

## 3. 表的操作

---

### 3.1 创建表

语法：

```
CREATE TABLE table_name (  
    field1 datatype,  
    field2 datatype,  
    field3 datatype  
) character set 字符集 collate 校验规则 engine 存储引擎;
```

说明：

- field 表示列名
- datatype 表示列的类型
- character set 字符集，如果没有指定字符集，则以所在数据库的字符集为准
- collate 校验规则，如果没有指定校验规则，则以所在数据库的校验规则为准

### 3.2 创建表案例

```
create table users (  
    id int,  
    name varchar(20) comment '用户名',  
    password char(32) comment '密码是32位的md5值',  
    birthday date comment '生日'  
) character set utf8 engine MyISAM;
```

说明：

- 不同的存储引擎，创建表的文件不一样。  
users 表存储引擎是 MyISAM，在数据目中有三个不同的文件，分别是：
  - users.frm：表结构
  - users.MYD：表数据
  - users.MYI：表索引

此电脑 > 系统 (C:) > ProgramData > MySQL > MySQL Server 5.7 > Data > test1

名称	修改日期	类型	大小
 db.opt	2018/10/24 21:36	StarUML Option	1 KB
 person.frm	2018/10/24 21:36	FRM 文件	9 KB
 person.ibd	2018/10/24 21:36	IBD 文件	96 KB
 t1.frm	2018/10/25 15:09	FRM 文件	9 KB
 t1.ibd	2018/10/25 15:10	IBD 文件	96 KB
 users.frm	2018/10/29 11:13	FRM 文件	9 KB
 users.MYD	2018/10/29 11:13	MYD 文件	0 KB
 users.MYI	2018/10/29 11:13	MYI 文件	1 KB

备注：创建一个engine是innodb的数据库，观察存储目录

### 3.3 查看表结构

`desc` 表名；

示例：

mysql> desc users;						索引类型
+-----+-----+-----+-----+-----+-----+						↓
Field	Type	Null	Key	Default	Extra	
+-----+-----+-----+-----+-----+-----+						
id	int(11)	YES		NULL		
name	varchar(20)	YES		NULL		
password	char(32)	YES		NULL		
birthday	date	YES		NULL		
+-----+-----+-----+-----+-----+-----+						
↑	↑	↑		↑	↑	
字段名字	字段类型	是否允许为空		默认值	扩充	

### 3.4 修改表

在项目实际开发中，经常修改某个表的结构，比如字段名字，字段大小，字段类型，表的字符集类型，表的存储引擎等等。我们还有需求，添加字段，删除字段等等。这时我们就需要修改表。

```
ALTER TABLE tablename ADD (column datatype [DEFAULT expr][,column datatype]...);
```

```
ALTER TABLE tablename MODIfy (column datatype [DEFAULT expr][,column datatype]...);
```

```
ALTER TABLE tablename DROP (column);
```

案例：

- 在users表添加二条记录

```
mysql> insert into users values(1,'a','b','1982-01-04'),(2,'b','c','1984-01-04');
```

- 在users表添加一个字段，用于保存图片路径

```
mysql> alter table users add assets varchar(100) comment '图片路径' after birthday;
```

```
mysql> desc users;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	YES		NULL	
name	varchar(20)	YES		NULL	
password	char(32)	YES		NULL	
birthday	date	YES		NULL	
assets	varchar(100)	YES		NULL	

插入新字段后，对原来表中的数据没有影响：

```
mysql> select * from users;
```

id	name	password	birthday	assets
1	a	b	1982-01-04	NULL
2	b	c	1984-01-04	NULL

<= 原来的数据仍然存在

- 修改name，将其长度改成60

```
mysql> alter table users modify name varchar(60);
```

```
mysql> desc users;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	YES		NULL	
name	varchar(60)	YES		NULL	<= 长度变成60
password	char(32)	YES		NULL	
birthday	date	YES		NULL	
assets	varchar(100)	YES		NULL	

- 删除password列

注意：删除字段一定要小心，删除字段及其对应的列数据都没了

```
mysql> alter table users drop password;
mysql> desc users;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	YES		NULL	
name	varchar(60)	YES		NULL	
birthday	date	YES		NULL	
assets	varchar(100)	YES		NULL	

- 修改表名为employee

```
mysql> alter table users rename to employee;
```

```
mysql> select * from employee;
```

id	name	birthday	assets
1	a	1982-01-04	NULL
2	b	1984-01-04	NULL

**to:** 可以省掉

- 将name列修改为xingming

```
mysql> alter table employee change name xingming varchar(60); --新字段需要完整定义
```

```
mysql> desc employee;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	YES		NULL	
xingming	varchar(60)	YES		NULL	
birthday	date	YES		NULL	
assets	varchar(100)	YES		NULL	

## 3.5 删除表

语法格式:

```
DROP [TEMPORARY] TABLE [IF EXISTS] tbl_name [, tbl_name] ...
```

示例:

```
drop table t1;
```