ROW EXECUTE FUNCTION save_del_studs();
CREATE TRIGGER
testdb=# DELETE FROM student WHERE student.id = 2;
DELETE 1
testdb=#
testdb=# SELECT * FROM student;
testdb=# SELECT * FROM deleted_student;
 id |    name    |           email           |                         address                          | track_id | birth_date | gender |         deleted_at         | deleted_by
----+------------+---------------------------+----------------------------------------------------------+----------+------------+--------+----------------------------+------------
  2 | John Smith | john.smith@university.edu | 123 Main St, Anytown                                     |        1 | 2001-05-30 | Male   | 2025-05-01 22:21:16.801026 | postgres
(1 row)

testdb=# select * from stu_sub;
 stu_id | sub_id
--------+--------
      1 |      2
      3 |      3
(2 rows)

testdb=# 
```

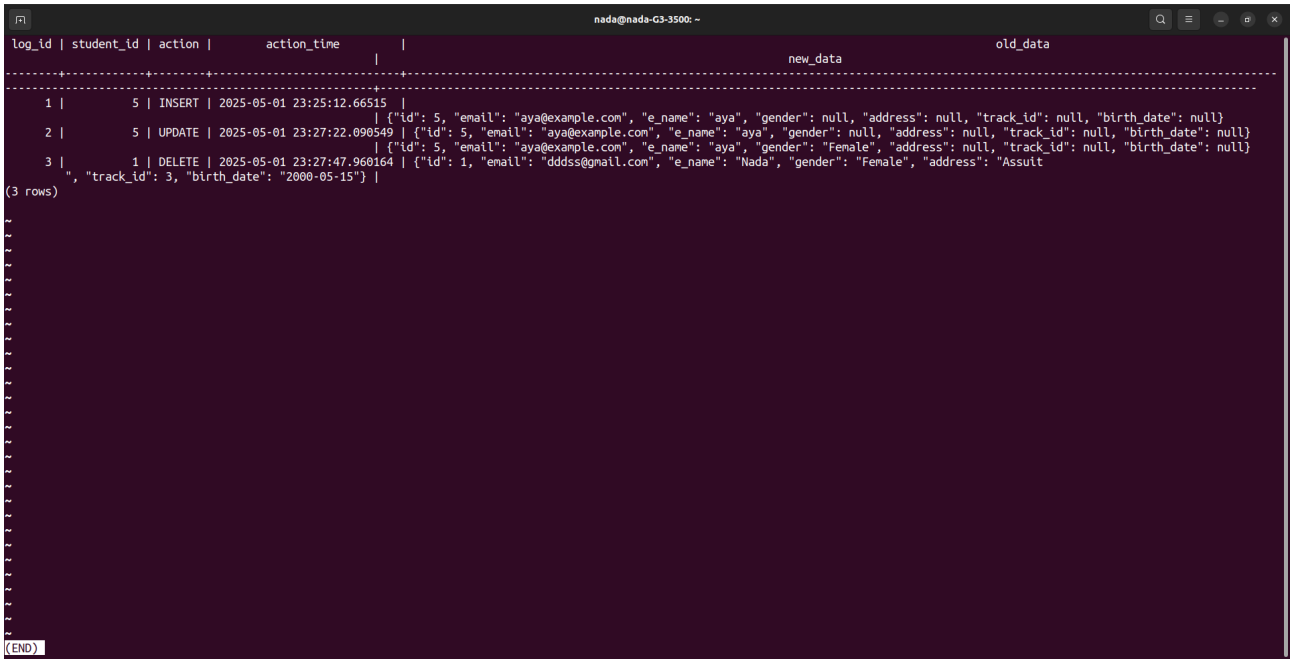
```
nada@nada-G3-3500: ~
 id | e_name  |         email          |                         address                          | track_id | birth_date | gender
----+---------+------------------------+----------------------------------------------------------+----------+------------+--------
  3 | Mohamed | emily.j@university.edu | 456 Oak Ave, Somewhere                                   |        2 | 2000-05-09 | Male
  1 | Nada    | dddss@gmail.com        | Assuit                                                   |        3 | 2000-05-15 | Female
(2 rows)

