**1、本章面试题**

     oracle表空间是什么，如何进行管理

oracle用户如何创建，如何进行管理

oracle常用函数有哪些，如何使用

**2、知识点**

**2.1、课程回顾**

oracle主要用户sys system scott

oracle数据类型 字符类型 数值类型 时间类型 大对象类型

DDL(create alter drop) DML(insert(union union all) update delete) DQL(select)

序列 （弥补oracle不可以主键自增而专门创建对象）

伪列（rownum 用于分页）pageNo第几页 pageSize每页显示数量

开始值：(pageNo-1)\*pageSize 结束值：pageNo\*pageSize+1

**2.2、本章重点**

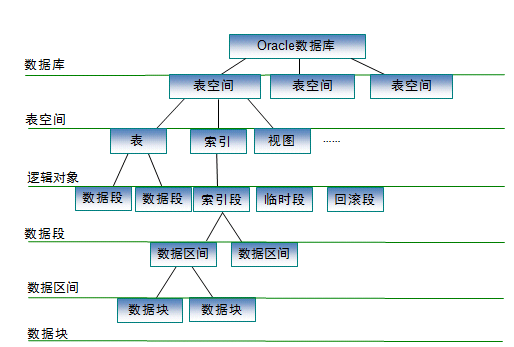
oracle表空间管理

用户创建和授权

常用函数

**3、具体内容**

**3.1 oracle 存储的体系结构**



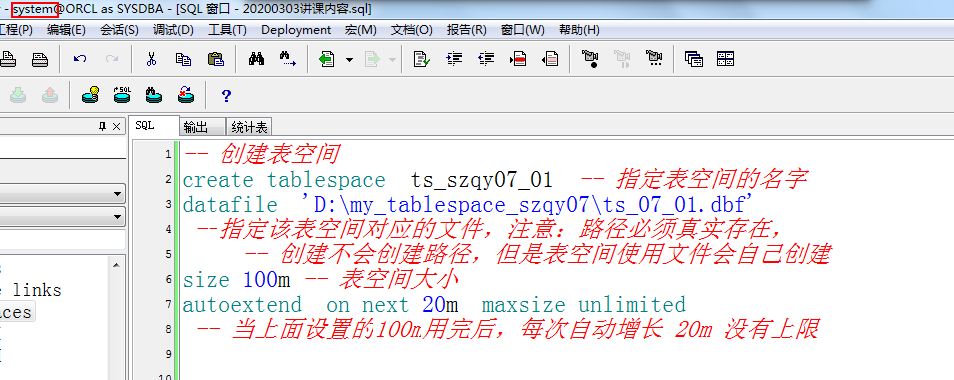
表空间：tablespace 系统表空间 临时表空间 等

**3.2 表空间的管理**

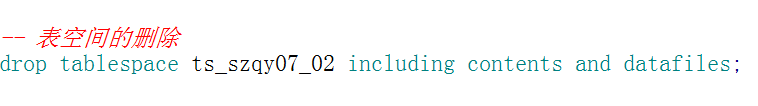
**概念：**

表空间：是数据库的逻辑划分，任何数据库对象都必须存储在某一个表空间（scott用户默认使用的表空间就是users,所以使用scott用户创建的表默认都users表空间）。表空间由一个或者多个磁盘文件构成，每个数据库至少有一个默认的表空间。表空间总大小等于所有从属于该表空间的文件大小的总和。

**创建：**



**删除：**



**3.3 用户管理**

**oracle常用的三个角色**

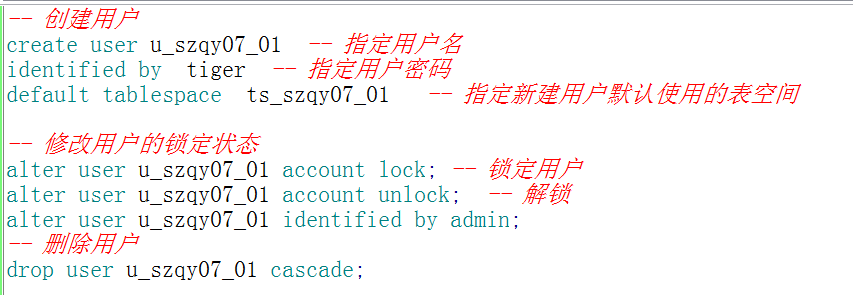
**connect** 普通的用户角色，仅可以具有创建session的权限，可以使用该用户进行登录。不具备DDL操作权限

