

Exercise of Function

Exercise : Car Number:

Providers: ZHU Yueming, ZHAO Yi

Description:

You need to finish one of following two task.

Task 1. submit any screenshot of learning this week.

Task 2. finish the question below and submit a sql file

Design a function named `get_city(car_num varchar)` that can return the city according to the license plate number of a car.

In your function, you should verify that the license plate number started by a valid license Vehicle registration institution code (for example: '粤A') and then have **5** or **6** characters (for example: '88888' or 'D88888')

In this exercise, we only need to consider following seven cities in Guang Dong province, which means for other input of registration institution code can be regarded as invalid input.

city	code
GUANG ZHOU	粤A
SHEN ZHEN	粤B
ZHU HAI	粤C
SHAN TOU	粤D
FO SHAN	粤E
SHAO GUAN	粤F
ZHAN JIANG	粤G

If there are 5 characters after the license Vehicle registration institution code, all the characters should be either capital English letters or numbers in the range of 0~9, inclusive.

If there are 6 characters after the license Vehicle registration institution code, the first character should be either 'D' or 'F', and other characters should be either capital English letters or numbers in the range of 0~9, inclusive.

If the license plate number is valid return the corresponding city, otherwise throw exceptions "Invalid Province", "Invalid car number!", "Invalid plate length", "Invalid city" respectively.

Sample input and output

parameter	exception or result
京A11111	Invalid Province
粤A111111	Invalid plate Length
粤A1111	Invalid plate Length
粤H1111	Invalid plate Length
粤GF11111	ZHAN JIANG
粤AD11111	GUANG ZHOU
粤AR11111	Invalid plate Length
粤ARRRRR	GUANG ZHOU
粤Arrrrr	Invalid car number

Sample solutions:

Sample Solution 1:

```
create function get_city(car_num varchar)
    returns varchar
    language plpgsql
as
$$
DECLARE
    cn_len      INT;
    i           INT      := 3;
    city_number INT;
    ascii       INT;
    city_names  varchar(80) := 'GUANG ZHOU,SHEN ZHEN,ZHU HAI,SHAN TOU,FO
SHAN,SHAO GUAN,ZHAN JIANG';
BEGIN
    -- check length
    if (substring(car_num, 1, 1) <> '粤')
    then
        raise exception 'Invalid Province';
    end if;

    cn_len = char_length(car_num);
    if not
        cn_len between 7 and 8
    then
        raise exception 'Invalid plate Length';
```

```

elseif
    cn_len = 8 and substring(car_num, 3, 1) not in ('F', 'D')
then
    raise exception 'Invalid plate Length';
end if;
-- check car number
while i <= cn_len
    LOOP
        ascii = ascii(substring(car_num, i, 1));
        if not (ascii >= 48 and ascii <= 57 or ascii <= 90 and ascii >= 65)
        then
            raise exception 'Invalid car number';
        END IF;
        i = i + 1;
    END LOOP;
-- check city info
city_number = ascii(substring(car_num, 2, 1));
if city_number not between 65 and 71 then
    raise exception 'Invalid city';
else
    return split_part(city_names, ',', city_number - 64);
end if;
END;
$$;

```

Sample Solution2:

```

create function get_city(car_num varchar)
returns varchar
language plpgsql
as
$$
DECLARE
    cn_len      INT;
    city_number INT;
    city_names  varchar(80) := 'GUANG ZHOU,SHEN ZHEN,ZHU HAI,SHAN TOU,FO
SHAN,SHAO GUAN,ZHAN JIANG';
BEGIN
    -- check length
    if (substring(car_num, 1, 1) <> '粤')
    then
        raise exception 'Invalid Province';
    end if;

    cn_len = char_length(car_num);
    if not
        cn_len between 7 and 8

```

```
then
    raise exception 'Invalid plate Length';
elseif
    cn_len = 8 and substring(car_num, 3, 1) not in ('F', 'D')
then
    raise exception 'Invalid plate Length';
end if;
-- check car number
if not substring(car_num, cn_len - 4, 5) ~ '^[A-Z0-9]+$'
then
    raise exception 'Invalid car number';
end if;
city_number = ascii(substring(car_num, 2, 1));
if city_number not between 65 and 71 then
    raise exception 'Invalid city';
else
    return split_part(city_names, ',', city_number - 64);
end if;
END;
$;$
```