# ORACLE

## SQL

### 解锁系统用户（Sql plus）

system/orclsec //登陆系统用户system密码为创建数据库时设置的密码

alter user scott account unlock; //scott用户默认密码为tiger

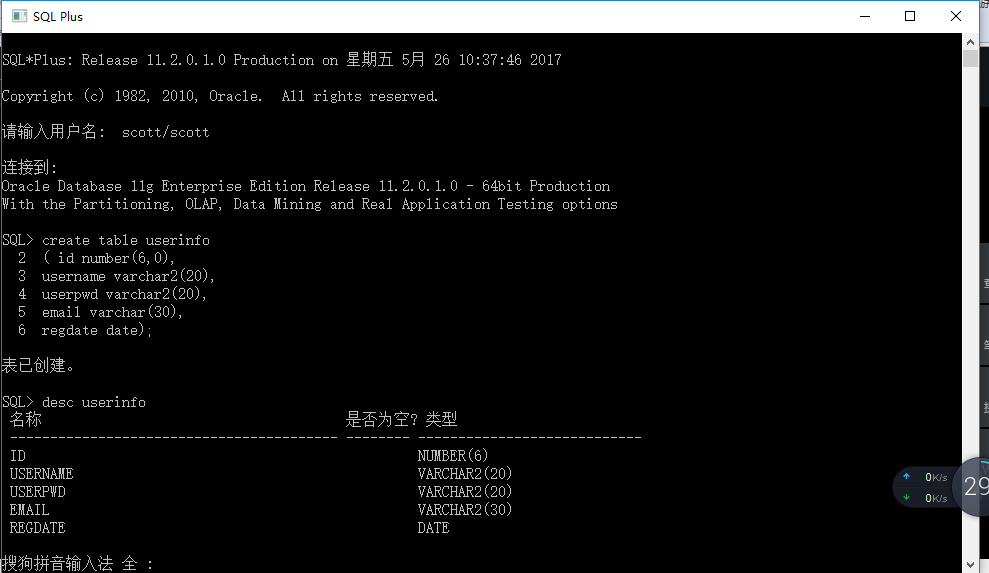
connect scott/scott //链接到scott用户

show user //查看该用户

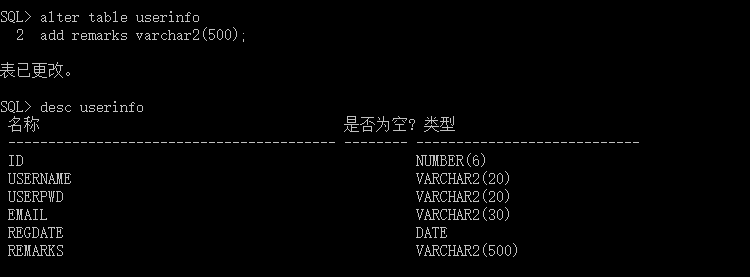
### 表空间

表空间:逻辑存储空间  
1)永久表空间: 永久存储的表/视图等  
2)临时表空间: 数据库操作中中间执行的过程, 完了自行删除将不永久保存  
3)UNDO表空间: 保存事务所修改的旧值,修改前的数据(利于撤销等方便回滚)

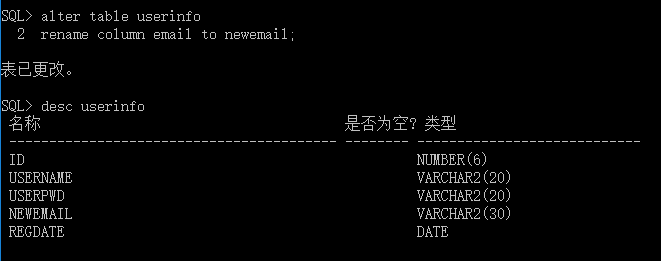
### 在Sql plus中创建表



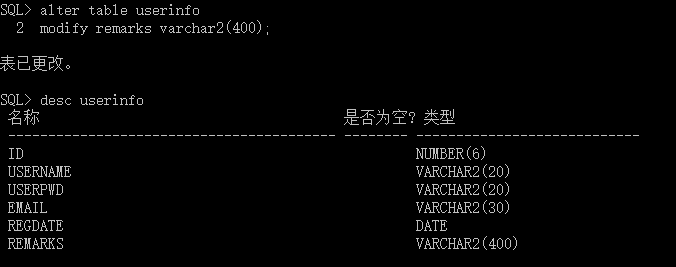
### 添加列



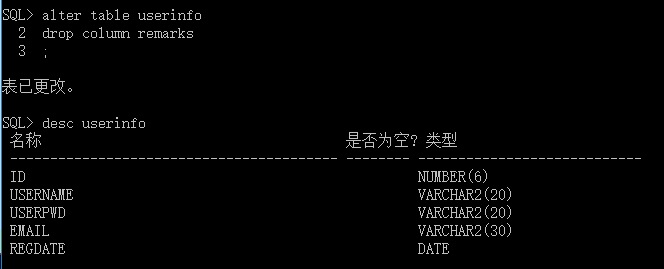
### 更改列名



### 更改列数据类型（表中没有数据时操作）



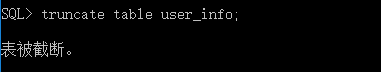
### 删除列



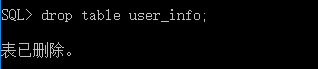
### 更改表名



### 清空表数据

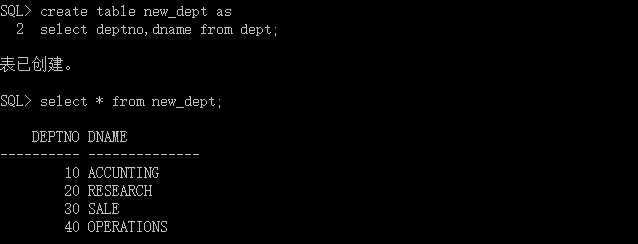


### 删除表

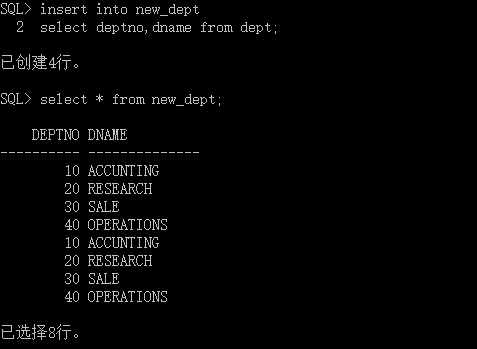


## SQL操作表数据

### 复制表数据（创建表时 选取列复制 若复制所有select \*）



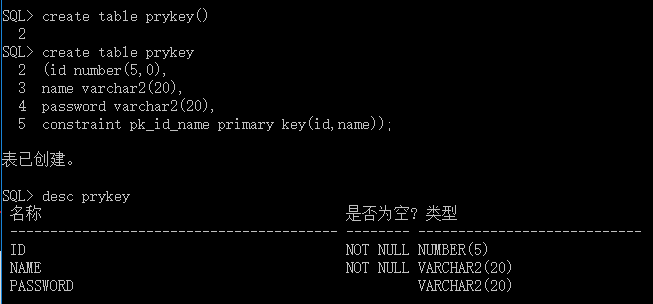
### 添加表数据（sqlplus中操作表数据后要执行 commit;）



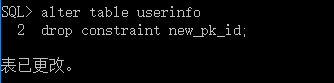
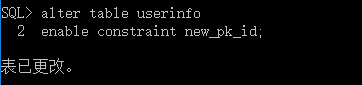
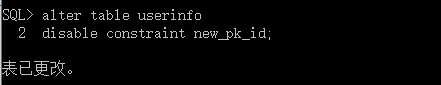


