《数据库系统原理》实验报告（六）					
题目：上机实验课（六）触发器					
学号	1853790	姓名	庄镇华	日期	2020.11.17
实验环境：VMware 虚拟机 Red Hat5 系统下的 oracle 环境					

实验步骤及结果截图：

1 建立 table studentsA

Sno char(9) primary key

Sname char(20) unique

Ssex char(20)

Sage smallint

Sdept char(20)

```
CREATE TABLE  studentsA
(
    Sno char(9) PRIMARY KEY,
    Sname char(20),
    Ssex char(20),
    Sage smallint,
    Sdept char(20),
    CONSTRAINT studentA_u1 UNIQUE (Sname)
);
DESC studentsA;
```

Table created.

Name	Null?	Type
SNO	NOT NULL	CHAR(9)
SNAME		CHAR(20)
SSEX		CHAR(20)
SAGE		NUMBER(38)
SDEPT		CHAR(20)

2 建立 table coursesA

Cno char(4) primary key

Cname char(40)

Cpno char(4)

Ccredit smallint

```
CREATE TABLE coursesA
(
    Cno char(4) PRIMARY KEY,
    Cname   char(40),
    Cpno    char(4),
    Ccredit smallint
);
DESC coursesA;
```

Name	Null?	Type
CNO	NOT NULL	CHAR(4)
CNAME		CHAR(40)
CPNO		CHAR(4)
CCREDIT		NUMBER(38)

3 建立 table SC

Sno char(9)

Cno char(4)

Grade smallint

primary key (Sno,Cno)

foreign key (Sno) references student(Sno)

foreign key (Cno) references course(Cno)

```
CREATE TABLE SC
(
    Sno char(9),
    Cno char(4),
    Grade   smallint,
    PRIMARY KEY (Sno, Cno),
    FOREIGN KEY (Sno) REFERENCES studentsA(Sno),
    FOREIGN KEY (Cno) REFERENCES coursesA(Cno)
);
DESC SC;
```

Name	Null?	Type
SNO	NOT NULL	CHAR(9)
CNO	NOT NULL	CHAR(4)
GRADE		NUMBER(38)

4 插入数据

insert into studentsA values ('200215121','Li Yong','male',20,'CS');

insert into studentsA values ('200215122','Liu Chen','female',19,'CS');

insert into studentsA values ('200215123','Wang Min','female',18,'MA');

insert into studentsA values ('200215125','Zhang Li','male',19,'IS');

SNO	SNAME	SSEX	SAGE	SDEPT
200215121	Li Yong	male	20	CS
200215122	Liu Chen	female	19	CS
200215123	Wang Min	female	18	MA
200215125	Zhang Li	male	19	IS

insert into coursesA(Cno,Cname,Cpno,Ccredit) values
('0001','database','5',4);

insert into coursesA (Cno,Cname,Ccredit) values ('0002','maths',2);

insert into coursesA (Cno,Cname,Cpno,Ccredit) values ('0003','IS','1',4);

insert into coursesA (Cno,Cname,Cpno,Ccredit) values ('0004','OS','6',3);

insert into coursesA (Cno,Cname,Cpno,Ccredit) values ('0005','data
struncture','7',4);

insert into coursesA (Cno,Cname,Ccredit) values ('0006','data process',2);

insert into coursesA (Cno,Cname,Cpno,Ccredit) values
('0007','PASCAL','6',4);

CNO	CNAME	CPNO	CCREDIT
0001	database	5	4
0002	maths		2
0003	IS	1	4
0004	OS	6	3
0005	data struncture	7	4
0006	data process		2
0007	PASCAL	6	4

```
insert into SC values ('200215121','0001',92);
insert into SC values ('200215121','0002',85);
insert into SC values ('200215121','0003',88);
insert into SC values ('200215122','0002',90);
insert into SC values ('200215122','0003',80);
```

