



课程名称:	数据库系统
实验名称:	SQL数据完整性
姓 名:	杨吉祥
学 院:	计算机科学与技术学院
系:	竺可桢学院图灵班
专 业:	计算机科学与技术
学 号:	3230106222
指导教师:	陈刚

一.实验目的:

熟悉通过SQL进行数据完整性控制的方法。

二.实验内容和要求:

1. 定义若干表，其中包括primary key, foreign key 和check的定义。

- 创建被引用表

```
mysql> CREATE TABLE Department (  
->     DeptID INT PRIMARY KEY,  
->     DeptName VARCHAR(50) NOT NULL,  
->     Budget DECIMAL(10, 2) CHECK (Budget > 0)  
-> );  
Query OK, 0 rows affected (0.05 sec)
```

- 创建引用表

```
mysql> CREATE TABLE Employee (  
->     EmpID INT PRIMARY KEY,  
->     EmpName VARCHAR(50) NOT NULL,  
->     Salary DECIMAL(10, 2) CHECK (Salary > 0),  
->     DeptID INT,  
->     FOREIGN KEY (DeptID) REFERENCES Department(DeptID)  
->         ON DELETE CASCADE  
->         ON UPDATE CASCADE  
-> );  
Query OK, 0 rows affected (0.04 sec)
```

2. 表中插入数据，考察primary key如何控制实体完整性。

- 插入数据到 Department 表

```
mysql> INSERT INTO Department (DeptID, DeptName, Budget) VALUES (1, 'HR', 100000.00);
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO Department (DeptID, DeptName, Budget) VALUES (2, 'IT', 200000.00);
Query OK, 1 row affected (0.01 sec)
```

- 插入数据到 Employee 表

```
mysql> INSERT INTO Employee (EmpID, EmpName, Salary, DeptID) VALUES (101, 'Alice', 50000.00, 1);
Query OK, 1 row affected (0.02 sec)

mysql> INSERT INTO Employee (EmpID, EmpName, Salary, DeptID) VALUES (102, 'Bob', 60000.00, 2);
Query OK, 1 row affected (0.01 sec)
```

- 尝试插入重复的主键，将会失败

```
mysql> INSERT INTO Employee (EmpID, EmpName, Salary, DeptID) VALUES (101, 'Charlie', 70000.00, 1);
ERROR 1062 (23000): Duplicate entry '101' for key 'employee.PRIMARY'
mysql>
```

3. 删除被引用表中的行，考察foreign key 中on delete 子句如何控制参照完整性。

- 删除 Department 表中的一行

```
mysql> DELETE FROM Department WHERE DeptID = 1;
Query OK, 1 row affected (0.01 sec)
```

- 由于 ON DELETE CASCADE, Employee 表中 DeptID = 1 的行也会被自动删除

```
mysql> SELECT * FROM Employee;
+-----+-----+-----+-----+
| EmpID | EmpName | Salary | DeptID |
+-----+-----+-----+-----+
|    102 | Bob     | 60000.00 |      2 |
+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

4. 修改被引用表中的行的primary key，考察foreign key 中on update 子句如何控制参照完整性。

- 修改 Department 表中的主键

```
mysql> UPDATE Department SET DeptID = 3 WHERE DeptID = 2;
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

- 由于 ON UPDATE CASCADE, Employee 表中 DeptID = 2 的行也会被自动更新为 DeptID = 3

```
mysql> SELECT * FROM Employee;
+-----+-----+-----+-----+
| EmpID | EmpName | Salary   | DeptID |
+-----+-----+-----+-----+
|    102 | Bob     | 60000.00 |    3   |
+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

5. 修改或插入表中数据, 考察check子句如何控制校验完整性。

- 尝试插入不符合 CHECK 约束的数据

```
mysql> INSERT INTO Employee (EmpID, EmpName, Salary, DeptID) VALUES (103, 'David', -1000.00, 3);
ERROR 3819 (HY000): Check constraint 'employee_chk_1' is violated.
mysql>
```

- 尝试更新数据为不符合 CHECK 约束的值

```
mysql> UPDATE Employee SET Salary = -5000.00 WHERE EmpID = 102;
ERROR 3819 (HY000): Check constraint 'employee_chk_1' is violated.
mysql>
```

6. 定义一个trigger, 并通过修改表中数据考察触发器如何起作用。

- 创建一个日志表

```
mysql> CREATE TABLE SalaryLog (
->     EmpID INT,
->     OldSalary DECIMAL(10, 2),
->     NewSalary DECIMAL(10, 2)
-> );
Query OK, 0 rows affected (0.05 sec)
```

- 创建触发器

```
mysql> DELIMITER $$
mysql> CREATE TRIGGER BeforeSalaryUpdate
-> BEFORE UPDATE ON Employee
-> FOR EACH ROW
-> BEGIN
->     INSERT INTO SalaryLog (EmpID, OldSalary, NewSalary)
->     VALUES (OLD.EmpID, OLD.Salary, NEW.Salary);
-> END$$
Query OK, 0 rows affected (0.01 sec)

mysql> DELIMITER ;
```

- 更新 Employee 表的 Salary

```
mysql> UPDATE Employee SET Salary = 70000.00 WHERE EmpID = 102;
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

- 查看 SalaryLog 表

```
mysql> SELECT * FROM SalaryLog;
```

EmpID	OldSalary	NewSalary
102	60000.00	70000.00

```
1 row in set (0.00 sec)
```