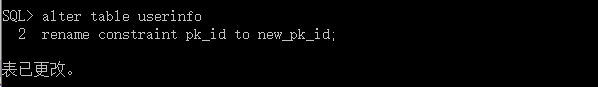
## SQL约束

### 联合主键约束

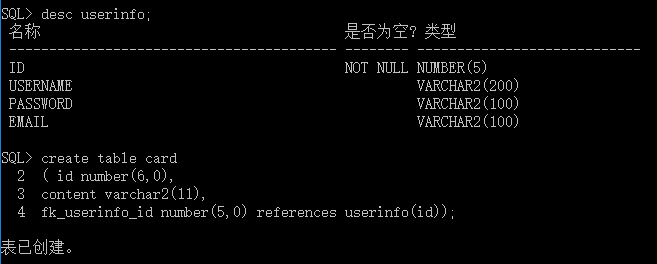


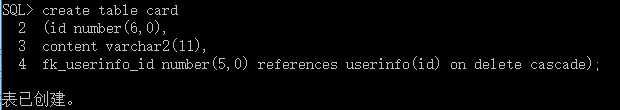
### 禁用、重启、删除、修改约束



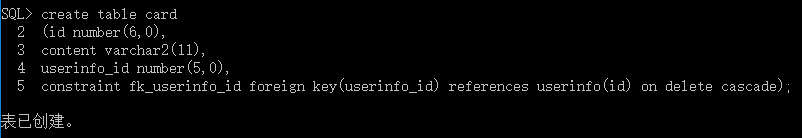


### 创建表时添加外键约束（1、列级添加）

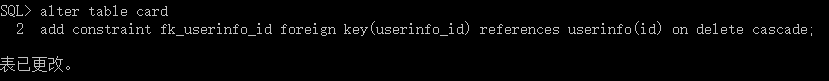
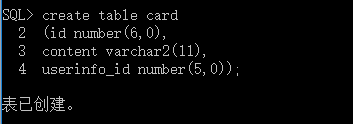




### 创建表时添加外键约束（2、表级添加）



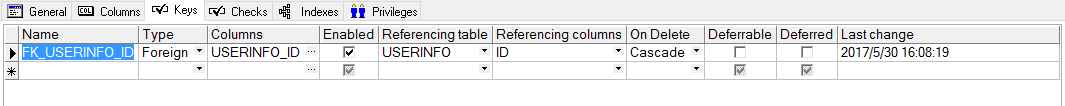
### 17.修改表时添加外键约束



### 两种创建约束的区别

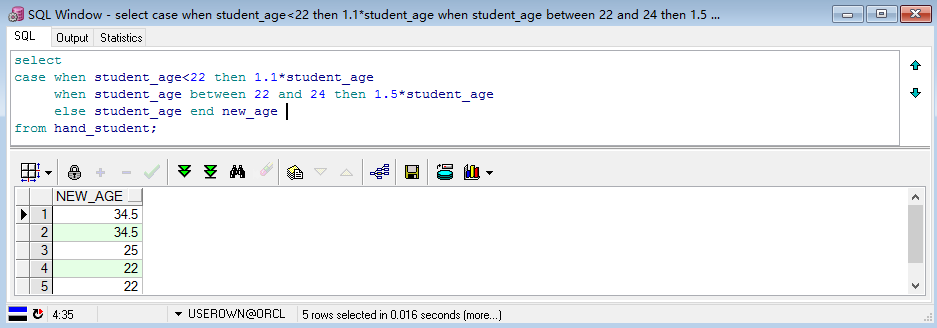
采用（2）时 就是为name取了一个别名

采用（1）时 name为系统默认名



## SQL查询

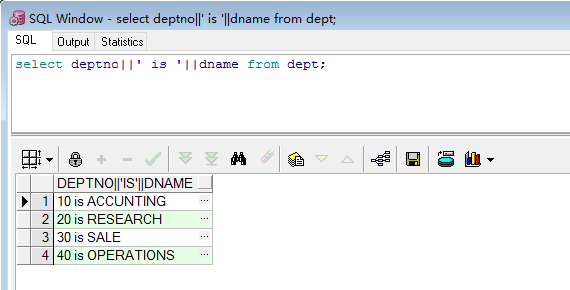
### Case..when



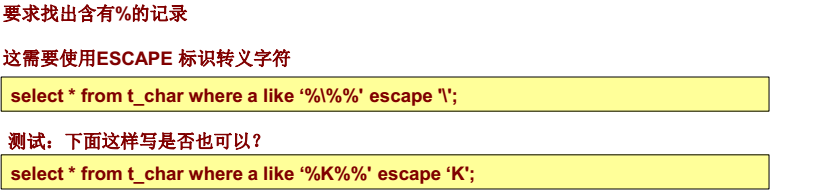
### Decode



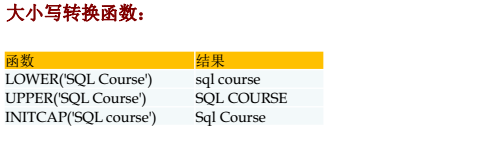
### 拼接（||）



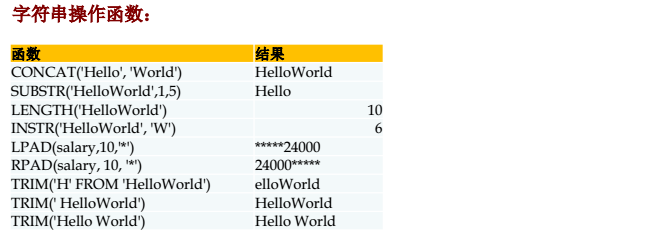
### 查询含有通配符的字段



### 大小写类型转换



### 字符串处理



### 数字处理

如下所示：

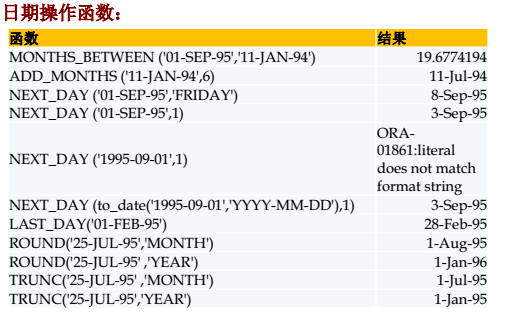
Round（x,y），数字型表达式,如果y不为整数则截取y整数部分，如果y>0则四舍五入为y位小数，如果y小于0则四舍五入到小数点向左第y位

