Oracle => 神谕 它的第一个客户 => CIA

SQLPLUS = 工具 用来让我们和Oracle数据库交流

SQL => Structured Query Language => 结构化查询语言

DDL DML DQL DCL TCL

DDL => Data Defination Language => 数据定义语言

create alter drop truncate

创建 修改 扔掉 截断

DML => Data Manipulation Language => 数据操纵语言

insert delete update

新增 删除 修改

DQL => Data Query Language => 数据查询语言

select

查询

DCL => Data Control Language => 数据控制语言

grant revoke

授权 撤销授权

TCL => Transaction Control Language => 事务控制语言

commit rollback savepoint

提交 回滚 保存还原点

#:如何锁定一个账户:

alter user scott account lock;

#:如何解锁一个账户:

alter user scott account unlock;

#:如何修改一个账户的密码:

alter user scott identified by tiger;

#:如何切换用户:

conn 用户名/密码

#:如何创建新账户:

create user et1709 identified by et1709;

#:如何给用户授权:

grant dba to et1709;

#:如何撤销授权:

revoke dba from et1709;

#:如何删除用户:

drop user ww;

\*:ed 编辑上一个命令

/ 回车 执行

conn 切换用户

host cls 清屏

===========================================================================================Oracle当中的数据类型:

字符型:

char(10) 固定长度

varchar2(10) 可变长度

\*:long

数值型:

number(7,2) integer float

日期型:

date

timestamp 时间戳（包括毫秒数）

class Student{ create table student(

String name; name varchar(20),

int age; birthday date,

double salary; salary number(7,2)

} );

\* Java当中的类 -> Oracle当中的表

类当中的每个属性 -> Oracle当中的表里的字段

Java当中这个类的每个对象 -> Oracle当中表里的每行数据

#:如何创建一张表:

create table student(

name varchar2(20),

birthday date,

salary number(7,2)

);

#:如何查看已经存在的表结构:

desc student

#:如果建表之后 发现表名写错了 如何修改表名:

alter table student rename to god;

#:如果建表之后 发现少了一个字段 如何添加字段:

alter table student add email varchar2(20);

#:如果建表之后 发现字段名字写错了 如何修改字段名字:

alter table student rename column name to sname;

#:如果建表之后 发现字段类型写错了 如何修改字段:

alter table student modify name varchar2(10);

#:如果建表之后 发现字段多余 需要删除字段:

alter table student drop column email;

上午没有恐龙 => a m 没有 column

#:如何删除一张表:

drop table student purge; purge净化

drop table s1;

drop table s2;

drop table s3;

purge recyclebin;

\*：flashback table student to before drop;

#:如何截断表（只保留表结构 扔掉所有数据）

truncate table student;

========== 我们能够创建表 以及修改表结构和删除表 ==========

DDL : create alter drop truncate

========== 如何操纵数据 ============

#: 如何插入一条数据:

1st way:

insert into student(name,salary,birthday) values('JayZhou',23456.78,sysdate);

2nd way:

insert into student values('GaryZhou',sysdate,12345.67);

#:如何修改数据:

update student set name = 'JayZhou';

update student set salary = salary + 500 where name='WuKong';

update student set salary = 1000,name = '大师兄' where name = '悟空';

#:如何删除数据:

delete from student; => truncate table student;

delete from student where name like '\_\_\_';

% 0到无数个任意字符 \_ 一个任意字符 escape

DML : Data Manipulation Language => 数据操纵语言

INSERT DELETE UPDATE

#: 如何查询表当中的数据

select \* from student; DQL

select \* from student where name like '周\_';

\*:难哭你的关键内容

\*：+ - \* / mod(x,y)

> < >= <= = != <>

between (a) and (b) >a and <b

and or

like

SQL => 结构化查询语言

DDL 数据定义语言 : create alter drop truncate

DML 数据操纵语言 : insert delete update

DQL 数据查询语言 : select

DCL 数据控制语言 : grant revoke

TCL 事务控制语言 : commit rollback savepoint

Oracle当中的系统函数:

