Ebram Rasmy task_day 3 lab 4

----------1----------

```
create or replace function multiplay(a integer , b integer)
returns integer as $$
begin
return a*b ;
end ;
$$ language plpgsql ;

select multiplay(5,5);
select multiplay(10,13);
```

2----------

```
create or replace function hello_world(name text)
returns text as $$
begin
return concat('Hello, ', name, '! Welcome to PostgreSQL.');
end;
$$ language plpgsql ;

select hello_world('Ebram')
```

3----------

```
create or replace function is_odd_or_even(num integer)
returns text as $$
begin
if num % 2 = 0 then
return 'Even';
else
return 'odd';
end if ;
end;
$$ language plpgsql;

select is_odd_or_even(152);
select is_odd_or_even(151);
```

4----------

```
create or replace function student_info(sid integer)
returns table (
    id integer,e_name varchar(50), email varchar(100),address text,  track_id integer,birth_date
date,gender varchar(10)) as $$
begin
    return query
    select student.id, student.e_name, student.email, student.address,student.track_id,
student.birth_date, student.gender
    from student
    where student.id = sid;
end;
$$ language plpgsql;
```

```
drop function if exists student_info(integer);

select * from student_info(1);
-------------------------5-----------------------------------------------------------------------------------
create or replace function avg_grade(sub_name_input text)
returns numeric as $$
declare
   avg_g numeric;
begin
   select avg(grades.grade)
   into avg_g
   from grades join subject  on grades.sub_id = subject.sid
   where subject.sub_name = sub_name_input;

   return avg_g;
end;
$$ language plpgsql;


select avg_grade('python');
--------------------------
6-----------------------------------------------------------------------------------------

create table deleted_students
(id integer,e_name varchar(100),email varchar(100),address text, track_id integer,birth_date date,
gender varchar(10));

create or replace function log_deleted()
returns trigger as $$
begin
        insert into deleted_students (id, e_name, email, address, track_id, birth_date, gender)
        values (old.id, old.e_name, old.email, old.address, old.track_id, old.birth_date, old.gender);
        return old;
end;
$$ language plpgsql;

create trigger trg_log_deleted before delete on student for each row
execute function log_deleted();


delete from student where id = 3;

select * from deleted_students;


-- alter table stu_sub
-- drop constraint stu_sub_stu_id_fkey;

-- alter table stu_sub
-- add constraint stu_sub_stu_id_fkey
-- foreign key (stu_id) references student(id)
```

```
-- on delete cascade;

-- alter table grades
-- drop constraint grades_stu_id_fkey;

-- alter table grades
-- add constraint grades_stu_id_fkey
-- foreign key (stu_id) references student(id)
-- on delete cascade;
------------------------
7-------------------------------------------------------------------------------------------------------------
create table student_actions
(id serial primary key, student_id int ,action varchar(10),action_time timestamp default
current_timestamp, description text );

create or replace function log_student_changes()
returns trigger as $$
begin
        if TG_OP = 'INSERT' then
        insert into student_actions(student_id, action, description)
        values(NEW.id,'INSERT','Student added');
        return NEW;
        elsif TG_OP = 'UPDATE' then
    insert into student_actions(student_id, action, description)
    values (NEW.id, 'UPDATE', 'Student updated');
    return NEW;
        elsif TG_OP = 'DELETE' then
    insert into student_actions(student_id, action, description)
    values (OLD.id, 'DELETE', 'Student deleted');
    return OLD;
        end if;

end;
$$ language plpgsql;

create trigger student_action_trg after insert or update or delete on student
for each row
execute function log_student_changes();

insert into student(e_name, email, address, track_id, gender, birth_date)
values ('mina hosam', 'test@iti.com', 'assuit', 1, 'Male', '2000-01-01');

update student set address = 'Alexandria' where e_name = 'mina hosam';

delete from student where e_name = 'mina hosam';

select * from student_actions;
```