SNO	CNO	GRADE
200215121	0001	92
200215121	0002	85
200215121	0003	88
200215122	0002	90
200215122	0003	80

5. 作业：创建触发器

1）创建一个触发器，使得每当新建一门课程时，判断输入信息的合法性（必须得有主键非空，实现其他有加分），只有合法的信息才能创建。

检查主键空白、学分为负等错误。

```
CREATE OR REPLACE TRIGGER trig_insert BEFORE UPDATE OR INSERT ON coursesA
    REFERENCING NEW AS nrow
    FOR EACH ROW
    WHEN (nrow.Cno IS NULL OR nrow.Ccredit < 0)
    BEGIN
        RAISE_APPLICATION_ERROR(-20001, '插入信息非法');
    END;
```

```
Trigger created.
```

```
INSERT INTO coursesA (Cname, Cpno, Ccredit)
    VALUES ('database','5',4);
INSERT INTO coursesA (Cno, Cname, Cpno, Ccredit)
    VALUES ('0000', 'database','5',-4);
```

```
INSERT INTO coursesA (Cname, Cpno, Ccredit)
            *

ERROR at line 1:
ORA-20001: 插入信息非法
ORA-06512: at "ORACLE.TRIG_INSERT", line 2
ORA-04088: error during execution of trigger 'ORACLE.TRIG_INSERT'
INSERT INTO coursesA (Cno, Cname, Cpno, Ccredit)
                      *

ERROR at line 1:
ORA-20001: 插入信息非法
ORA-06512: at "ORACLE.TRIG_INSERT", line 2
ORA-04088: error during execution of trigger 'ORACLE.TRIG_INSERT'
```

2）利用系统触发器实现数据库启动和关闭的日志（ppt18 页）

```
CREATE TABLE database_logs
(
    ts TIMESTAMP PRIMARY KEY,
    log varchar(500)
);
DESC database_logs;
```

Table created.

Name	Null?	Type
TS	NOT NULL	TIMESTAMP(6)
LOG		VARCHAR2(500)

建立触发器

```
CREATE OR REPLACE TRIGGER trig_startup_log AFTER STARTUP ON DATABASE
    BEGIN
        INSERT INTO database_logs VALUES(sysdate, ora_sysevent);
    END;
```

```
CREATE OR REPLACE TRIGGER trig_shutdown_log BEFORE SHUTDOWN ON DATABASE
    BEGIN
        INSERT INTO database_logs VALUES(sysdate, ora_sysevent);
    END;
```

Trigger created.

Trigger created.

进行测试

```
SQL> shutdown immediate
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL> startup
ORACLE instance started.

Total System Global Area  444596224 bytes
Fixed Size                  1219904 bytes
Variable Size             188744384 bytes
Database Buffers          251658240 bytes
Redo Buffers                2973696 bytes
Database mounted.
Database opened.
SQL> select * from database_logs;

TS
---------------------------------------------------------------------------
LOG
---------------------------------------------------------------------------
17-NOV-20 03.13.35.000000 AM
SHUTDOWN

17-NOV-20 03.14.05.000000 AM
STARTUP
```

出现的问题：

oracle 触发器中 when 语句中不能有子查询，本来打算使用子查询实现查询主键是否重复，但是运行出错。

解决方案：

通常情况下，oracle 数据库禁止在行级触发器或行级触发器所调用的子程序中使用查询语句。如果必须要在 trigger 中使用查询语句，oracle 也提供了一种途径。主要利用声明变量解决。

下面以简单的代码为例做说明：

```
CREATE OR REPLACE TRIGGER TRG_TEST
    BEFORE INSERT OR UPDATE OR DELETE ON SCOTT.EMP
    FOR EACH ROW -- 行级触发器
DECLARE
    PRAGMA AUTONOMOUS_TRANSACTION; -- 解决行级触发器不能使用查询语句的关键

    V_COUNT PLS_INTEGER;
BEGIN
   SELECT COUNT(*) INTO V_COUNT FROM SCOTT.EMP;
    IF V_COUNT > 0 THEN
        -- DO SOMETHING
        NULL;
    END IF;

    COMMIT; -- 提交
EXCEPTION
    WHEN OTHERS THEN
        ROLLBACK; -- 回滚
END TRG_TEST;
```