

# 数据库高级部分

## 1、编码mysql

### 1.1、查看mysql编码

```
mysql> show variables like 'character%';
```

Variable_name	Value
character_set_client	utf8
character_set_connection	utf8
character_set_database	latin1
character_set_filesystem	binary
character_set_results	utf8
character_set_server	latin1
character_set_system	utf8
character_sets_dir	/usr/share/mysql/charsets/

### 1.2、设置mysql编码

```
# vi /etc/my.cnf
```

如下(少补):

```
[mysqld]
character-set-server=utf8
collation-server=utf8_general_ci
sql_mode='NO_ENGINE_SUBSTITUTION'
```

```
[mysql]
default-character-set = utf8
```

```
[mysql.server]
default-character-set = utf8
```

```
[mysqld_safe]
default-character-set = utf8
```

```
[client]
default-character-set = utf8
```

重启mysql

```
# service mysqld restart
```

再次查看编码:

```
# mysql -uroot -p
mysql> show variables like 'character%';
```

```
+-----+-----+
| Variable_name | Value |
+-----+-----+
| character_set_client | utf8 |
| character_set_connection | utf8 |
| character_set_database | utf8 |
| character_set_filesystem | binary |
| character_set_results | utf8 |
| character_set_server | utf8 |
| character_set_system | utf8 |
| character_sets_dir | /usr/share/mysql/charsets/ |
+-----+-----+
```

```
8 rows in set (0.00 sec)
```

## 2、mysql的目录及配置文件

A、**/etc/my.cnf** 这是mysql的主配置文件

B、**/var/lib/mysql** mysql数据库的数据库文件存放位置

```
[root@mysql etc]# cd /var/lib/mysql/
[root@mysql mysql]# ll
total 176172
-rw-rw----. 1 mysql mysql      56 Oct 20 00:10 auto.cnf
-rw-rw----. 1 mysql mysql 79691776 Oct 21 04:46 ibdata1
-rw-rw----. 1 mysql mysql 50331648 Oct 21 04:46 ib_logfile0
-rw-rw----. 1 mysql mysql 50331648 Oct 20 00:09 ib_logfile1
drwx-----. 2 mysql mysql    4096 Oct 21 01:38 itcast
drwx-----. 2 mysql mysql    4096 Oct 21 04:45 menagerie
drwx--x--x. 2 mysql mysql    4096 Oct 20 00:09 mysql
-rw-r-----. 1 mysql root     2219 Oct 20 00:14 mysql.err
-rw-rw----. 1 mysql mysql      5 Oct 20 00:10 mysql.pid
srwxrwxrwx. 1 mysql mysql      0 Oct 20 00:10 mysql.sock
drwx-----. 2 mysql mysql    4096 Oct 20 00:09 performance_schema
-rw-r--r--. 1 root  root       111 Oct 20 00:09 RPM_UPGRADE_HISTORY
-rw-r--r--. 1 mysql mysql     111 Oct 20 00:09 RPM_UPGRADE_MARKER-LAST
drwxr-xr-x. 2 mysql mysql    4096 Oct 20 00:09 test
```

我们的mysql数据库的数据库文件通常是存放在了/var/lib/mysql这个目录下

C、**/var/log/mysql**数据库的日志输出存放位置

D、查看端口。Netstat -nltp 看是否能找到3306的端口

```
[root@mysql menagerie]# netstat -nltp
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:57958          0.0.0.0:*               LISTEN      1306/rpc.statd
tcp        0      0 0.0.0.0:111           0.0.0.0:*               LISTEN      1284/rpcbind
tcp        0      0 0.0.0.0:22            0.0.0.0:*               LISTEN      1573/sshd
tcp        0      0 127.0.0.1:631         0.0.0.0:*               LISTEN      1362/cupsd
tcp        0      0 127.0.0.1:25         0.0.0.0:*               LISTEN      1654/master
tcp        0      0 :::3306               :::*                     LISTEN      37860/mysqld
tcp        0      0 :::49647              :::*                     LISTEN      1306/rpc.statd
tcp        0      0 :::111                :::*                     LISTEN      1284/rpcbind
tcp        0      0 :::22                 :::*                     LISTEN      1573/sshd
tcp        0      0 :::1:631              :::*                     LISTEN      1362/cupsd
tcp        0      0 :::1:25               :::*                     LISTEN      1654/master
```

```
[root@mysql mysql]# cat mysql.pid
37860
```

## 3、数据库-原理部分

常用的术语

### 3.1、数据模型

数据模型（Data model）是数据库结构的基础，是用来描述数据的一组概念和定义，数据模型主要有三个要素：数据结构、数据操作、数据约束条件。

数据结构：对象类型的集合，是对静态属性的描述。

数据操作：是对数据库中的各种对象性数据，允许执行的操作的集合，如增删改查等；数据操作是对系统动态热性的描述。

数据的约束条件：是一组完整性规则的集合，也就是说，对于具体的应用必须先遵循特定的语义约束条件。比如：性别只能取“男”或者“女”中的之一。考试成绩：（满分100）只能是0-100的数值。