~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
(END)
```

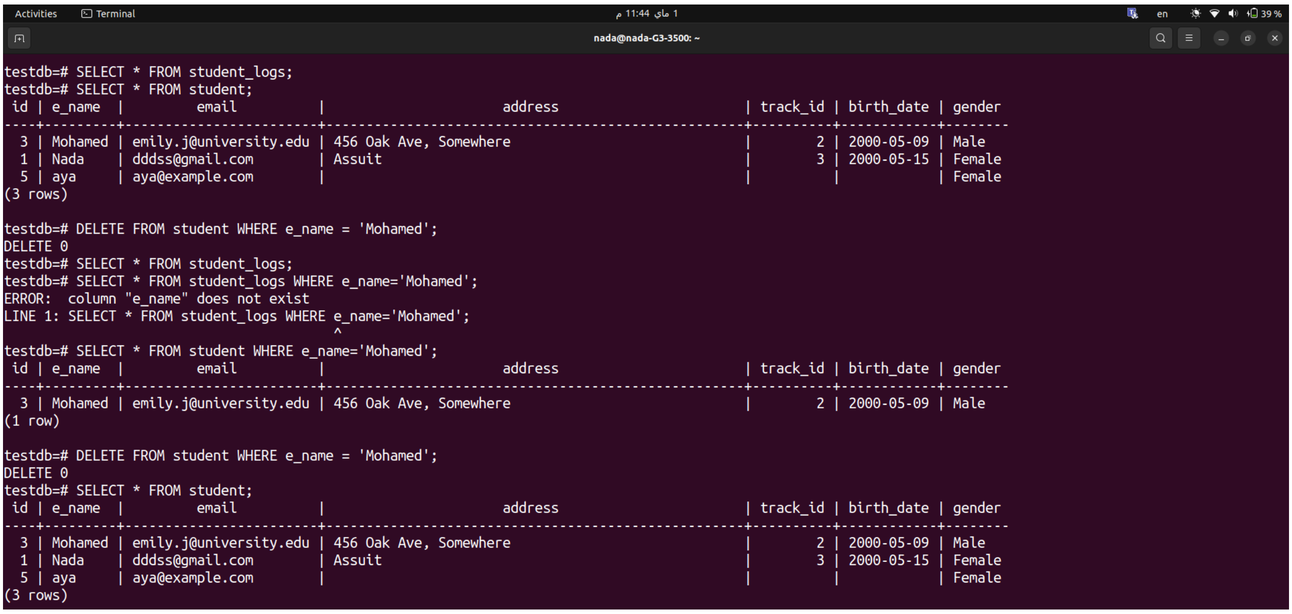
7. Create a trigger to monitor changes made to the student table, including additions, updates, and deletions. This trigger will record the time of each action and provide a description of the action in another table.

```
Activities   Terminal                1 ماي 11:39 م                en   34 %
nada@nada-G3-3500: ~
testdb=# CREATE TABLE student_logs (
    log_id SERIAL PRIMARY KEY,
    student_id INTEGER,
    action TEXT CHECK (action IN ('INSERT', 'UPDATE', 'DELETE')),
    action_time TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    old_data JSONB,
    new_data JSONB
);
CREATE TABLE
```

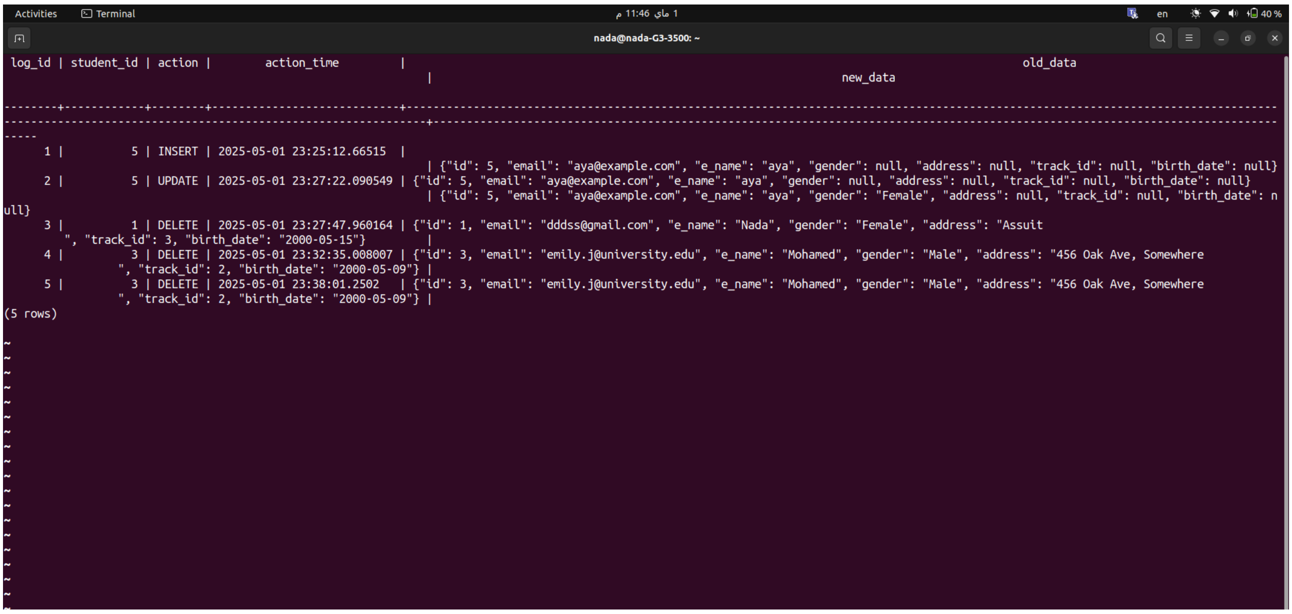
```
testdb=# CREATE FUNCTION monitor_changes()
RETURNS TRIGGER AS $$
BEGIN
IF TG_OP = 'INSERT' THEN
INSERT INTO student_logs(student_id, action, new_data)
VALUES (NEW.id, 'INSERT', to_jsonb(NEW));
ELSIF TG_OP = 'UPDATE' THEN
INSERT INTO student_logs(student_id, action, old_data, new_data)
VALUES (OLD.id, 'UPDATE', to_jsonb(OLD), to_jsonb(NEW));
ELSIF TG_OP = 'DELETE' THEN
INSERT INTO student_logs(student_id, action, old_data)
VALUES (OLD.id, 'DELETE', to_jsonb(OLD));
END IF;
RETURN NULL;
END;
$$ LANGUAGE plpgsql;
CREATE FUNCTION
testdb=# CREATE TRIGGER log_insert_update
AFTER INSERT OR UPDATE ON student
FOR EACH ROW EXECUTE FUNCTION monitor_changes();
CREATE TRIGGER
testdb=# CREATE TRIGGER log_delete
BEFORE DELETE ON student
FOR EACH ROW EXECUTE FUNCTION monitor_changes();
CREATE TRIGGER
```