多行函数:聚组函数:组函数:聚簇函数

count() sum() max() min() avg()

记录数 求和 求最大 求最小 平均数

1.如何得到一个表当中有多少条记录?

select count(\*) from 表名;

select count(name) from 表名;

\*: 如果字段的值是空 则不记录总数

2.如何统计某个字段的总和?

select sum(salary) from 表名;

3.如何求出最大值?

select max(salary) from student;

4.如何求出最小值?

select min(salary) from student;

5.如何求出平均数?

select avg(salary) from student;

单行函数: x.method(y,z) method(x,y,z)

字符函数:

length() : 统计字符个数 length()

lengthb() : 统计字节个数 中文算俩 getBytes().length

upper() : 全部转换成大写 toUpperCase()

lower() : 全部转换成小写 toLowerCase()

initcap() : 每个单词首字母大写 split() + substring() + toUpperCase()

replace() : 查找替换 replaceAll()

concat() : 连接字符串 || + concat()

substr() : 截取字符串

\*:substr(a,b,c)

a=>原始内容

b=>第几个字开始

c=>截取几个字

instr() : 查找指定内容出现的位置 indexOf()

\*:注意不会返回-1

如果第一个字就是那就返回1

返回0代表不存在！！！

\*:instr(a,b,c,d)

a=>原始内容 必须有

b=>要查找的内容 必须有

c=>从第几个字开始 注意支持负数

d=>找第几个

trim():去除两端空白

ltrim():去除左侧空白

rtrim():去除右侧空白

lpad():在指定内容的左侧填充 填充到多少位 以什么内容填充

rpad():在指定内容的右侧填充 填充到多少位 以什么内容填充

数值函数:

ceil() : 求不大于指定值的最小正数 => 向上取整

floor() : 求不小于指定值的最大正数 => 向下取整

round() : 四舍五入

trunc() : 直接截断

sign() : 求符号位 正数1 负数-1 零返回0

abs() : 求绝对值

power() : power(a,b)求a的b次方

sqrt() : 求正平方根

日期函数:

sysdate

systimestamp

add\_months(日期,多少个月) : 从指定日期的基础上添加多少个月

months\_between(日期1,日期2) : 两个日期差多少个月

last\_day() : 求指定日期的当月的最后一天...

\*:求上个月的第一天~

next\_day() : 求指定日期的下一个周几

转换函数:

to\_number()

to\_char(birthday,'yyyy-mm-dd')

to\_date('2017-10-31','yyyy-mm-dd')

\*:Java:yyyy-MM-dd HH:mm:ss

Oracle:yyyy-mm-dd hh24:mi:ss

mm月份 mi分钟

hh24代表24小时制

通用函数:

nvl() : 空值处理

nvl(字段,替换显示的内容)

nvl2() : 空值处理二代

nvl(字段,不是空显示什么, 是空显示什么)

decode() : 解码 解码一个字段 可变参哦

\*: 经常结合sign来做范围解码

===========================================================================================

约束:

主键 primary key

外键 foreign key \*: references

非空 not null

唯一 unique

检查 check

create table teacher(

id number(3) primary key,

name varchar2(20) not null unique,

salary number(8,2) check( salary <= 200000)

);

insert into teacher values(1,'JayZhou',20888);

insert into teacher values(2,'Joshua',19999);

insert into teacher values(3,'Comi5v',500);

insert into teacher values(4,'aYi',19998);

create table class(

id number(3) primary key,

name char(6) not null unique,

qq varchar2(20) check(length(qq) >=8 ),

tid number(3) references teacher(id)

);

insert into class values(1,'ET1709','1111111111',3);

insert into class(id,name,tid) values(2,'ET1710',4);

insert into class(id,name,tid) values(3,'ET1707',1);

create table school(

id number(3) primary key,

name varchar2(20) not null unique,

phone varchar2(20) check( length(phone)>=7 )