Trunc（x,y），返回截取后的值，用法同round(x[,y]),只是不四舍五入

MOD（x,y），取余数

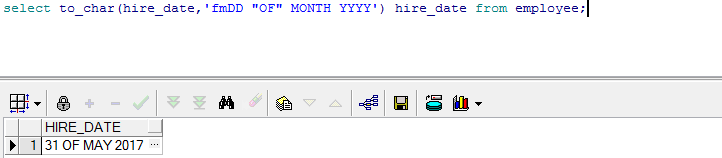


### 日期处理

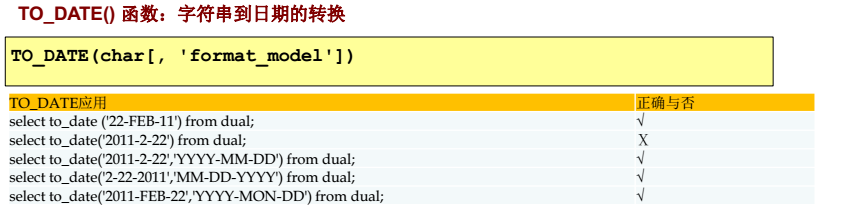


### 日期到字符串转换（to\_char）





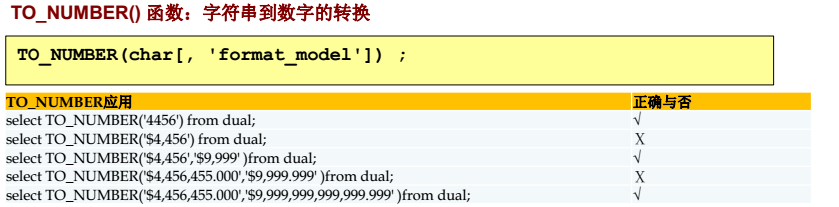
### 字符串到日期的转换



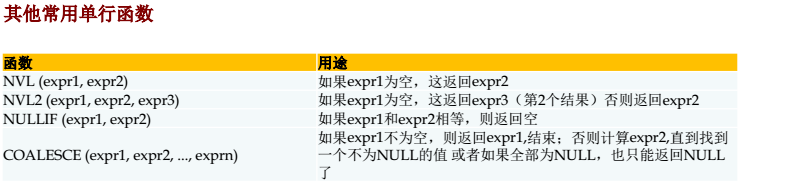
### 数字到字符串转换（to\_char）



### 字符串到日期的转换

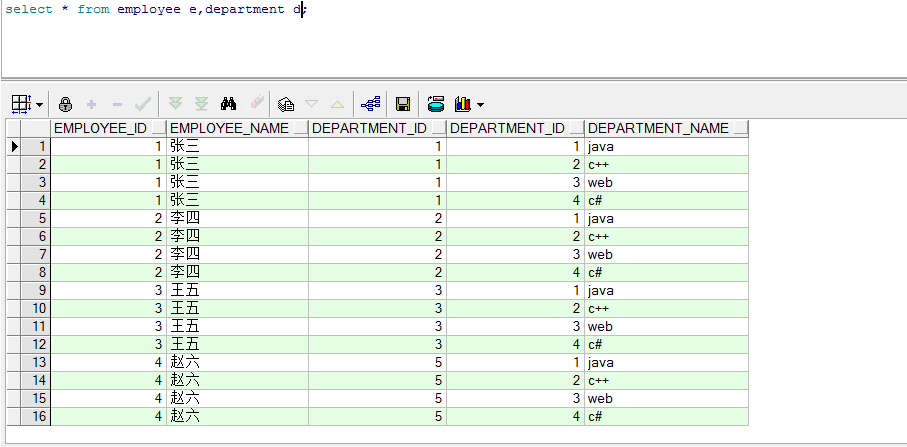


### 其他常用单行函数



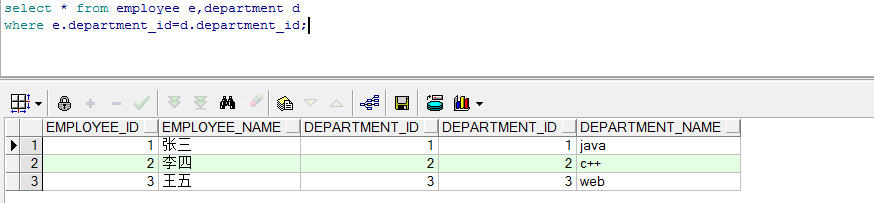
## SQL多表查询

### 笛卡尔积

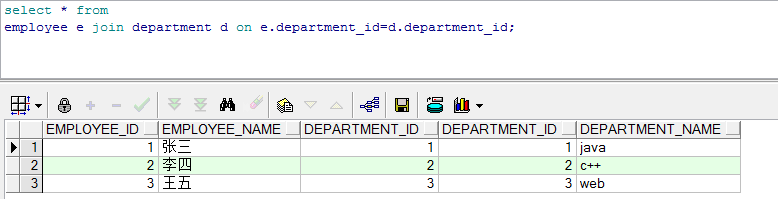


### 等值连接

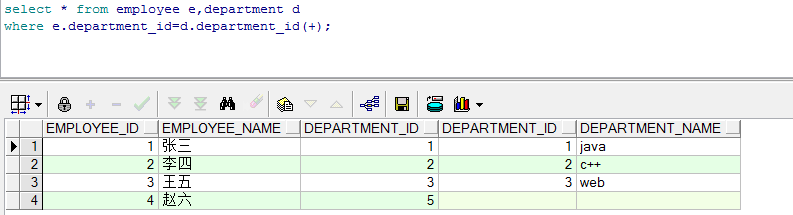
（1）



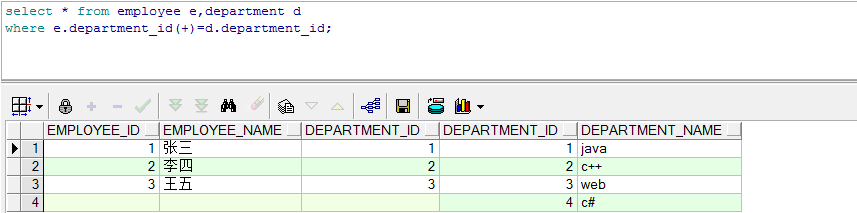
（2）



### 左外连接（左表全显 右表不全显）

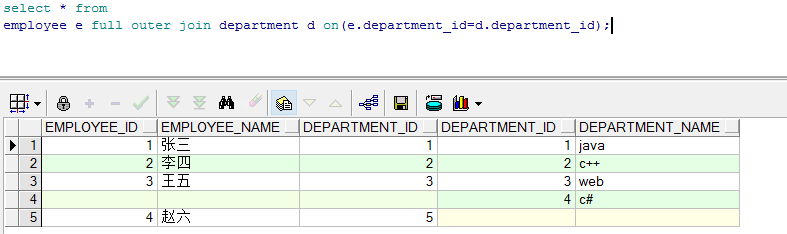


