دانشگاه صنعتی امیرکبیر

(پلی تکنیک تهران)

تمرین پنجم

آزمایشگاه پایگاه داده

گروه اول

عطیه براتی نیا ۹۶۳۱۰۱۰

```
sql ='''

    CREATE or replace FUNCTION public.get_number(IN phone_number character(11))
    returns varchar(100)
    language 'plpgsql'
    as $BODY$
    begin
    if left(phone_number, 2) = '09' then
        return 'mobile phone number';
    elseif left(phone_number, 3) = '021' then
        return concat('city code: ', left(phone_number, 3), ', last 8 digits: ',
                                    substring(phone_number from 4 for 8), ', city: Tehran');
    elseif left(phone_number, 3) = '031' then
        return concat('city code: ', left(phone_number, 3), ', last 8 digits: ',
                                    substring(phone_number from 4 for 8), ', city: Esfahan');
    elseif left(phone_number, 3) = '077' then
        return concat('city code: ', left(phone_number, 3), ', last 8 digits: ',
                                    substring(phone_number from 4 for 8), ', city: Bushehr');
    else
        return 'Invalid phone number';
    end if;
    end;
    $BODY$;

'''
```

```
sql ='''
    DROP FUNCTION films_between_two_dates(date,date);
    CREATE or replace FUNCTION public.films_between_two_dates(IN date1 date, IN date2 date)
    returns table(film_title varchar(100), rental_id integer, customer_id smallint)

    as $BODY$
    begin
        return query select f.title, r.rental_id, r.customer_id from rental r
        inner join inventory i on i.inventory_id=r.inventory_id
        inner join film f on f.film_id=i.film_id
        where rental_date between date1 and date2;
    end;
    $BODY$
     language 'plpgsql';

'''
```

```
sql ='''
    CREATE or replace FUNCTION public.customers_did_not_return(IN date1 date, IN date2 date)
    returns table(rental_id integer, first_name varchar(100), last_name varchar(100), rental_date timestamp
    ,email varchar(100), address_id smallint)

    as $BODY$
    begin
        return query select r.rental_id, c.first_name, c.last_name, r.rental_date
        , c.email, c.address_id
        from
        rental r
        left join customer c on r.customer_id=c.customer_id
        where r.return_date is null and
        exists(select f.rental_id from public.films_between_two_dates(date1, date2) f where f.rental_id=r.rental_id);
    end;
    $BODY$
     language 'plpgsql';
'''
```

```
sql ='''
    CREATE OR REPLACE FUNCTION public.check_valid_rental()
        RETURNS TRIGGER
        language 'plpgsql'
    as $BODY$
    BEGIN

        IF NEW.return_date::DATE - NEW.rental_date::DATE > (select f.rental_duration
        from film f right join inventory i on i.film_id=f.film_id
        where NEW.inventory_id = i.inventory_id) then
            INSERT INTO public.log values(
                    NEW.customer_id
                    ,EXTRACT(epoch FROM (NEW.return_date::DATE - NEW.rental_date))/86400);

        end if;

        RETURN NEW;
    END;
    $BODY$;

    CREATE or replace TRIGGER check_return_data
      AFTER INSERT OR UPDATE
      ON rental
      FOR EACH ROW
      EXECUTE PROCEDURE public.check_valid_rental();


'''
```

```
sql ='''
    CREATE OR REPLACE PROCEDURE increase_rent(
        amount int,
        value2 int
    ) LANGUAGE plpgsql as $BODY$
    begin

    update film set rental_duration = rental_duration + amount;

    end;
    $BODY$


'''
```