);

insert into school values(1,'易大','0531-55555511');

insert into school values(2,'山大','0531-77777722');

insert into school values(3,'海大','0532-33333333');

insert into school values(4,'石油大','053X-1111111');

create table student(

id number(3) primary key,

name varchar2(20) not null,

birthday date,

salary number(7,2) check(salary between 10000 and 30000),

email varchar2(50) unique,

sid number(3) references school(id),

cid number(3) references class(id)

);

insert into student values(1,'王老板儿',to\_date('19990918','yyyymmdd'),10800,'990918@qq.com',1,1);

insert into student values(2,'刘总',to\_date('19370418','yyyymmdd'),20800,'370418@qq.com',4,2);

insert into student values(3,'哪吒',to\_date('19440501','yyyymmdd'),20800,'111111@qq.com',3,1);

insert into student values(4,'三毛儿',to\_date('19960507','yyyymmdd'),20800,'333333@qq.com',1,3);

#:如何查看所有已经存在的约束:

select constraint\_name,constraint\_type from user\_constraints where table\_name=upper('student');

#:如何删除约束:

alter table student drop constraint SYS\_C0011129;

#:学会在创建表的时候直接给约束起名字 以防止系统命名根本看不懂

create table test(

id number(5) constraint PK\_ID primary key,

name varchar2(20) constraint C\_NAME\_NN not null

);

alter table test drop constraint PK\_ID;

排序

order by 排序字段 asc

asc代表升序 desc 代表降序 默认升序

order by 后面也可以跟上多个排序字段

order by 字段1 asc,字段2 desc,字段三 asc

== 不要因为讲的时间短 就认为它不重要！ ==

分组

group by

非常重要!!!

【如果以字段a分组 那么只能直接查询字段a

或者用组函数统计其它字段 不能直接查询其它字段】

非常重要!!!

【分组之后 很可能又要对小组进行条件过滤

比如 我要显示所有3个动物以上的小组

having 用于分组之后的过滤

过滤的不是一行数据 而是一个小组】

select 分组字段,某组函数(某字段) from 表名 where 对每条记录的过滤条件

group by 分组字段

having 对组的过滤条件

where 是在分组之前 等于对每个同学的判断

有没有资格去参与分组

having 是在分组之后 等价于对最后的分组结果进行判断

有没有资格显示出来

子查询 = 嵌套查询 = 某些查询的条件是通过查询的出来的

【子查询出现在列名的位置 绑定到列的子查询】

1st.请显示 学生名字,学校名字

select name,子查询 from student;

在student表只能直接查到学校的id

但是可以通过学校id去school表查名字啊~

【子查询出现在where条件当中】

2nd.跟王老板一个学校的同学都有谁

a>我们先要查出王老板的学校id

select sid from student where name = '王老板'

查询所有学生当中跟a相同的

select name from student where sid = (select sid from student where name like '王老板%');

【明天讲：出现在from后面 where前面 把子查询当做一张表 再做查询】

\*\*\*\*\*: select id,子查询 from (子查询) where (子查询)

连表查询

内连接:

select student.name,school.name from student join school on student.sid = school.id;

左外连接:以左表为主，左表当中每一行数据，无论是否满足连表的条件，最终都要显示，但是关联表的所有信息都以空显示

select student.name,school.name from student left join school on student.sid = school.id;

右外连接:右表为主，右表当中每行数据都要显示，无论是否满足连表条件

select student.name,school.name from student right join school on student.sid = school.id;

全外连接:两张表的数据都要显示

select student.name,school.name from student full join school on student.sid = school.id;

交叉连接:两张表的数据一一对应 交叉形成最终结果

select student.name,school.name from student cross join school;

\*: join on

left join on

right join on

full join on

cross join 没有on

\*: 很可能在将来的工作当中你会遭遇自己连自己的情况...

