

第六章（1）：索引

6.1：理解索引

索引是由数据库表中一列或多列组合而成的一种特殊的数据库结构，利用索引是可以用来快速查询数据库表中的特定记录信息。在MySQL中，所有的数据类型都可以被索引。

1. 理解索引

MySQL的索引是为了加速对数据进行检索而创建的一种分散的、物理的数据结构。索引包含从表或视图中一个或多个列生成的键，以及映射到指定数据行的存储位置指针。

索引是依赖于表建立的，提供了数据库中编排表中数据的内部方法。表的存储由两部分组成，一部分是表的数据页面，另一部分是索引页面。索引就存放在索引页面上。

1. 理解索引

数据库中的索引形式与图书的目录相似，键值就像目录中的标题，指针相当于页码。索引的功能就像图书目录能为读者提供快速查找图书页面内容一样，不必扫描整个数据表而找到想要的数据行。

当MySQL数据库在执行一条查询语句的时候，默认的执行过程是根据搜索条件进行全表扫描，遇到匹配条件的就加入搜索结果集合。如果查询语句涉及多个表连接，包括了许多搜索条件（例如大小比较、like匹配等），而且表数据量特别大的时，在没有索引的情况下，MySQL需要执行的扫描行数会很大，速度也会很慢。

索引一旦创建，将由数据库自动管理和维护。例如，向表中插入、更新和删除一条记录时，数据库会自动在索引中做出相应的修改。在编写SQL查询语句时，具有索引的表与不具有索引的表没有任何区别，索引只是提供一种快速访问指定记录的方法。实际过程中，当MySQL执行查询时，查询优化器会对可用的多种数据检索方法的成本进行估计，从中选用最有效的查询计划。

在数据库中使用索引的优点如下。

- (1) 加速数据检索：索引能够以一系列或多列值为基础实现快速查找数据行。
- (2) 优化查询：查询优化器是依赖于索引起作用的，索引能够加速连接、排序和分组等操作。
- (3) 强制实施行的唯一性：通过给列创建唯一索引，可以保证表中的数据不重复。

需要注意的是，索引并不是越多越好，要正确认识索引的重要性和设计原则，创建合适的索引。

6.2：索引的分类

索引的分类

按照分类标准的不同，MySQL的索引有多种分类形式。MySQL的索引通常包括普通索引（index）、唯一性索引（unique）、主键（primary key）、全文索引（[fulltext](#)）和空间索引（spatial）等类型。

(1) 普通索引（index）。索引的关键字是index。普通索引是MySQL中的基本索引类型，允许在定义索引的列中插入重复值和空值。

(2) 主键索引（primary key）。是一种特殊的唯一索引，不允许有空值。一般是在建表的时候同时创建主键索引。也可通过修改表的方法增加主键，但一个表只能有一个主键索引。

(3) 唯一性索引 (unique)。unique索引列的值必须唯一，允许有空值。如果是组合索引，则列值的组合必须唯一。在一个表上可以创建多个unique索引。

(4) 全文索引 (fulltext)。全文索引是指在定义索引的列上支持值的全文查找，允许在这些索引列中插入重复值和空值。该索引只能对char、varchar和text类型的列编制索引，并且只能在MyISAM表中编制。即MySQL中只有MyISAM存储引擎支持全文索引。在MySQL默认情况下，对于中文作用不大。

(5) 空间索引 (spatial)。空间索引是对空间数据类型的字段建立的索引。MySQL中的空间数据类型有4种，分别是geometry、point、linestring和polygon。MySQL使用spatial关键字进行扩展，使得能够用于创建正规索引类似的语法创建空间索引。创建空间索引的列，必须将其声明为not null，空间索引只有在存储引擎MyISAM的表中创建。对于初学者来说，这类索引很少会用到。

- 如果按照创建索引键值的列数分类，索引还可以分为单列索引和复合索引。
- 如果按照存储方式分类，MySQL的索引分为B-Tree索引和Hash索引。

MySQL的MEMORY数据引擎还支持Hash索引。Hash索引相对于B-Tree索引，检索效率要高上不少。但MySQL Hash索引本身的特殊性也带来了许多限制和弊端，主要有以下内容。

- 仅仅能满足“=”，“in”和“<=>”查询，不能使用范围查询。
- 无法被用来避免数据的排序操作。
- 无法被用来避免数据的排序操作。
- 不能利用部分索引键查询。
- 在任何时候都不能避免表扫描。
- 遇到大量Hash值相等的情况后，性能并不一定会比B-Tree索引高。