## 3.2、数据库

数据库（database）是长期存储在计算机外存上有结构，可共享的数据集合；数据库中的数据按照一定的数据模型描述、组织和存储，具有较小的冗余度，较高的数据独立性和可扩展性，并可以为多个用户共享。

常见数据库分类：

关系型数据库：如：MySQL, oracle, sqlserver

非关系型数据库：如：redis, hbase等

## 3.3、数据库管理系统

数据库管理系统（database management system, DBMS）是指数据库系统对数据库进行管理的软件系统，是数据库系统的核心组成部分。数据库的一切操作，如增删改查以及各种控制，都是通过DBMS进行的。

具有以下4个基本的功能：

### 1、数据定义功能

用户可以通过DBMS提供的数据库定义语言对数据库的数据进行定义。

### 2、数据操纵功能

用户可以通过数据操纵语言实现对数据库的增删改查操作

### 3、数据库运行管理

管理数据库的运行是DBMS运行时的核心工作。所有访问数据库的操作都要在DBMS的统一管理下进行，以保证数据的安全性、完整性、一致性以及多用户对数据库的并发使用。

## 4、数据库的建立和维护

建立数据库，包括数据库初始数据的输入与数据转换等。维护数据库，包括数据库的转储与恢复，数据库的重组织，性能监控和分析。

## 3.4、数据库系统相关人员

数据库系统的相关人员是数据库系统的重要组成部分，具体可以分为以下的三类人员

### 1、数据库管理员

职责：负责数据库的建立、使用、维护的专门人员

### 2、应用程序开发人员

职责：开发数据库应用程序的人员，可以使用数据库管理系统的所有功能。

### 3、最终用户

职责：一般来说，是通过应用程序使用数据库的人员，最终用户无需自己编写应用程序。

## 3.5、数据库系统

数据库系统（database system DBS）是由硬件系统，数据库管理系统，数据库，数据库应用程序，数据库系统相关人员构成的人-机系统，是指有数据库的整个计算机系统。

说明：在许多场合下，数据库，数据库管理系统，数据库系统不做严格区分；

## 4、关系代数分类

### 4.1、基本运算

并、差、笛卡尔积、选择、投影；

关系代数的基本操作（原始运算）：“选择”、“投影”、笛卡尔积（也叫做“叉积”或“交叉连接”）、并集、差集和“重命名”。

## 4.2、组合运算

交、连接、自然连接和除；

其中最重要的是交集、除法和自然连接。(inner join [join])

## 4.3、扩充的关系代数操作

外连接（左外和右外）、外部并和半连接；(left join ,right join)

# 5、数据库设计

关系型数据库的设计分为以下5个阶段：

## 5.1、需求分析

(1) 明确用户需求，到底做什么？

## 5.2、概念模式设计

(1) 该阶段是整个数据库设计的关键，它通过对用户需求进行综合、归纳与抽象。主要是通过E-R图表示。

(2) 优点：

A、简单明了，容易理解

B、独立于计算机与具体的RDBMS无关。

(3) E-R模型的基本元素

A、实体（Entity）如：学生

B、属性（attribute）如：姓名

C、键码（key）如：身份证号码；

D、关系（relationship）如：两个实体之间的关系

- a) 一对一 (1:1) : 一个人一个身份证号码; 一个学校一个校长
- b) 一对多 (1: n) : 学校和老师的关系
- c) 多对多 (n: n) : 学生选课, 一个学生可以选择多门课程, 一门课程被多名学生进行选修。

E、E-R符号表示:

## 5.3、逻辑模式设计

(1) 该阶段会涉及到更多的概念, 方法, 理论。

(2) 主要任务:

A、与具体的数据库相关

B、规范化处理, 尽可能的消除关系操作过程中的异常情况。

C、E-R图转换为如下的关系模式

电影 (片名, 出品年份, 影片长度, 影片类型, 公司名称)

明星 (姓名, 联系地址, 公司名称)

扮演 (片名, 出品年份, 姓名, 角色)

影片公司 (公司名称, 地址)

卡通片 (片名, 出品年份, 设计平台)

## 5.4、数据库实施

(1) 创建数据库, 定义数据库结构, 组织数据入库, 调试数据库并进行数据库的试运行。

## 5.5、数据库的运行和维护



(1) 数据库正式运行之后，对数据库运行过程中对其进行评价，调整，修改，调优等。

## 6、数据库设计遵循的原则

### 6.1、范式概念

概念：范式就是符合某一规范级别的关系模式的集合。共有7种范式：

1NF  $\supset$  2NF  $\supset$  3NF  $\supset$  BCNF  $\supset$  4NF  $\supset$  5NF  $\supset$  6NF

#### 6.1.1、第一范式（First Normal Form）

如果一个关系模式R的所有属性都是不可分割的基本数据项，则这个关系属于第一范式。

举例说明：(学生选课：学号，姓名，系别，系部地址，课程名称，课程成绩)