**resource** 比connect权限大很多 具有大部分的DDL操作权限，创建索引，序列，表，触发器，存储过程，函数等等。

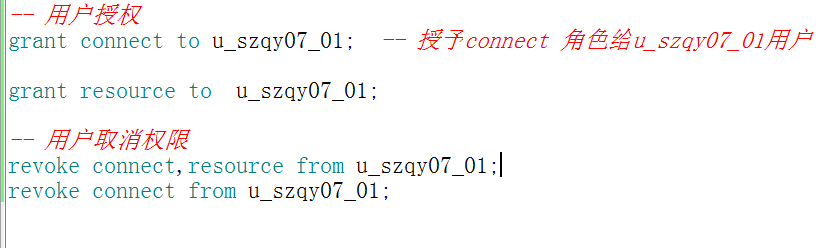
**dba** 拥有所有权限，该角色在实际使用过程一定要谨慎使用。

**oracle 用户的创建更新和删除**

仍然需要使用system或者是sys用户。



**用户的授权管理**



**3.4 函数**

**3.4.1 字符串函数**

-- **concat** 字符串连接函数 与mysql中使用不同 在oracle中只能两个参数

-- mysql 语法：select concat('hello',',','szqy07',' 啊啊');

select concat('hello','szqy07') from dual;

select concat(concat('hello',','),'szqy07') from dual;

-- || 连接字符串

select 'hello'||','||'szqy07' from dual;

-- 实际语句使用

select t.job,t.ename, t.job||'\_'||t.ename from emp t;

select t.job,t.ename, concat(concat(t.job,'\_'),t.ename) from emp t;

select t.job,t.ename, concat(t.job||'\_',t.ename) from emp t;

-- initcap 把单子的首字母变为大写

select initcap('software is like sex, it is better when it`s free') from dual;

select initcap('SOFTWARE IS LIKE SEX, IT IS BETTER WHEN IT`S FREE') from dual;

-- 实际使用

select t.job,t.ename, initcap(t.job||'\_'||t.ename) from emp t;

-- **upper lower**  所有字母变大写/小写

select upper('software is like sex, it is better when it`s free') from dual;

select lower('SOFTWARE IS LIKE SEX, IT IS BETTER WHEN IT`S FREE') from dual;

select lower('SOFTWARE IS LIKE SEX, it IS BETTER WHEN IT`S FREE') from dual;

-- **instr**  相当于java中String的indexOf

select instr('hello','e') from dual; -- 在hello 中找到第一次出现e的位置

select instr('helloe','e',3) from dual; -- 在helloe中从第3位置开始查找第一次出现e的位置

select instr('helloeaeae','e',3,2) from dual; -- 在helloe中从第3位置开始查找第2次出现e的位置

select instr('helloeaeae','e',3,3) from dual;

-- **substr** 与java中string 的subString(start,end)不同 与js中 substr 是一样的 截取字符串功能

select substr('hello AAA',2) from dual; -- 从第2个位置开始截取到字符串末尾

select substr('hello AAA',2,2) from dual; -- 从第2个位置开始截取，截取2个长度的字符串

select substr('hello AAA',2,4) from dual;

-- 使用上面两个方法的结合，截取以下字符串中的汉字，字符串有可能随时变化

'a2df1113111d,a1223f21111ds,为武汉加油!!!!共抗疫情!!,fdasfa'

-- 思路

-- 找到第2个逗号的位置 开始截取位置

select instr('a2df1113111d,a1223f21111ds,为武汉加油!!!!共抗疫情!!,fdasfa',',',1,2) from dual;

-- 找到第3个逗号的位置

select instr('a2df1113111d,a1223f21111ds,为武汉加油!!!!共抗疫情!!,fdasfa',',',1,3) from dual;

-- 结算截取长度 第3个逗号位置-第2个逗号位置

select instr('a2df1113111d,a1223f21111ds,为武汉加油!!!!共抗疫情!!,fdasfa',',',1,3)-

instr('a2df1113111d,a1223f21111ds,为武汉加油!!!!共抗疫情!!,fdasfa',',',1,2) from dual;