说明：

- MySQL中运算符“<=>”除了能够像常规的“=”运算符一样，对两个值进行比较外，还能够用于比较null值。
- 运算符“<=>”和“=”号的相同点。像常规的“=”运算符一样，两个值进行比较，结果是0（不相等）或1（相等）。例如，'A'<=>'B'的值为0，'a'<=>'a'的值为1。
- 运算符“<=>”和“=”号的不同点。和“=”运算符不同的是，null与常量进行比较运算，其值直接处理为null。使用“<=>”运算符时，例如，'a'<=>null的值为0，null<=>null的值为1；相当于'a' is null和null is null。而'a' is not null则相当于 not('a'<=>null)。

6.3：设置索引的原则

在数据表中创建索引，为使索引的使用效率更高，必须考虑在哪些字段上创建索引和创建什么类型的索引。首先要了解以下常用的基本原则。

- (1) 一个表创建大量索引，会影响insert、update和delete语句的性能。应避免对经常更新的表创建过多的索引，要限制索引的数目。
- (2) 若表的数据量大，对表数据的更新较少而查询较多，可以创建多个索引来提高性能。在包含大量重复值的列上创建索引，查询的时间会较长。
- (3) 经常需要排序、分组和联合操作的字段一定要建立索引，即将用于join、where判断和order by排序的字段上创建索引。

- (4) 在视图上创建索引可以显著的提升查询性能。
- (5) 尽量不要对数据库中某个含有大量重复的值的字段建立索引，在这样的字段上建立索引有可能降低数据库的性能。
- (6) 在主键上创建索引，在InnoDB中如果通过主键来访问数据效率是非常高的。每个表只能创建一个主键索引。
- (7) 要限制索引的数目，对于不再使用或者很少使用的索引要及时删除。
- (8) InnoDB数据引擎的索引键最长支持是767字节，MYISAM数据引擎支持1000字节。

6.4：创建索引

4. 创建索引

- 创建索引通常有3种命令方式，即创建表时附带创建索引、通过修改表来创建索引和使用alter table语句来创建索引。当然利用MySQL Workbench等工具也可以实现可视化方式创建索引。下面将详细讲解这些创建索引的方法。
- **利用create index语句创建3种索引。**如果基表已经创建完毕，就可以使用create index语句建立索引。
- 创建索引基本形式如下。

```
create [unique|fulltext|spatial] index index_name
on table_name (index_col_name,...)
```
- 创建索引时，可以使用col_name(length)语法对前缀编制索引。前缀包括每列值的前length个字符。对于char和varchar列，只用1列的一部分就可创建索引。blob和text列也可以编制索引，但是必须给出前缀长度。
- 因为多数名称的前10个字符通常不同，所以前缀索引不会比使用列的全名创建的索引速度慢很多。另外，使用列的一部分创建索引可以使索引文件大大减小，从而节省了大量的磁盘空间，有可能提高insert操作的速度。

【例1】为便于按电话进行查询，为student表的phone列上建立一个升序普通索引phone_index。

```
mysql> use teaching;
mysql> create index phone_index on student(phone asc);
```

【例2】在course表的cname列上建立一个唯一性索引cname_index。

```
mysql> create unique index cname_index on course (cname);
```

【例3】在score表的studentno和courseno列上建立一个复合索引sc_index。

```
mysql> create index sc_index on score(studentno,courseno);
```

- 创建表时创建索引。创建表时可以直接创建索引，这种方式最简单、方便。
- 【例4】创建teacher1表的tname字段建立一个唯一性索引tname_index，一个前缀索引dep_index。
- `mysql> use mysqltest;`
- `mysql> create table if not exists teacher1 (`
- `-> teacherno char(6) not null comment '教师编号',`
- `-> tname char(8) not null comment '教师姓名',`
- `-> major char(10) not null comment '专业',`
- `-> prof char(10) not null comment '职称',`
- `-> department char(16) not null comment '部门',`
- `-> primary key (teacherno),`
- `-> unique index tname_index(tname),`
- `-> index dep_index(department(5))`
- `->);`
- 通过alter table语句创建索引。

【例5】在teacher1表上建立teacherno主键索引（假定未创建主键索引），建立tname和prof的复合索引。

```
mysql> alter table teacher1
-> add primary key(teacherno),
-> add index mark(tname, prof);
```