### 右外连接（右表全显 左表不全显）

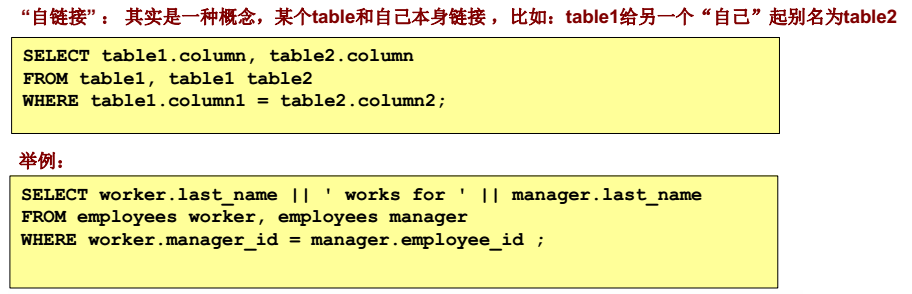


### 全外连接

将full改为left或right可作为左右外链接

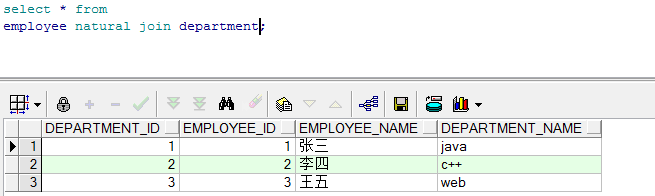
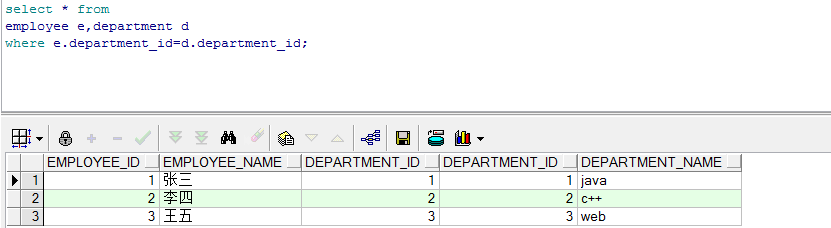


### 自连接



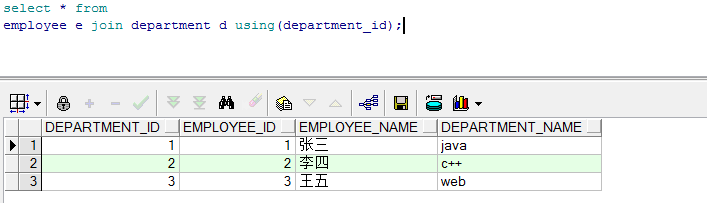
### 自然连接（1）

QQ截图20170531143217

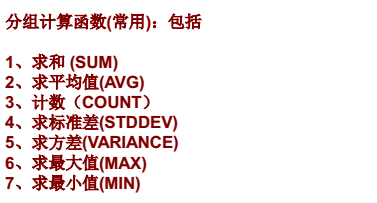


### 自然连接（2）

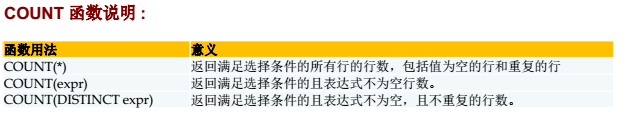


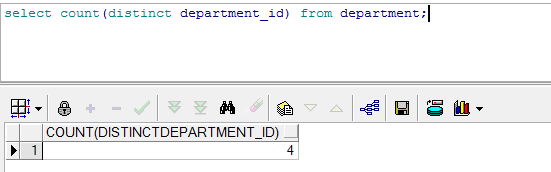
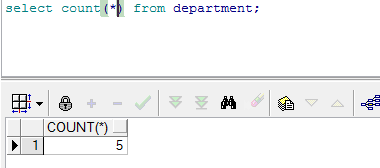


## SQL分组查询

QQ截图20170531150041

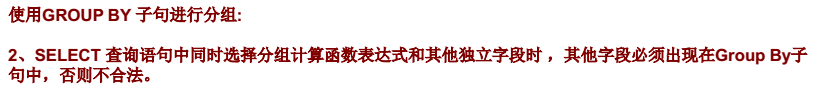
### Count

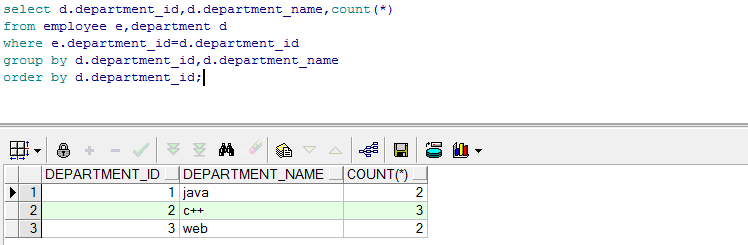




### Group By

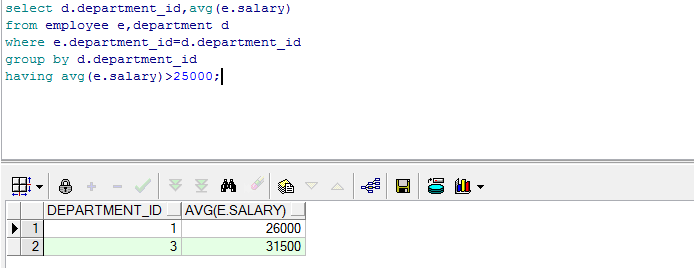
查询每个department有多少个employee 按department分组