```
testdb=# SELECT * FROM student_logs;
testdb=# SELECT * FROM student;
 id | e_name  |         email          |                         address                          | track_id | birth_date | gender
----+---------+------------------------+----------------------------------------------------------+----------+------------+--------
  3 | Mohamed | emily.j@university.edu | 456 Oak Ave, Somewhere                                   |        2 | 2000-05-09 | Male
  1 | Nada    | dddss@gmail.com        | Assuit                                                   |        3 | 2000-05-15 | Female
  5 | aya     | aya@example.com        |                                                          |          |            | Female
(3 rows)

testdb=# DELETE FROM student WHERE e_name = 'Mohamed';
DELETE 0
testdb=# SELECT * FROM student_logs;
testdb=# SELECT * FROM student_logs WHERE e_name='Mohamed';
ERROR:  column "e_name" does not exist
LINE 1: SELECT * FROM student_logs WHERE e_name='Mohamed';
                                         ^
testdb=# SELECT * FROM student WHERE e_name='Mohamed';
 id | e_name  |         email          |                         address                          | track_id | birth_date | gender
----+---------+------------------------+----------------------------------------------------------+----------+------------+--------
  3 | Mohamed | emily.j@university.edu | 456 Oak Ave, Somewhere                                   |        2 | 2000-05-09 | Male
(1 row)

testdb=# DELETE FROM student WHERE e_name = 'Mohamed';
DELETE 0
testdb=# SELECT * FROM student;
 id | e_name  |         email          |                         address                          | track_id | birth_date | gender
----+---------+------------------------+----------------------------------------------------------+----------+------------+--------
  3 | Mohamed | emily.j@university.edu | 456 Oak Ave, Somewhere                                   |        2 | 2000-05-09 | Male
  1 | Nada    | dddss@gmail.com        | Assuit                                                   |        3 | 2000-05-15 | Female
  5 | aya     | aya@example.com        |                                                          |          |            | Female
(3 rows)
```




```
 log_id | student_id | action |        action_time         |                                                                old_data
                                                            |                                                        new_data

--------+------------+--------+----------------------------+--------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------+--------------------------------------------------------------------------------------------------------------------------------------
-----
      1 |          5 | INSERT | 2025-05-01 23:25:12.66515  |
                                                            | {"id": 5, "email": "aya@example.com", "e_name": "aya", "gender": null, "address": null, "track_id": null, "birth_date": null}
      2 |          5 | UPDATE | 2025-05-01 23:27:22.090549 | {"id": 5, "email": "aya@example.com", "e_name": "aya", "gender": null, "address": null, "track_id": null, "birth_date": null}
                                                            | {"id": 5, "email": "aya@example.com", "e_name": "aya", "gender": "Female", "address": null, "track_id": null, "birth_date": n
ull}
      3 |          1 | DELETE | 2025-05-01 23:27:47.960164 | {"id": 1, "email": "dddss@gmail.com", "e_name": "Nada", "gender": "Female", "address": "Assuit
        ", "track_id": 3, "birth_date": "2000-05-15"}       |
      4 |          3 | DELETE | 2025-05-01 23:32:35.008007 | {"id": 3, "email": "emily.j@university.edu", "e_name": "Mohamed", "gender": "Male", "address": "456 Oak Ave, Somewhere
               ", "track_id": 2, "birth_date": "2000-05-09"} |
      5 |          3 | DELETE | 2025-05-01 23:38:01.2502   | {"id": 3, "email": "emily.j@university.edu", "e_name": "Mohamed", "gender": "Male", "address": "456 Oak Ave, Somewhere
               ", "track_id": 2, "birth_date": "2000-05-09"} |
(5 rows)

~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
```