如果主键索引已经创建，则会出现如下信息：

ERROR 1068 (42000): Multiple primary key defined

说明：

- (1) 只有表的所有者才能给表创建索引。索引的名称必须符合MySQL的命名规则，且必须是表中唯一的。
- (2) 主键索引必定是唯一的，唯一性索引不一定是主键。一张表上只能一个主键，但可以有一个或者多个唯一性索引。
- (3) 当给表创建unique约束时，MySQL会自动创建唯一索引。创建唯一索引时，应保证创建索引的列不包括重复的数据，并且没有两个或两个以上的空值（null）。因为创建索引时将两个空值也视为重复的数据，如果有这种数据，必须先将其删除，否则索引不能被成功创建。
- (4) 若要查看表中已经创建索引的情况，可以使用show index from table_name语句实现。

6.5：删除索引

5. 删除索引

- 删除不再需要的索引，可以通过drop语句来删除索引，也可用alter table语句删除。利用drop index语句删除索引的语法格式如下：
- ```
drop index index_name on table_name;
```
- 例如，删除teacher1表的mark索引。
- ```
mysql> drop index mark on teacher1;
```

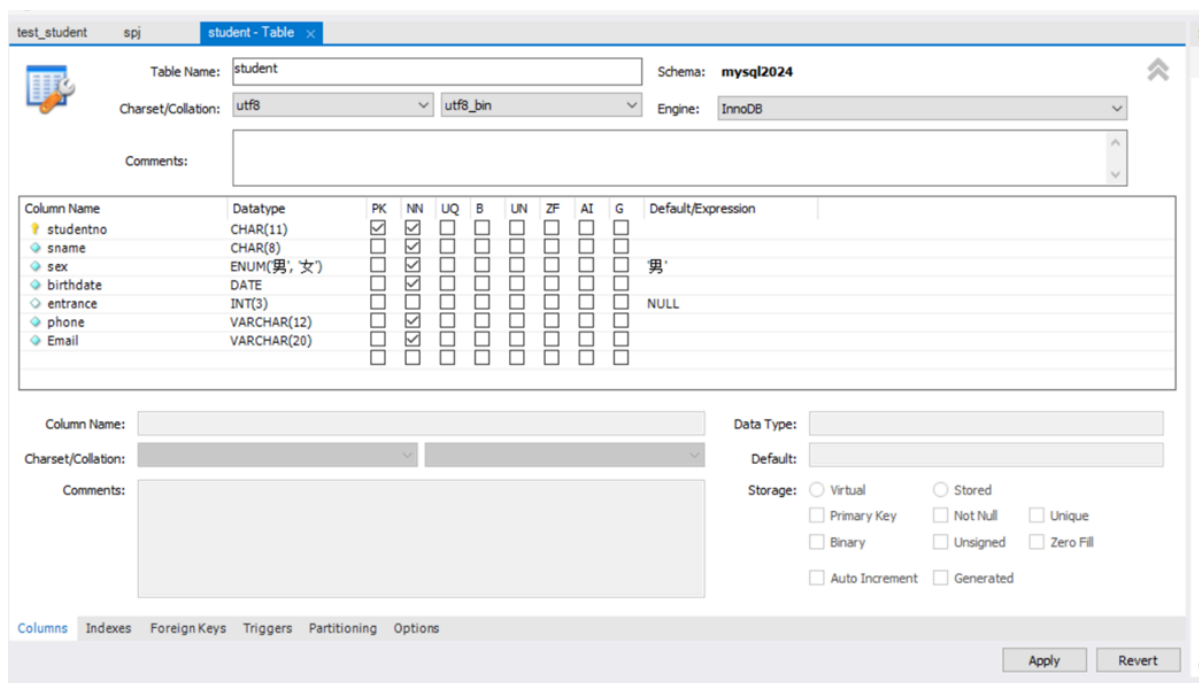

- 利用alter table语句删除索引的语法格式如下：
 alter [ignore] table table_name
 | drop primary key
 | drop index index_name
 | drop foreign key fk_symbol
- 利用alter table语句同样可以删除前面表中创建的索引。例如：
 mysql> alter table course drop index cname_index;