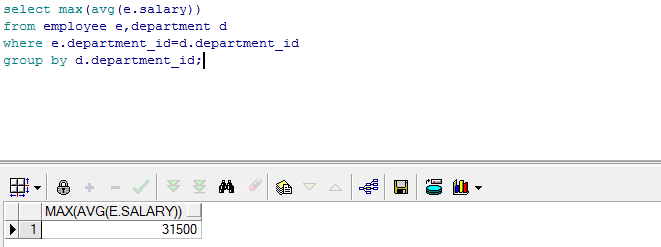


### AVG

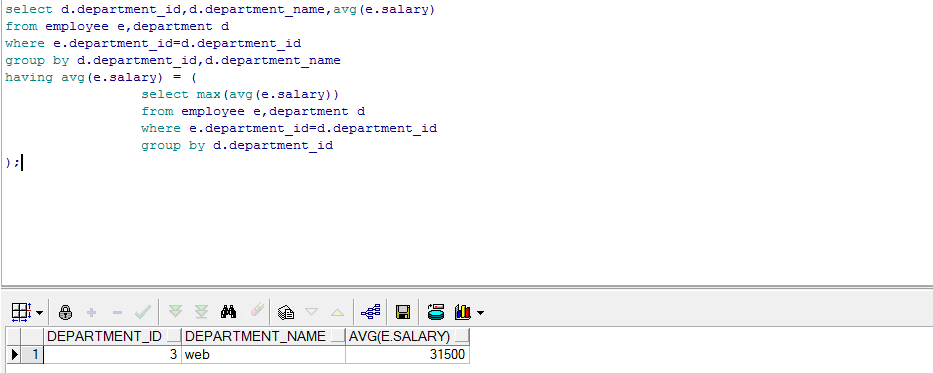
查询平均薪水在25000以上的department



获取最高的department平均薪水

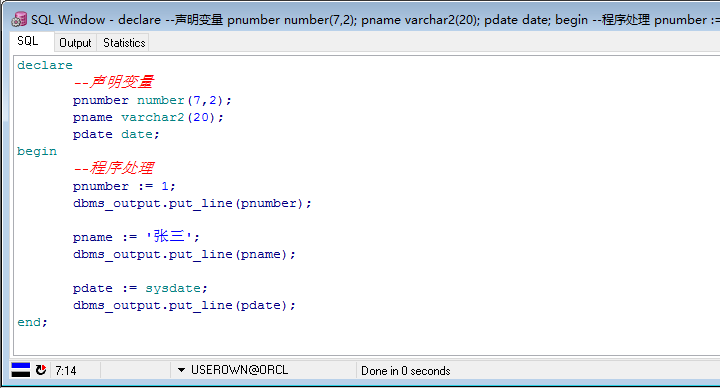


获取拥有最高平均薪水的department

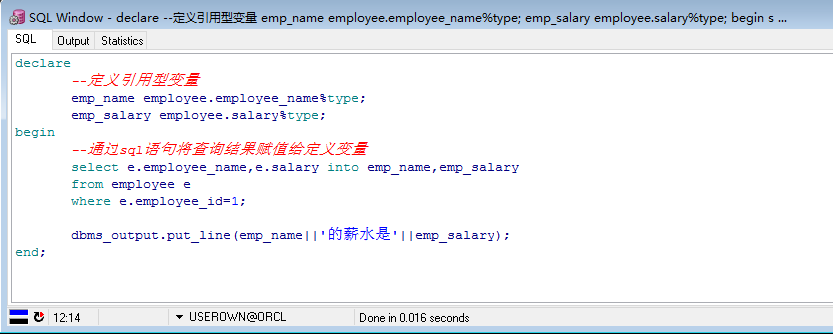


## PL/SQL

### 基本变量类型

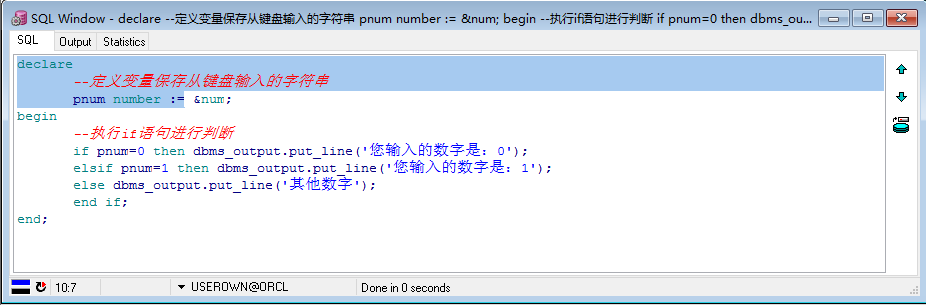


### 引用型变量和记录型变量



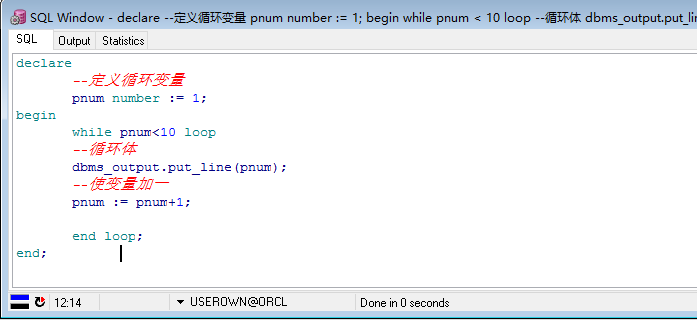


### If语句

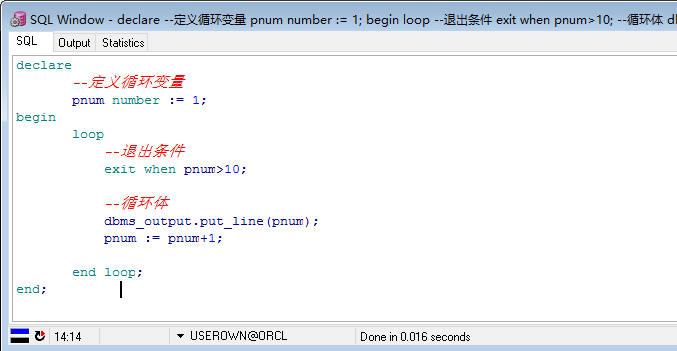


### 循环语句

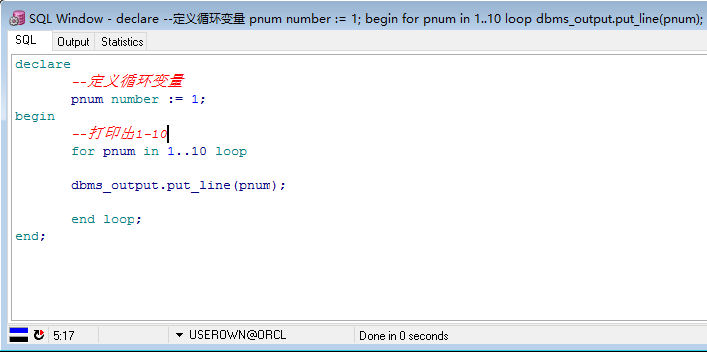
While循环



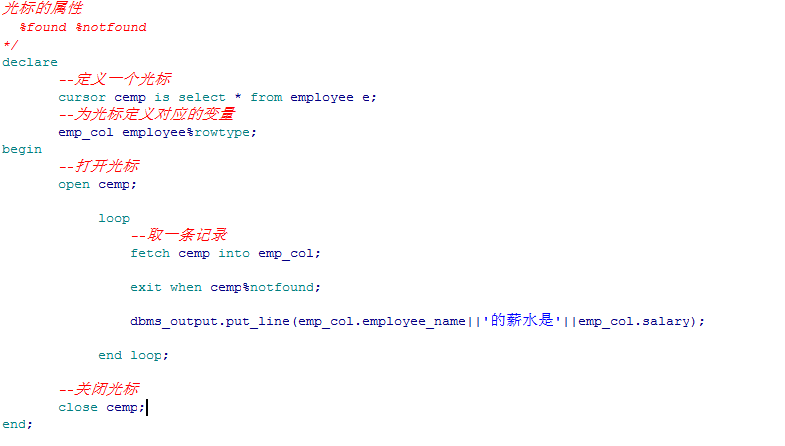
Loop循环(推荐使用)



