

实验三 SQL数据完整性

学号: 3220104519

姓名: 蔡佳伟

一、实验目的

1. 熟悉通过SQL进行数据完整性控制的方法

二、实验环境

Windows11;

DBMS: MySQL

三、实验原理

数据完整性在SQL Server中,分为实体完整性、域完整性和引用完整性三类。

实体完整性

主键约束: 主键的列取值不能空也不能重复

唯一性约束: 非主键的列没有重复值所用的约束

域完整性 (列完整性)

空值约束:设置可以为空/不能为空

默认约束: 为列设置默认值

检查约束: 为列设置取值范围

引用完整性(参照完整性)(外键约束)

主键在被参照表中(主键表),外键在参照表中(外键表)

四、实验过程

1. 定义约束

在实验2的基础上,对表增加约束。

```
mysql> select * from student;
 id
                             birthday
               name
                                           major
                                                    score
 3220100009
                             2003-05-11
                                                       90
               pear
                                           IS
 3220100101
                                           CS
               Myfriend
                                                       66
                             2000-01-01
 3220100121
               apple
                             2003-11-01
                                           CS
                                                       76
 3220101234
               Caijianwei
                             2003-01-01
                                           SE
                                                      100
 3220104519
               Caijiawei
                             2004-05-17
                                           SE
                                                       90
 rows in set (0.02 sec)
mysql> select * from dorm;
          number
 part
                   name
  紫云5
             601
                    Applepen
  碧峰1
             111
                   Caijiawei
 rows in set (0.02 sec)
```

```
mysql> desc dorm;
 Field
                           Nu11
                                         Default
           Type
                                   Kev
                                                    Extra
                           YES
            varchar(10)
                                         NULL
 part
                                         NULL
                           YES
 number
            int
            varchar (20)
                           NO
                                   PRI
                                         NULL
 name
 rows in set (0.00 sec)
```

student中添加sex一栏,默认值为B,要求非空且字段只能在B和G中取值。

```
mysql> alter table student add sex char(1) default 'B' not null check(sex='B' or sex='G');
Query OK, 5 rows affected (0.05 sec)
Records: 5 Duplicates: 0 Warnings: 0
```

为dorm添加依赖于student中主键id的外键id

```
mysql> alter table dorm add foreign key(id) references student(id);
Query OK, 2 rows affected (0.05 sec)
Records: 2 Duplicates: 0 Warnings: 0
mysql> select * from dorm;
  part
           number
                      name
                                    id
                                    NULL
   紫云5
               601
                      Applepen
  碧峰1
               111
                      Caijiawei
                                    NULL
  rows in set (0.00 sec)
```

```
mysql> desc dorm;
 Field
                           Nu11
                                  Kev
                                         Default
           Type
                                                    Extra
                           YES
           varchar(10)
                                         NULL
 part
                           YES
 number
           int
                                         NULL
                           NO
           varchar(20)
                                  PRI
                                         NULL
 name
           varchar(10)
                           YES
  id
                                  MUL
                                         NULL
 rows in set (0.01 sec)
```

2.主键约束验证

向dorm中插入一行,其中主键重复,结果出现以下信息。证明主键约束。

```
mysql> insert into dorm values('紫云3',222,'Applepen','3220101111');
ERROR 1062 (23000): Duplicate entry 'Applepen' for key 'dorm.PRIMARY'
```

3.字段类型约束验证

前面说明了student中sex只能在B和G中取值,尝试更新语句如下。证明了字段类型约束。

```
mysql> update student set sex='W' where name='Caijiawei';
ERROR 3819 (HY000): Check constraint 'student_chk_1' is violated.
```

4.外键约束验证

part1

更新外键属性,不能成功更新,证明外键约束。

```
mysql> update dorm set id='3230100101' where name='Caijiawei';
ERROR 1452 (23000): Cannot add or update a child row; a foreign key constraint fails (`lab2`.`dorm`, CONSTRAINT `dorm_ib
fk_l`FOREIGN KEY (`id`) REFERENCES `student` (`id`))
```