6.6 : 利用MySQL Wokrbench工具创建和管理索引

6.利用MySQL Workbench工具创建和管理索引

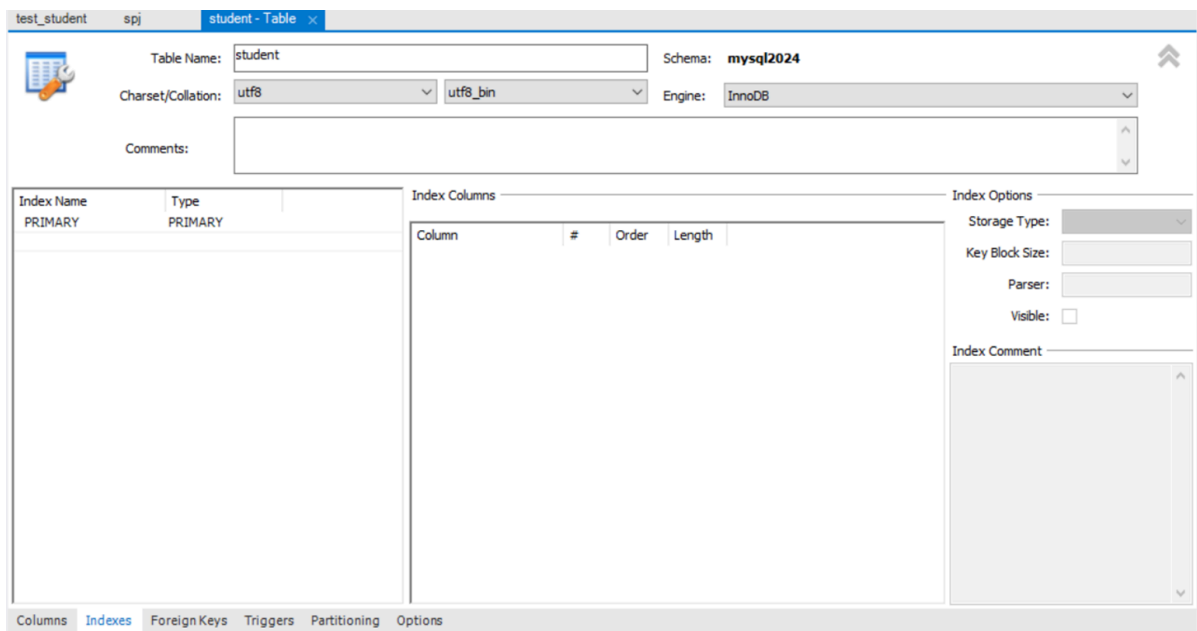
- 利用MySQL Workbench创建索引

(1) 启动MySQL Workbench工具，单击数据库mysql2024，选择tables，展开tables选项，选择表student，右击选择执行Alter Tables命令，如图所示。



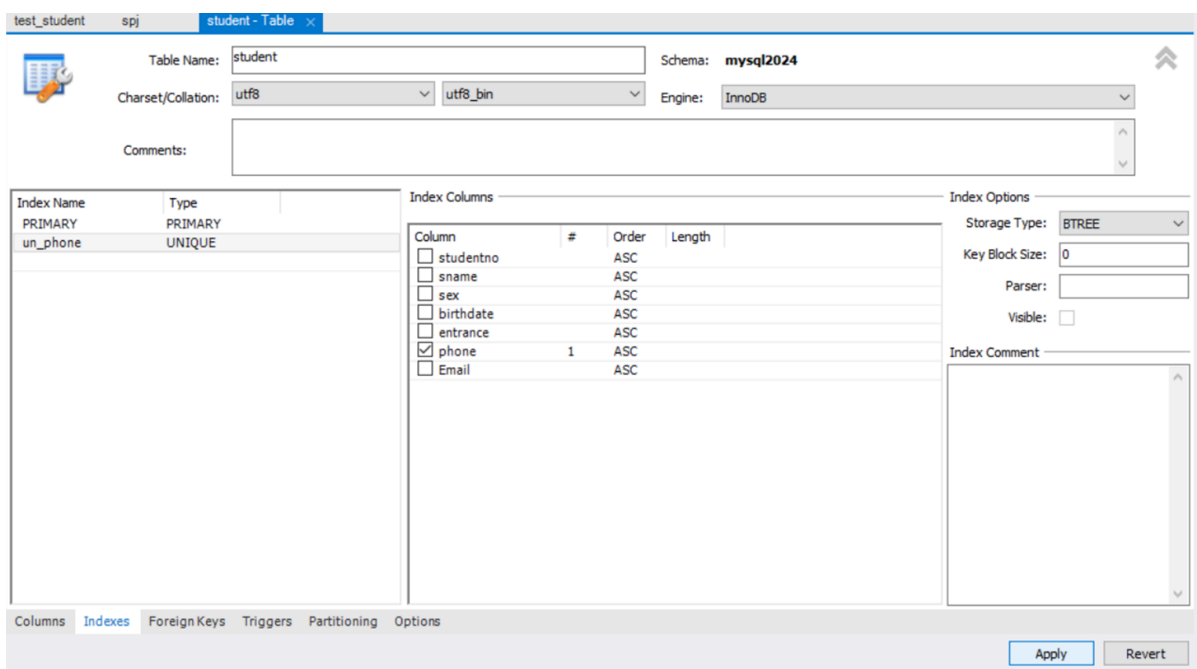
(3) 进入修改表student界面。选择index选项卡，如图所示的界面中，可以观察到如下信息。