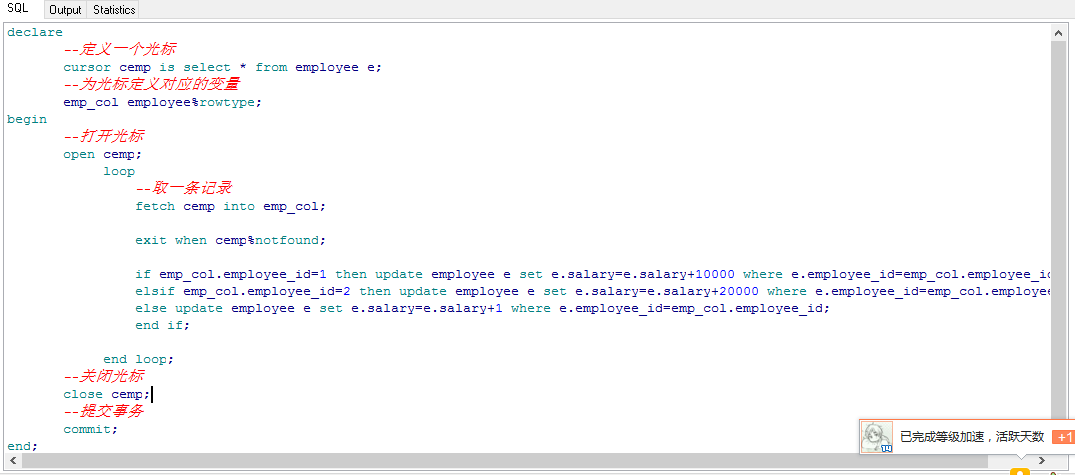
For循环



### 光标（就是一个结果集）

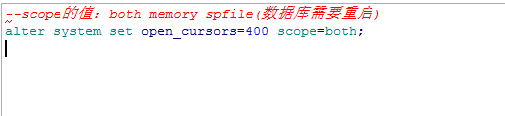


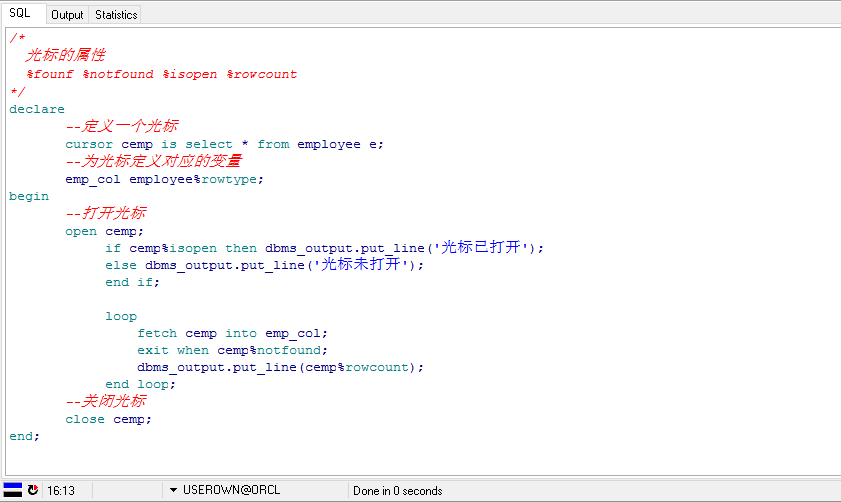
### 光标的实例（给指定员工涨工资）



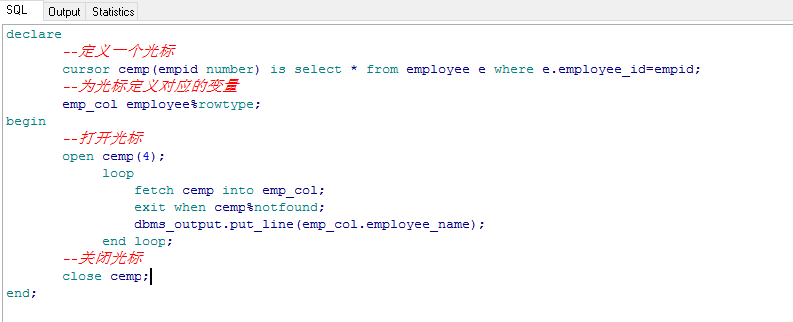
### 光标的属性

在同一个会话中 光标打开上线300个



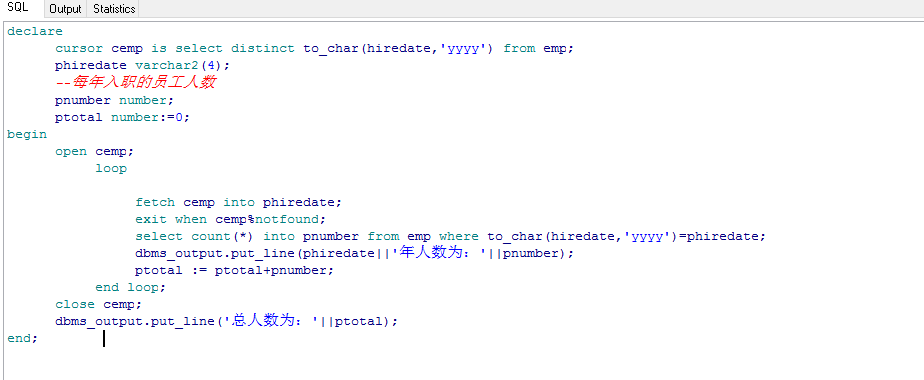


### 光标带参

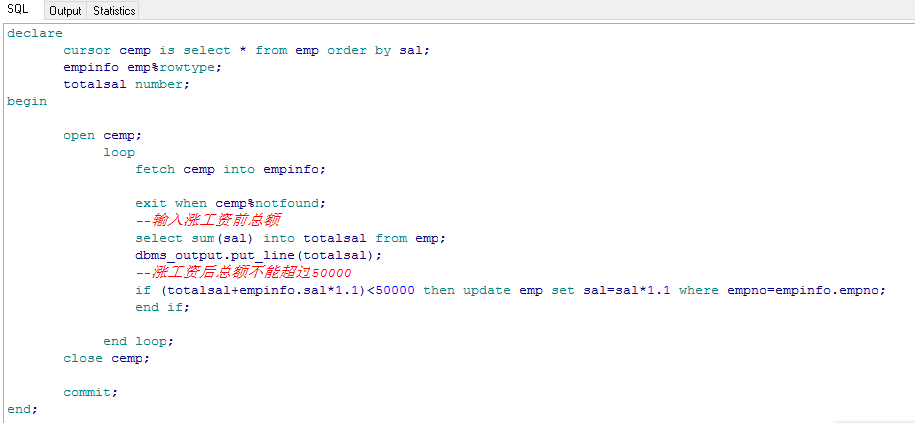


### 案例集锦

每年入职的员工数