必须要给表起别名~

select a.name,c.name from student a

left join student c

on a.bfid = c.id;

===========================================================================================

排序 order by

分组 group by

子查询

(字段 子查询) from(子查询) where (子查询)

连表查询

内连接 : inner join == join on

左外连接 : left outer join == left join on

右外连接 : right outer join == right join on

全外连接 : full outer join == full join on

交叉连接 : cross join

伪列 = 假列 = 根本不存在的列

ROWNUM = 这是Oracle对查询结果动态的编号

用来实现分页查询！

ROWID = 是映射每一行数据物理地址的唯一标识

通常用于删除完全重复的数据

delete from lyric where rowid not in (select min(rowid) from lyric group by content);

rownum实现分页查询 思路和步骤

1.假如我们想要的是第一页数据 编号1-10

select rownum,substr(content,1,10)||'...' from lyric where rownum between 1 and 10;

2.但是如果我们需要的是第二页数据 ： 11-20

select rownum,substr(content,1,10)||'...' from lyric where rownum between 1 and 20;

3.我们需要用2的结果当成一张表 从其中取后一部分截

a>需要给rownum起别名 否则会形成误解

b>子查询也得起个名字

select rownum,x.\* from (select rownum rn,substr(content,1,10)||'...' from lyric where rownum <=20)x where rn >=11;

4.如果在分页显示的时候需要排序数据，请注意，必须先排序，然后再去rownum，也就是说rownum不能和order by出现在同一层查询

select rownum,x.\* from (select rownum rn,substr(content,1,10)||'...' from (select \* from lyric order by content)where rownum <=20)x where rn >=11;

没有排序的：

截取1-30数据：

select rownum,lyric.\* from lyric where rownum <= 30;

从1-30当中截取21到30;

select rownum,a.\* from (select rownum rn,lyric.\* from lyric where rownum <= 30)a where rn >=21;

带排序的：

select rownum,a.\* from (select rownum rn,x.\* from (select \* from lyric order by content asc) x where rownum <= 30) a where rn >= 21;

不推荐用 但是也行的方法

select rownum,a.\* from (select rownum rn,c.\* from (select \* from lyric order by content asc)c)a where rn between 21 and 30;

#:涉及到分页 那么分页的N要素 大家必须得会计算:

1.总记录数 总共有多少件商品需要展现

[从数据库当中查出来 select count ]

假设:a 77

2.每页显示多少条商品

[用户选择 或者 系统内置]

假设:b 10

3.总共有多少页

[通过a和b计算得出 ]

假设:c 则 c = (a+b-1) / b

4.当前页:

[由用户选择决定]

[假设为:d]

4-1:该页起始点

b\*(d-1) + 1 101

4-2:该页结束点

b\*d 150

select b.\* from (select rownum rn,a.\* from (select \* from student where name like '王%' and salary between 10000 and 50000 order by id) a where rownum <= 150)b where rn >= 101

===========================================================================================

#:联合关键字

union 相当于Set集合的addAll() 结果唯一

union all 相当于List集合的addAll() 结果不唯一

intersect retainAll() 求交集

minus removeAll()

#: in

not in

all

any/some 有范围判断的子查询条件当中经常出现

我想查询大于 09班所有同学薪资的 其它同学

select name,salary from student where salary > any (select salary from student where cid = 1);

索引 序列 视图

index sequence view

1st.索引 => 目录

create index idx\_student on student(name);

\*:所有的主键都会自动创建索引

\*:创建索引是不是只有优没有劣?

如何删除索引:

drop index idx\_student;

2nd.序列 => int i 一个计数的变量而已

\*: 通常情况下用于生成主键

create sequence seq\_student

increment by 5 步长 一次涨几

start with 10 从几开始

minvalue 5 最小值

maxvalue 100 最大值

cycle 循环

nocache 不缓存

\*:如何取序列的下一个值 ++i

seq\_student.nextval

\*:如何取序列的当前值 i

seq\_student.currval

\*:如何删除序列

drop sequence seq\_student;

视图 => 假表 用查询结果动态的生成一张表 view

create view viewname as select student.name sn,school.name cn from student left join school on student.sid = school.id;

drop view viewname;