-- 截取

select substr('a2df1113111d,a1223f21111ds,为武汉加油!!!!共抗疫情!!,fdasfa',

instr('a2df1113111d,a1223f21111ds,为武汉加油!!!!共抗疫情!!,fdasfa',',',1,2)+1,

instr('a2df1113111d,a1223f21111ds,为武汉加油!!!!共抗疫情!!,fdasfa',',',1,3)-

instr('a2df1113111d,a1223f21111ds,为武汉加油!!!!共抗疫情!!,fdasfa',',',1,2)-1) from dual;

-- 字符串变为'fadfdasf,ffadf,为武汉加油!!!!!共抗疫情!!!,fadsfa'

select substr('fadfdasf,ffadf,为武汉加油!!!!!共抗疫情!!!,fadsfa',

instr('fadfdasf,ffadf,为武汉加油!!!!!共抗疫情!!!,fadsfa',',',1,2)+1,

instr('fadfdasf,ffadf,为武汉加油!!!!!共抗疫情!!!,fadsfa',',',1,3)-

instr('fadfdasf,ffadf,为武汉加油!!!!!共抗疫情!!!,fadsfa',',',1,2)-1) from dual;

-- **rpad lpad** r =right l=left

select rpad('hello',10,'@') from dual; -- 把hello的右边用@补全到10位

select rpad('hello',2,'@') from dual; -- 无意义的用法

select lpad('hello',10,'@') from dual;-- 把hello的左边用@补全到10位

-- 项目中真正使用 假如说有一个订单表，添加数据时，想让你生成 有序 定长 的订单编号

-- order202003030000001 order202003030000002 order20200303000003...... 明天 order20200304000004....

-- 创建序列从1开始 增量为1

create sequence seq\_public\_id;

-- 使用lpad补全订单的后7位

select lpad(seq\_public\_id.nextval,7,0) from dual;

-- 格式化当前日期to\_char(日期,格式)

-- sysdate 获取数据库所在服务器的当前时间

select to\_char(sysdate,'yyyymmdd') from dual;

select to\_char(sysdate,'yyyy-mm-dd') from dual;

-- 最终生成订单编号为：

select 'order'||to\_char(sysdate,'yyyymmdd')||lpad(seq\_public\_id.nextval,7,0) from dual;

-- **ltrim rtrim**  trim js中，去除字符串左右空格 ' abc '.trim() ='abc'

select ' hello',ltrim(' hello') from dual;-- 去除hello左侧的空格

select 'hello ',rtrim('hello ') from dual; -- 去除hello右侧的空格

-- length 计算字符串长度

select length('hello '),length(rtrim('hello ')) from dual;

-- **trim (from leading trailing both)** 不同方式的去除字符串左右两侧字符串的函数

select trim('a' from 'aaahelloaaa') from dual; -- 从字符串aaahelloaaa左右去除a

select trim('a' from 'baaahelloaaa') from dual;

select trim(leading 'a' from 'aaahelloaaa') from dual; -- 从字符串aaahelloaaa左侧去除a

select trim(trailing 'a' from 'aaahelloaaa') from dual; -- 从字符串aaahelloaaa右侧去除a

select trim(both 'a' from 'aaahelloaaa') from dual; -- 同不加一个意思

**3.4.2 日期相关函数**

-- **sysdate**

select sysdate from dual;

select \* from tb\_class;

-- 为已经创建的表添加字段

alter table tb\_class add addtime date;

-- 更新所有数据的添加日期为当前时间

update tb\_class set addtime =sysdate;

-- 指定字段添加（要求不为空字段必须写上去）

insert into tb\_class(cid,cname,addtime)

values(seq\_tbclass\_id.nextval,'测试班',sysdate);

-- **add\_months**

select add\_months(sysdate,1) from dual; -- 在当前日期加1月

select add\_months(sysdate,10) from dual;

select add\_months(sysdate,-1) from dual; -- 在当前日期减1月

-- 指定时间 加或减

select add\_months(to\_date('2011-11-11 11:11:11','yyyy-mm-dd hh24:mi:ss'),-12)

from dual;

-- 实际使用 查询emp表中入职超过10年的人

select add\_months(sysdate,-120) from dual;

select \* from emp where hiredate<add\_months(sysdate,-120);

-- **to\_date** 把字符串转换为时间 与mysql不同，mysql语法不严谨，数据是时间类型

-- 插入时直接写字符串就可以，但是oracle不行，时间类型，必须是时间才能插入

select to\_date('2011-11-11 11:11:11','yyyy-mm-dd hh24:mi:ss') from dual;