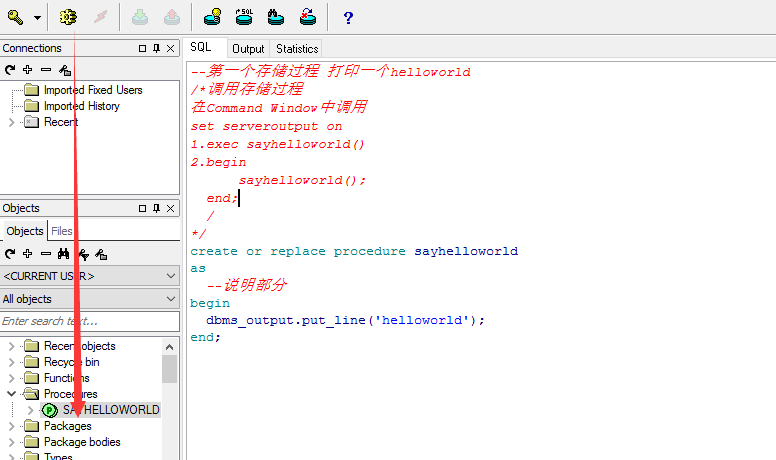


按最低薪水开始给员工涨工资10%，涨后总工资不能超过50000

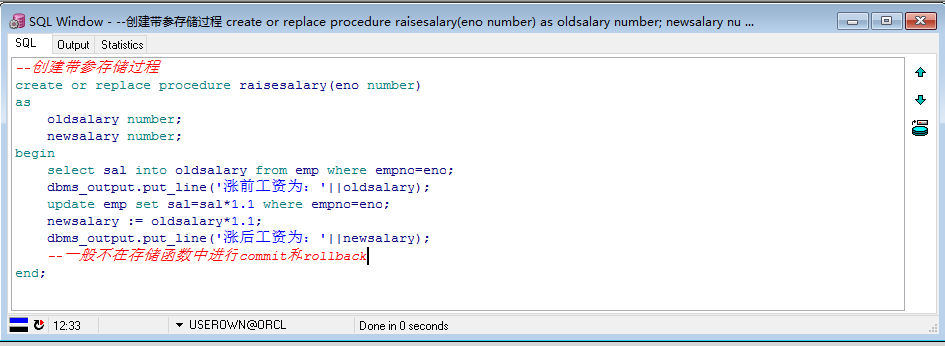


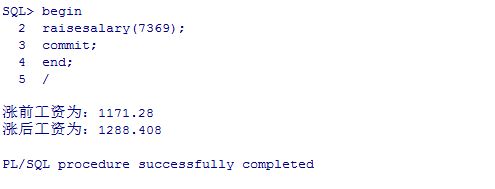
### 存储过程和自定义函数

基本创建

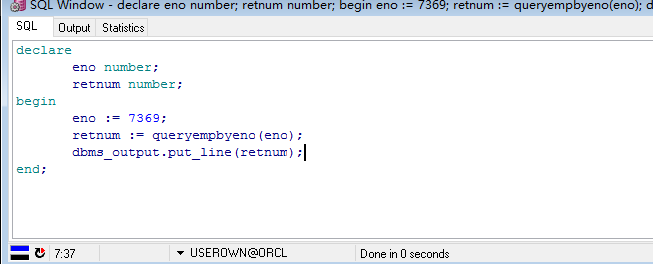
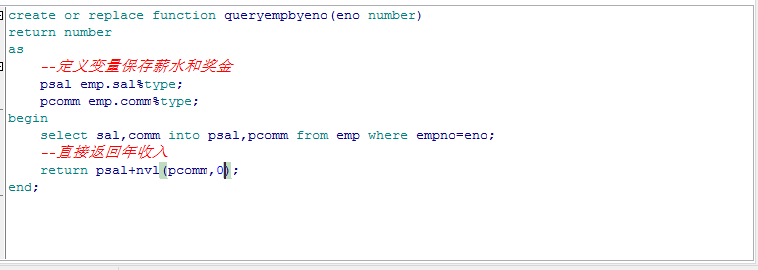


创建带参存储过程

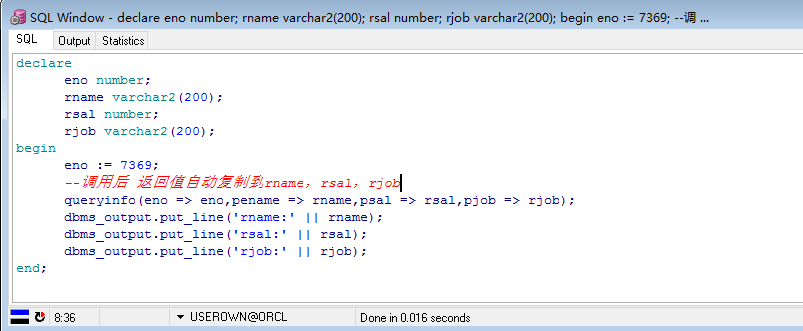
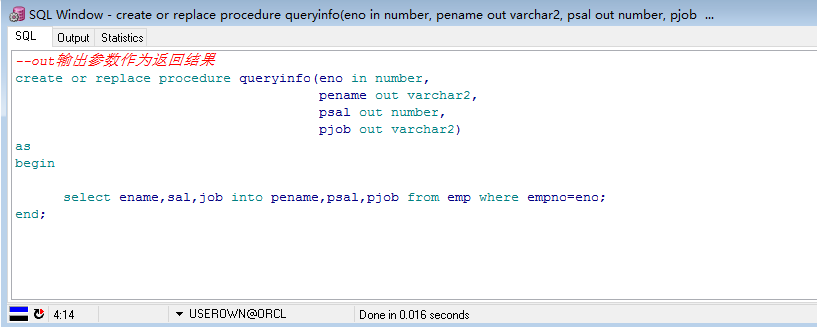




