理论部分

1、如果需要向表中插入一批已经存在的数据，可以在INSERT 语句中使用select 语句。

2、通过视图修改数据时，实际上是在修改基本表中的数据。

3、在众多的事务控制语句中，用来撤销事务的操作的语句为Rollback，用于持久化事务对数据库操作的语句是 commit。

4、在多进程Oracle 实例系统中，进程分为用户进程、后台进程和服务器进程。

5、SYS用户模式存储数据字典表和视图对象。

6、iSQL\*Plus必须的服务是OracleServiceSID

7、调用一个过程使用EXEC 命令

8、字符串数据”test”分别存放到char(10)和varchar2(10)类型的字段中，其实际存储长度为(  10，4   )

9、控制一个显式游标时，FETCH命令包含INTO子句(     )

10、判断列值是否为空的操作符是AS  NULL

11、AVG(\*)         B. SUM(\*)         C. COUNT(\*)       D. MAX(\*) 的作用

12、SQL 命令的运行结果保存到文件中使用SPOOL  命令

13、去除结果集中重复的行，可在SELECT 语句中使用DISTINCT关键字

14、角色是具有名称的一组相关权限的组合。

15、一个表空间物理上对应一个或多个数据文件。

16、标准的SQL 语言语句类型可以分为：数据定义语言（DDL）、数据操纵语句（DML）和数据控制语言（DCL）。

  17、在Oracle中，一个用户拥有的所有数据库对象统称为 模式

18、在 Oracle数据库的逻辑结构中组件从大到小依次是   表空间   段   . 区    数据块

19、实现对一个结果集进行分组和汇总：GROUP  BY

20、命名的PL/SQL块有程序包    过程       函数

21、新建用户时会将 system的表空间users分配给用户作为默认表空间。

操作部分：

1、查看本用户下有哪些表:select table\_name from user\_tables;

2、查看本用户下的emp表的结构: desc emp

创建一个表class，字段为id(number),stuname(varchar2(20))

Create table class(id number, stuname varchar2(20));

3、为Class 表的ID 列添加一个名为PK\_CLASS\_ID 的主键约束， ALTER TABLE Class Add constraint PK\_CLASS\_ID PRIMARY KEY(ID)

4、假设当前用户是SCOTT用户以普通用户登录，口令为tiger，请写出该用户以管理员身份登录的命令 conn scott/tiger as sysdba;

5、假设用户SCOTT以管理员身份登录，现需创建用户Lisa，口令是Lisa，请写出相应的命令create user Lisa identified by Lisa；

 6、假设用户Lisa用Lisa以普通用户身份登录到系统，现需创建一个UPDATE 语句来修改本用户下ARTISTS 表中的数据，并且把每一行的T\_ID 值都改成15，应该使用的SQL 语句是\_ update artists set T\_ID =15

7、假设用户SCOTT以管理员身份登录，已知用户账号Lisa被锁定，需为他解除锁定，请写出相应的命令 alter user Lisa account unlock;

8、假设用户SCOTT以管理员身份登录，需授予用户Lisa对SCOTT用户模式下的EMP表进行查询(SELECT)的权限，请写出相应的命令\_ grant select on emp to Lisa；

9、用命令select \* from emp查询emp表中的数据，现需将缓冲区中的SQL 命令保存到C盘根目录下的emp.txt文件中, 请写出相应的命令：save  C:\emp.txt；

;

**综合实例：**

student表（学生表）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 学号 | 姓名 | 性别 | 年龄 | 系别 |
| 1 | 吴好 | 男 | 18 | 计算机系 |
| 2 | 崔平 | 女 | 21 | 经管系 |
| 3 | 钱筱 | 女 | 19 | 电子系 |

course表(课程表)

|  |  |  |
| --- | --- | --- |
| 课程号 | 课程名 | 学分 |
| 1 | SQL Server | 4 |
| 2 | 数据结构 | 3 |
| 3 | 专业英语 | 2 |

sc表（选课表）

|  |  |  |
| --- | --- | --- |
| 学号 | 课程号 | 成绩 |
| 1 | 1 | 88 |
| 2 | 1 | 90 |
| 2 | 2 | 70 |
| 3 | 3 | 79 |

1. 把course表中课程号为3的课程的学分修改为3。

Update course set 学分=3 where 课程号=3;

1. 在student表中查询年龄大于18的学生的所有信息，并按学号降序排列。

Select \* from student where 年龄>18 order by 学号 desc;

1. 在以上三个表中查询选的课程的“学分”为3，并且成绩大于80的学生的学号、姓名和性别。

Select s.学号,s.姓名,s.性别 from student s，course c，sc where c.学分=3 and sc.成绩>80 and s.学号= sc.学号and c.课程号=sc.课程号;

4、为student表创建一个名称为my\_trig的触发器，当用户成功删除该表中的一条或多条记录时，触发器自动删除SC表中与之有关的记录。

（注：在创建触发器之前要判断是否有同名的触发器存在，若存在则删除之。）

**Create or replace trigger my\_trig**

**before delete on student**

**Begin**

**Delete from sc where 学号 not in (select 学号 from** student**)**

**End my\_trig;**

以下存储过程借助游标用于实现对员工工资(Sal)的调整，调整方案为：对于员工号(empno)在7000到7200间的员工，若其原工资在1000之下，则加薪20%。 (1) 请完成以下程序：

CREATE OR REPLACE PROCEDURE Rise\_sal

 IS  v\_empno    Emp.empno%type;

 v\_sal       Emp.sal%type;

 V\_ErrorText      Varchar(200);

//定义游标

CURSOR cursor\_sal

  IS

SELECT empno,sal  FROM   emp WHERE empno BETWEEN 7000 AND 7200;

BEGIN

OPEN cursor\_sal   ;    //打开游标

 LOOP

 FETCH cursor\_sal  INTO v\_empno,v\_sal ;

  EXIT WHEN cursor\_sal%NOTFOUND;//逐行提取数据

 IF v\_sal<1000 THEN

UPDATE Emp SET sal= sal\*0.20+Sal

//更新数据

 END LOOP;

CLOSE cursor\_sal;   //关闭游标

COMMIT;          //提交事务

Exception

 When others THEN

 CLOSE cursor\_sal;    //关闭游标

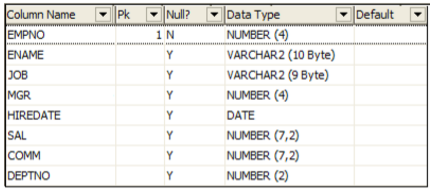
rollback;         //回滚事务

v\_Errortext:='游标使用错误';

dbms\_output.put\_line('程序异常终止，出现以下错误：'||v\_Errortext);

END;

(2) 假设用户已登录SQL Plus系统，处于“SQL>\_”命令提示符下，现需执行存储 过程Rise\_sal， 命令如下：EXEC Rise\_sal；

2. 创建一个函数get\_sal，用于实现对表Emp(如下图)给定一个员工号(Empno)， 通过函数获取该雇员的工资值 ，请完成以下程序。

(1)请将函数get\_sal补充完整

 CREATE OR REPLACE FUNCTION get\_sal  (v\_emp\_no  IN  emp.empno％TYPE)

RETURN  NUMBER

IS

v\_emp\_sal  emp.sal％TYPE:＝0；

  BEGIN

 SELECT sal      INTO v\_emp\_sal      FROM emp

WHERE empno= v\_emp\_no ;

RETURN(v\_emp\_sal)；

 END get\_sal；

(2)用函数get\_sal查询员工7521号的工资，请写出一种查询方式。

Select get\_sal (7521)  from dual;