Student(s\_no,s\_name,s\_dept,s\_location,s\_course\_name,s\_grade)

注：1NF是关系模式应具备的最起码的条件，如果数据库设计不能满足第一范式，就不能称作是关系模式；关系数据库设计研究的关系规范化是在1NF基础之上进行的。

#### 6.1.2、第二范式（Second Normal Form）

定义：若关系模式R属于第一范式，且每个非主属性都是完全函数依赖于主键，则R属于第二范式。

说明：从2NF的定义可以看出，从2NF开始讨论的是主键和非主属性之间的函数依赖关系，所以分析关系模式是属于2NF，首先指明关系模式的主键，然后在讨论非主属性和主键之间的函数依赖关系。

例如：选课关系模式

SC (s\_no,c\_no,score) 中，主键为 (s\_no,c\_no)，而非主属性score与主键之间不存在部分函数依赖关系，所以关系模式SC属于2NF

### 6.1.3、第三范式 (Third Normal Form)

定义：若关系模式R属于第一范式，且每个非主属性都不传递函数依赖于主键，则R属于第三范式。

说明：3NF说明的是非主属性和主键之间的函数依赖关系

例如：选课关系模式

SC (s\_no,c\_no,score) 中,由于除了主键之外，只有一个非主属性score，所以score不可能构成与主键之间的传递函数依赖，所以SC属于3NF

### 6.1.4、BCNF (Boyce-Codd Normal Form)

定义：若关系模式R属于第一范式，且每个属性都不传递依赖于主键，则R属于BC范式。

说明：也就是说，在关系模式R中，凡是决定因素的属性或属性集包含键码，决定因素是函数依赖的左部属性集，比如 $X \rightarrow Y$ ，X称为决定因素。

由BC范式的定义可以得到以下的结论，一个满足BC范式的关系模式有：

- 1、所有非主属性对每一个候选码都是完全函数依赖
- 2、所有的主属性对每一个不包含它的候选码都是完全函数依赖
- 3、没有任何属性完全函数依赖于非候选码的任何一组属性

## 7、事务并发操作出现几种问题

所谓事务，是用户定义的一个数据库操作序列，是数据库环境中的逻辑工作单元，是一个不可分割的整体。

事务的这个4个特性简称为ACID特性，事务ACID特性可能遭到破坏的因素有：

- ①多个事务并发执行，不同事务的操作交叉执行；
- ②事务在运行过程中被强行终止。

如何保证在多个事务并发执行的过程中不发生上述的两种情况，是数据库管理系统并发控制的主要责任。

## 7.1、丢失修改数据

举例：银行卡有100元，事务A取10元，事务B取10元，事务AB两人同时取钱，初始值都是100

## 7.2、读“脏”数据

数据库技术中，如果正常提交的事务A使用了事务B未提交的撤销数据，这种数据成为“脏数据”，会造成数据的脏读和污读。

## 7.3、不一致分析

造成这种数据不一致的主要原因是并发执行的两个事务中，一个事务在读取数据时，另一个事务正在修改同一个数据。这样就可能导致两个事务的相互干扰及“读”事务的错误执行结果。

# 8、数据库并发的控制（了解部分）

## 8.1、并发调度的可串行化

可串行化准则：多个事务的并发执行时正确的，当且仅当其结果按某一次序串行执行它们时的结果相同，这种调度策略称为可串行化调度。可串行化是并发事务正确性的准则，一个给定的并发调度，当且仅当它是可串行化的，才认为是正确的。

## 8.2、封锁

封锁是实现并发控制的非常重要的技术。封锁是指某事务在对某数据对象进行操作以前，先请求系统对其加锁，成功加锁之后该事务就对该数据对象有了控制权，只有该事务对其进行解锁之后，其他的事务才能更新它，DBMS有两种锁：

### ① 排它锁（也称作X锁）

如果事务T在对某个数据对象实施了X锁，那么其他的事务必须要等到T事务接触对该数据对象的X锁之后，才能对这个数据进行加锁。

### ② 共享锁（也称作S锁）

如果事务T在对某个数据对象实施了S锁，那么其他的事务也能对该数据对象实施S锁，但是对这个数据对象施加的所有S锁都接触之前不允许任何事务对该数据对象实施X锁。

## 8.3、死锁

封锁技术可以避免一些并发操作引起的不一致错误，但也会产生其他的一些问题，活锁和死锁。

### ① 活锁

如果某个事务处在永远等待的状态，得不到封锁的机会，这种现象为活锁，避免这种锁最好的方法就是采用先来先服务的策略。

### ② 死锁

两个或两个以上的事务都处于等待状态每个事务都在等待对方事务接触封锁，它才能继续执行下去，这样任何事务都处于等待状态而无法继续执行的现象称为死锁

解决死锁问题方法有两类：

A、死锁的预防

B、死锁的诊断与预防



黑马程序员  
www.itheima.com

传智播客旗下  
高端IT教育品牌

改变中国IT教育，我们正在行动

传智播客