暂时删除外键的约束属性, 然后更新成功。

```
mysql> set foreign_key_checks=0;
Query OK, 0 rows affected (0.00 sec)

mysql> update dorm set id='3230100101' where name='Caijiawei';
Query OK, 1 row affected (0.02 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> set foreign_key_check=1;
ERROR 1193 (HY000): Unknown system variable 'foreign_key_check'
mysql> set foreign_key_checks=1;
Query OK, 0 rows affected (0.00 sec)
```

更新dorm中的外键id,成功了,因为和student表中的结果符合。

```
mysql> update dorm set id='3220104519' where name='Caijiawei';
Query OK, 1 row affected (0.02 sec)
Rows matched: 1 Changed: 1 Warnings: 0
```

更新student表中的id,失败,由于受到dorm中的外键约束

mysql> update student set id='3220101111' where name='Caijiawei'; ERROR 1451 (23000): Cannot delete or update a parent row: a foreign key constraint fails (`lab2`.`dorm`, CONSTRAINT `dor m_ibfk_1` FOREIGN KEY (`id`) REFERENCES `student` (`id`))

part2

删除之前的外键后,为dorm新增一个外键did,依赖于student表中的id,采用update cascade模式

```
mysql> alter table dorm add column did varchar(10)
Query OK, O rows affected (0.03 sec)
Records: O Duplicates: O Warnings: O
         alter table dorm add constraint d_id foreign key(did) references student(id) on update cascade;
nvsa1>
Query OK, 2 rows affected (0.06 sec)
Records: 2 Duplicates: 0 Warnings: 0
                                             Key
                                                      Default
  part
               varchar(10)
  number
                                                      NULL
NULL
               varchar(20)
                                             PRI
MUL
               varchar(10)
  did
                                   YES
  rows in set (0.00 sec)
```

```
nysql> select * from dorm;
                                 did
 part
          number
                   name
 紫云5
碧峰1
             601
                   Caijianwei
                                 3220101234
                                 3220104519
             111
                   Caijiawei
 rows in set (0.00 sec)
nysql> update student set id='766666' where name='Caijiawei';
Query OK, 1 row affected (0.02 sec)
Rows matched: 1 Changed: 1 Warnings: 0
nysql> select * from dorm;
                                 did
          number
 part
                   name
 紫云5
碧峰1
             601
                   Caijianwei
                                 3220101234
             111
                   Caijiawei
                                  766666
 rows in set (0.00 sec)
```

可以看到,改变主表的id以后,从表的did也随之改变,体现了"主删从不见"的性质。

5.触发器使用

nysql> create trigger mytrigger after update on student for each row update student set score=score-10 where id=new.id Duery OK, O rows affected (0.02 sec)

为表插入触发器,尝试修改成员,报错,这是因为避免递归触发循环

```
mysql> update student set name='ZB' where id='3220100101';
ERROR 1442 (HY000): Can't update table 'student' in stored function/trigger because it is already used by statement whic
h invoked this stored function/trigger.
```

删除之前的触发器,增加新的触发器: 当更新student表时,为dorm表中id相同的行的number-10。

```
mysql> select * from dorm;
                                                                     did
                     number
    part
                                         name
     紫云5
                                         Caijianwei
                                                                     3220101234
                            601
                                                                     766666
                            111
                                        Caijiawei
    rows in set (0.01 sec)
mysql> create trigger mytrigger after update on student for each row update dorm set number=number-10 where dorm.did=new.id;
Query OK, 0 rows affected (0.02 sec)
mysql> update student set birthday='2000-01-01' where name='Caijiawei';
Query OK, 1 row affected (0.02 sec)
Rows matched: 1 Changed: 1 Warnings: 0
        number
                Caijianwei
Caijiawei
 紫云5
碧峰1
                            766666
```

插入触发器后,更新student表的birthday,发现对应id相同的student的number成功改变,说明触发器使用成功。

五、总结

本次实验中我主要学会了约束的使用和触发器的使用,尤其是外键约束和触发器的使用花费了不少时间,不过还是掌握不够熟练。数据库博大精深还是需要我反复练习反复尝试才能掌握得更好啊。