- 数据库名和表名：指出创建索引的数据库teaching和表student的名称。
- 表student的默认字符集及排序规则和数据引擎InnoDB索引名称。
- 索引名Index Name，可以查看到前面创建的主键索引primary和唯一索引phone_index。其后依次是索引类型Type、索引引用字段Index Columns、索引参数Index Options和索引注解Index Comment等。



(4) 在Index Name的文本框中输入索引名称`un_phone`，右侧的Index Columns会自动显示表student中的所有列名，选择phone列。存储类型选择BTREE，选择索引类型unique，表示创建唯一性索引，其他参数采用默认值。

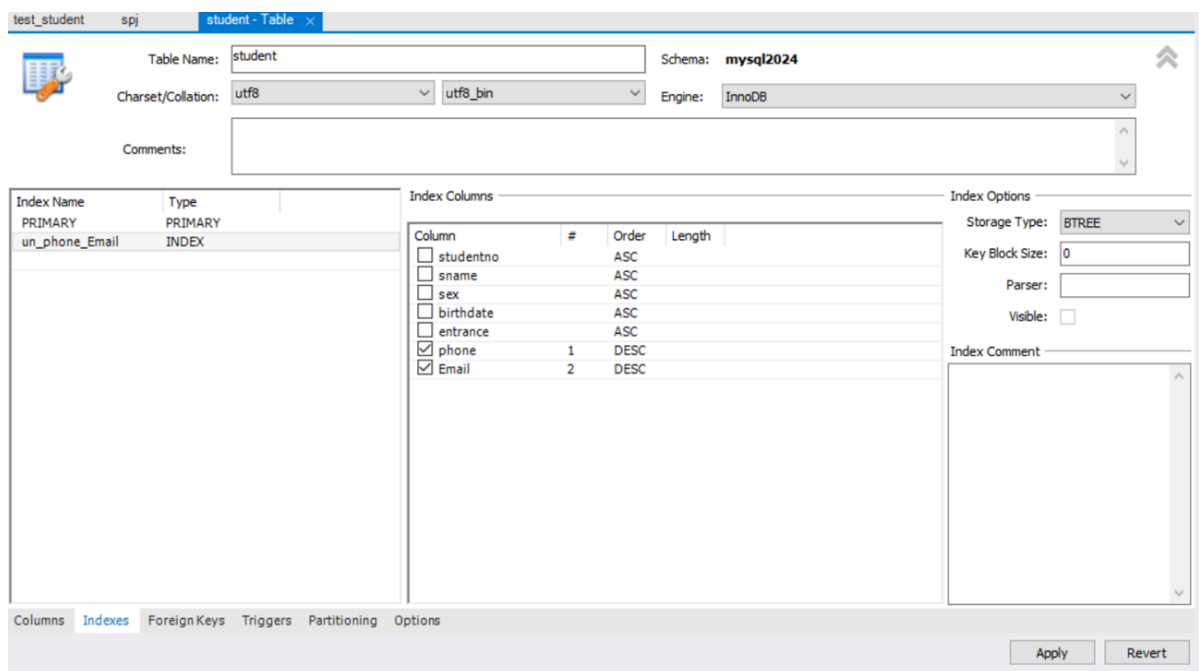
(5) 设置完成后，单击Apply按钮，出现应用脚本对话框。再单击Apply按钮，进入完成对话框，单击finish按钮，即可完成student表上的唯一索引`un_phone`的创建。



• 利用MySQL Workbench管理索引

(1) 利用MySQL Workbench修改索引，可以修改索引的名字、类型、索引引用字段和索引参数等。

- 修改student表中的`un_phone`索引为普通索引`un_phone_Email`，索引类型改为index，引用字段为phone和Email，且为降序排列。
- 单击Apply按钮，出现如图6-8所示的应用脚本对话框。
- 再单击Apply按钮，进入完成对话框如图6-9所示，单击Show Logs (Hide Logs)，可以查看(隐藏)日志消息。
- 单击finish按钮，即可完成在数据库teaching中student表上的索引`un_phone`的修改。



(2) 利用MySQL Workbench删除索引。

- 删除普通索引un_phone_Email。在索引界面中，右击索引un_phone_Email，执行Delete Selected命令，索引un_phone_Email即从列表中消失。
- 单击单击 Apply按钮，出现删除索引的应用脚本对话框。
- 再单击 Apply按钮，进入完成对话框。
- 单击finish按钮，即可删除索引un_phone_Email。