-- mysql 这样写可以 但oracle不行

insert into tb\_class(cid,cname,addtime)

values(seq\_tbclass\_id.nextval,'测试时间班',

'2011-11-11 11:11:11');

-- oracle必须这样玩

insert into tb\_class(cid,cname,addtime)

values(seq\_tbclass\_id.nextval,'测试时间班',

to\_date('2011-11-11 11:11:11','yyyy-mm-dd hh24:mi:ss'));

-- **last\_day**

select last\_day(sysdate) from dual; -- 显示当前日期的最后一天

select last\_day(add\_months(sysdate,-1)) from dual; -- 显示上个月的最后一天

-- 指定日期的最后一天

select last\_day(to\_date('2011-11-11 11:11:11','yyyy-mm-dd hh24:mi:ss')) from dual;

--假如员工入职第一个月薪水按照基本工资的日平均工资，计算每个员工入职的第1个月，发多少工资

select t.ename,t.sal,t.hiredate from emp t;

-- 计算日平均工资

-- 获取入职日期的最后一天

select hiredate,last\_day(hiredate) from emp;

-- 取出当月的天数

select hiredate,last\_day(hiredate),to\_char(last\_day(hiredate),'dd') from emp;

-- 计算日均工资

select hiredate,last\_day(hiredate),sal,sal/to\_char(last\_day(hiredate),'dd') from emp;

-- 计算该员工上班天数 oracle两个时间差是天数差

select hiredate,last\_day(hiredate),last\_day(hiredate)-hiredate+1 from emp;

-- 计算第一个月工资

select ename,hiredate,last\_day(hiredate),sal/to\_char(last\_day(hiredate),'dd')\*

(last\_day(hiredate)-hiredate+1) from emp;

-- 保留小数点后2位

select ename,sal,hiredate,last\_day(hiredate),round(sal/to\_char(last\_day(hiredate),'dd')\*

(last\_day(hiredate)-hiredate+1),2) 第一月薪酬 from emp;

-- **round** 四舍五入保留数字

select round(11.115,2) from dual; -- 把11.111使用四舍五入保留小数点2位

select round(11.114,2) from dual;

select round(15.114,-1) from dual;

select round(14.114,-1) from dual;

-- to\_char 其中功能之一就是可以把日期转换为字符串

select to\_char(sysdate,'dd') from dual; -- 获取当前日期的天

select to\_char(sysdate,'yyyy-mm') from dual; --获取当前日期的年月

select to\_char(sysdate,'yyyy') from dual; -- 获取年

-- **trunc** 截取时间或者数字

select trunc(11.51) from dual; -- 截取 数字

select sysdate,trunc(sysdate) from dual; -- 去掉时分秒

select trunc(sysdate,'yyyy') from dual; -- 截取到年

select trunc(sysdate,'mm') from dual; -- 截取到月

select trunc(sysdate,'dd') from dual;

select trunc(sysdate,'hh') from dual;

select trunc(sysdate,'mi') from dual;

-- select trunc(sysdate,'ss') from dual; -- 直接错误，因为到秒就不需要截取

-- **to\_number**

select to\_number('11.11') from dual;

select '11.11'+1 from dual;

-- 实际使用

drop table tb\_test\_tonumber;

create table tb\_test\_tonumber(

title varchar2(100),

clicknum varchar2(10) -- 故意写成字符串类型的

);

insert into tb\_test\_tonumber

select '疫情过去了1','1' from dual union

select '疫情过去了2','11' from dual union

select '疫情过去了3','12' from dual union

select '疫情过去了4','121' from dual union

select '疫情过去了5','1112' from dual union

select '疫情过去了6','10' from dual;

-- 按点击量倒序排列 明显错误

select \* from tb\_test\_tonumber order by clicknum desc;

-- 正确写法

select \* from tb\_test\_tonumber order by to\_number(clicknum) desc;

**3.4.3 其他函数**

**-- nvl** 判断是否为null 为null可以使用默认值 相当于mysql中的 ifnull

-- mysql select name,ifnull(name,'名字不详') 名字 from person;

select nvl(null,0) from dual;

-- 实际应用 查询员工表的奖金，如果奖金为空，显示0

select \* from emp;

select ename,comm 原奖金,nvl(comm,0) 显示奖金 from emp;

**4、本章总结**

**4.1 总结本章知识点**

**4.2 面试题答案**

**4.3 预习下一章重点**