### 存储函数（返回一个值使用）

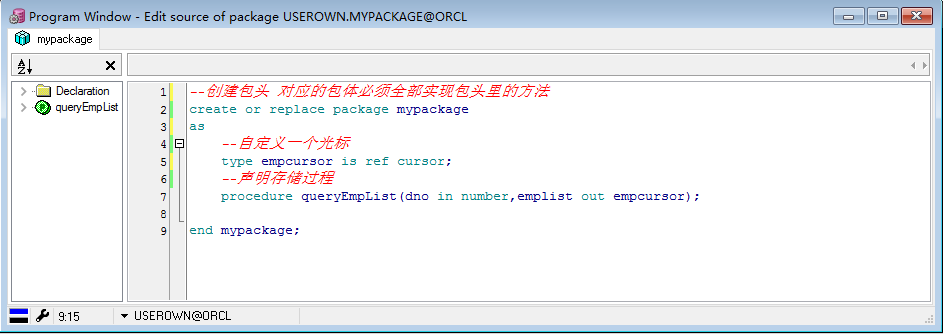


### Out参数（返回多个值使用）

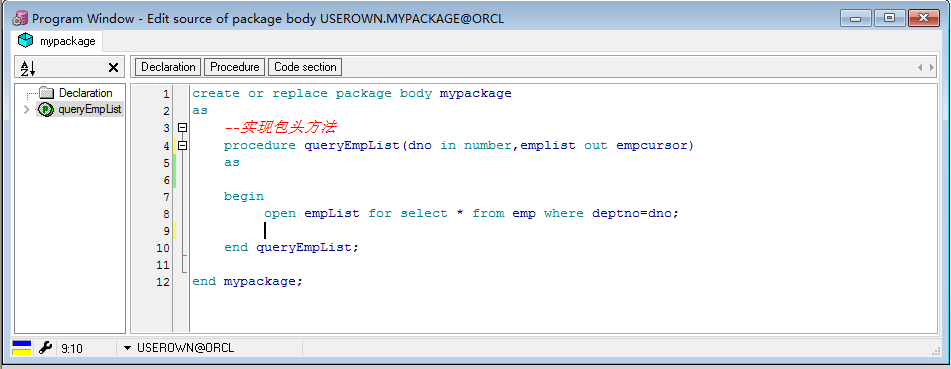


### Out参数（返回集合体）

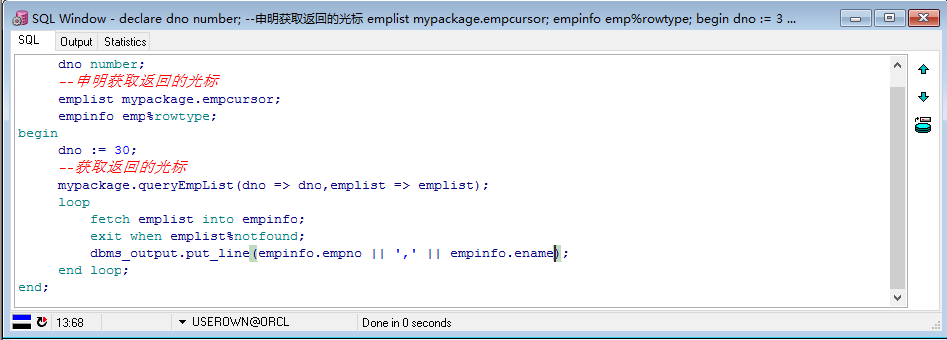
创建包头



创建包体



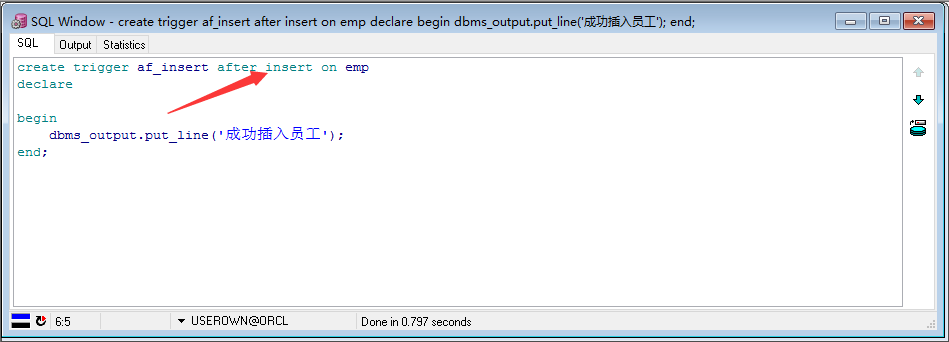
调用包内存储过程



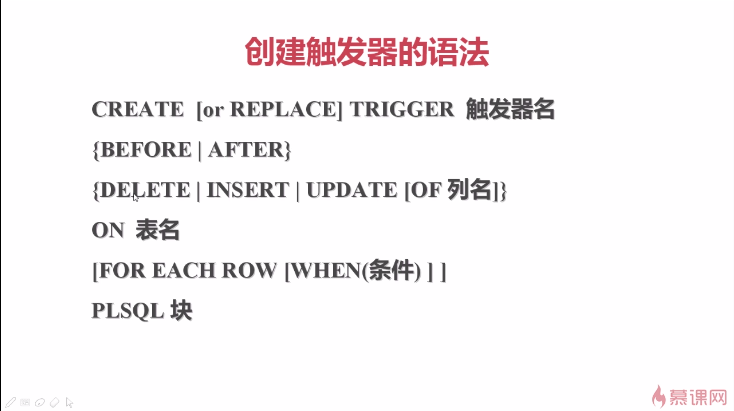
### 触发器

触发条件根据红色箭头处对应修改

此图为语句级触发器（针对表）



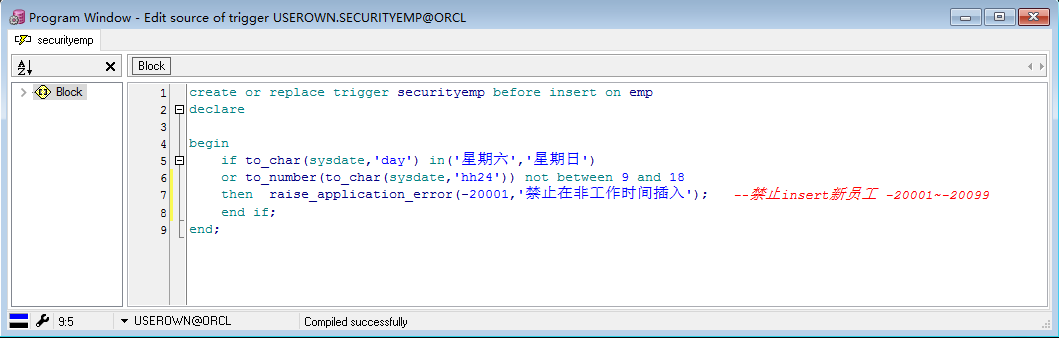
此图为定义行级触发器（针对行）



### 触发器的应用

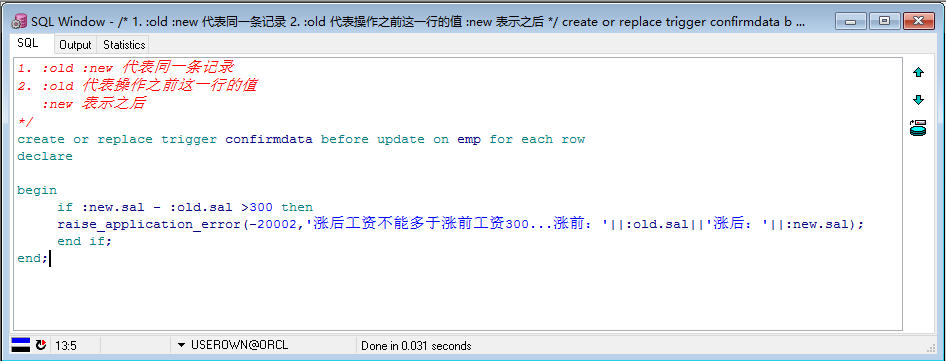
复杂的安全性检查

禁止在非工作时间插入员工



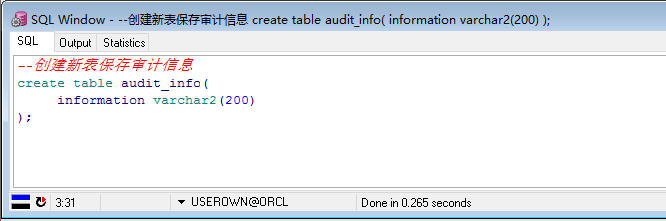
数据的确认

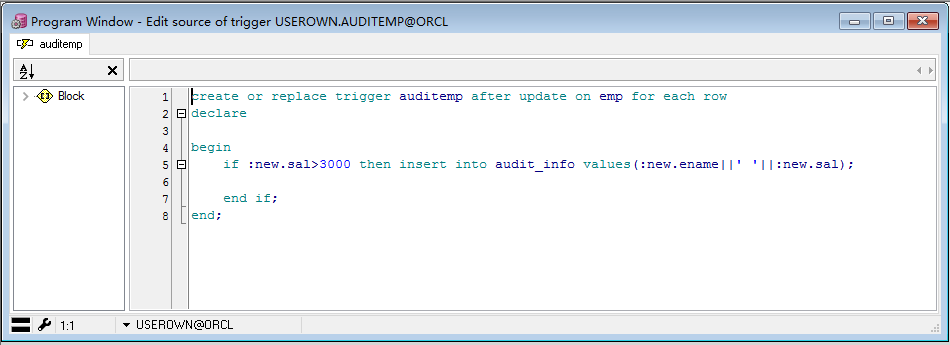
涨后工资不能大于涨前工资300



数据库审计

当涨后工资超过3000时 审计员工信息





数据的备份和同步

