Create tablespace xxx

Datafile ‘c:/xxxx.dbf’

Size 100m

Autoextends on

Next 10m

Create user aaaa

Identified by hello

Default tablespace xxx

Grant dba/connection/resources to aaaa

程序 = 数据+算法

Sql:查询，多表查询，子查询

## Oracle自己的编程的语法

## 今日重点

1. 索引
2. **存储过程**
3. **触发器**
4. **Java调用存储过程**

## 01视图

1. 语法

Create view xxxx as select ename,job from emp;

特点：数据来源于表，存储了一个查询语句

试图不存储数据，查询效率表要高于试图

物化视图

作用：屏蔽敏感字段,数据更安全

## 02索引

作用：提高查询效率

Create index xxx on 表名（字段）

什么时候适合给表建立索引？

1. 经常改动字段不适合，经常作为查询条件的字段适合
2. 表中数据量大，查询结果集小
3. 字段中空值多
4. 多表连接的连接字段

查询中以下情况索引失效！

原始值：索引才能使用

单行函数，运算，模糊查询like等都会让索引失效

默认创建的索引

主键和唯一约束的字段会自动创建唯一索引

## 03plsql编程语言定义变量

1. 变量定义

变量名称 类型 ：= 赋值

类型：number,varchar,varchar,date,

引用类型：表名.字段名%type 引用某个表中的某个字段的类型和长度

记录类型：emp%rowtype, 跟这个表的一条记录一样，存储一行

控制台打印输出：dbms\_output.put\_line(xxxxx)

常量：

c\_pi CONSTANT NUMBER(3,2) := 3.14;

结构块：

Declare

变量定义

Begin

逻辑

End;

*--plsql*

DECLARE

*--定义变量*

v\_id NUMBER(10) := 30 ;

*--常量*

c\_pi CONSTANT NUMBER(3,2) := 3.14;

*--存储7839的人的名字*

v\_name emp.ename%TYPE ;

*--存储一条记录*

*--emprow emp%ROWTYPE ; --存储属性名和表字段名称一致*

BEGIN

*--v\_id := 20;*

*--c\_pi := 3.14;*

v\_id := LENGTH('hello');

*--通过select赋值*

*--SELECT \* INTO emprow FROM emp WHERE empno=7839;*

SELECT ename INTO v\_name FROM emp WHERE empno = 1111;

*--输出*

*--dbms\_output.put\_line(v\_id);*

*--dbms\_output.put\_line(c\_pi);*

dbms\_output.put\_line(v\_name);

END;

总结：

视图 create view xxx as select \*…….

索引 Create index xx on 表名称（字段）

什么时候适合建立索引，什么情况索引失效！

Plsql入门，基本语法：变量的定义赋值

Create talbe xx (id number(10))

## 04plsql中的if判断

If then

Elsif then

Else

End if;

## 05plsql中的循环

Loop

Exit when 条件；

循环

End loop;

For 变量 in 开始..结束 loop

End loop;

BEGIN

FOR i IN REVERSE 1..10 LOOP

dbms\_output.put\_line(i);

END LOOP;

END;

## 06plsql中的游标

类似java中的集合，可以存储多行多列

1. 定义游标

Cursor 名字 is select 。。。。。

1. 开启游标

Open 名字

1. 抓取数据

Fetch 游标 into 变量

1. 关闭游标

Close 名字

## 07存储过程的使用

长期存储在数据库中，类似表一样，定义存储过程

Create or replace procedure 名字（参数名 类型）

Is

定义变量

Begin

End;

存储过程的调用

Begin

名字（）；

End;

CREATE OR REPLACE PROCEDURE p1(eno emp.empno%TYPE)

IS

v\_sum NUMBER(20);

BEGIN

SELECT sal+nvl(comm,0) INTO v\_sum FROM emp WHERE empno=eno;

dbms\_output.put\_line(v\_sum);

END;

有输出的存储过程

CREATE OR REPLACE PROCEDURE p1(eno emp.empno%TYPE,v\_sum OUT NUMBER)

IS

BEGIN

SELECT sal+nvl(comm,0) INTO v\_sum FROM emp WHERE empno=eno;

END;

复习：

Create view xxx as select ……

Create index xxx on 表（字段）

Plsql 在数据库中编程 基本语法：变量，判断，循环，游标

## 08存储函数

Create or replace function 名字（参数名 类型） return 类型

Is

定义变量

Begin

Return xxx;

End;

调用函数

SQL> select fun1(7788) from dual;

什么时候使用？

算数运算，计算类型的交给函数

逻辑业务处理的交给存储过程

## 09out类型参数的讲解

参数值返回出去就使用out

## 10存储函数和存储过程的区别

区别？

函数必须有return关键字

存储过程没有return关键字

函数可以在select查询中调用执行

## 11触发器的概念和分类

触发器自动触发执行：数据发生增删改的时候执行

分类

语句级别：

行级别 ：敏感，每一行触发，for each row

## 12触发器案例演示

Create or replace trigger xxxx

After|before update or insert or delete on 表名称 for each row

Declare

Begin

End;

## 13触发器实现主键自增

:old 更新前的旧值

:new 更新后的新值

## 14java调用存储过程之环境准备

PreparedStatement

## 15java调用存储过程之环境测试

## 16java调用存储过程和存储函数

CallableStatement cstmt = conn.prepareCall(“{call 存储过程名字（？，？）}”);

设置参数实际值

**输出参数则设置类型，OracleTypes.xxxx**

执行存储过程

Cstmt.execute();

获得输出参数的值

String out =Cstmt.getObject(xxx) ;

总结：

讲了几个对象 view,index,procedure,function,trigger,jdbc调用存储过程和函数

1. 语法：

Create view xxxx as select ….

Create index xxxx on 表（字段）

Create procedure xxx(参数 类型)

Is

Begin

End;

Create or replace function xxx(参数 类型)

Return 类型

Is

Begin

Return 值；

End;

Create or replace trigger xxx

Before|after insert or update or delete on 表 [for each row]

Declare

Begin

End;

作业：

1. 写一个存储过程，输入1-7数字，输出是星期几？
2. 写一个存储过程输入一个数字，控制台输出1-数字之内的所有素数！
3. 定义一个触发器，周六，周日不能对emp表进行增删改操作！
4. 做表的备份！
5. 定义一个函数，输入员工编号，计算某个人的年薪！（sal+comm）\*12
6. 定义一个触发器，只有‘MANAGER’能够月薪上万！

回顾:

Oracle数据库

Sql:类似mysql sql99标准

数据类型 number varchar2

To\_char()

To\_date()

Nvl()

视图

Create view xxxx as select …. With read only

存储sql,数据来源于表

索引

Create index xxxx on 表名（字段）

数据库优化？ 原值

索引提高查询效率，

存储过程

Create or replace procedure xxxx(参数 类型，参数 out 类型)

Is

Begin

End;

存储函数

Create or replace function xxx(参数 类型) return 类型

is

begin

return xxxx;

end;

触发器

Create or replace trigger xxx

Before|after update or insert or delete on 表名 [for each row]

Declare

Begin

End;

序列

Create sequence xxxx increment by 1 start with 1 ;