mysql 学习笔记

几个dos命令：

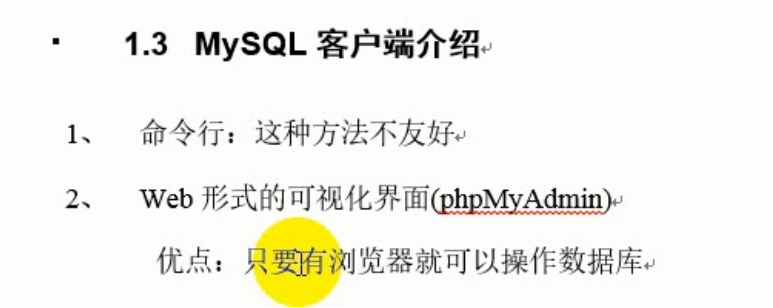
1、进入盘符：进入D盘， D:

2、进入文件夹： cd filename

3、返回上一级目录： cd ../

4、返回根目录： cd /

/ ：表示根目录。



安装好数据库后，MySQL**自带一个MySQL命令行终端**，它很方便，但它**只能连接本地的MySQL。**
我们用**Windows命令行**连接数据库服务器。

连接数据库

进入MySQL 的**bin**目录

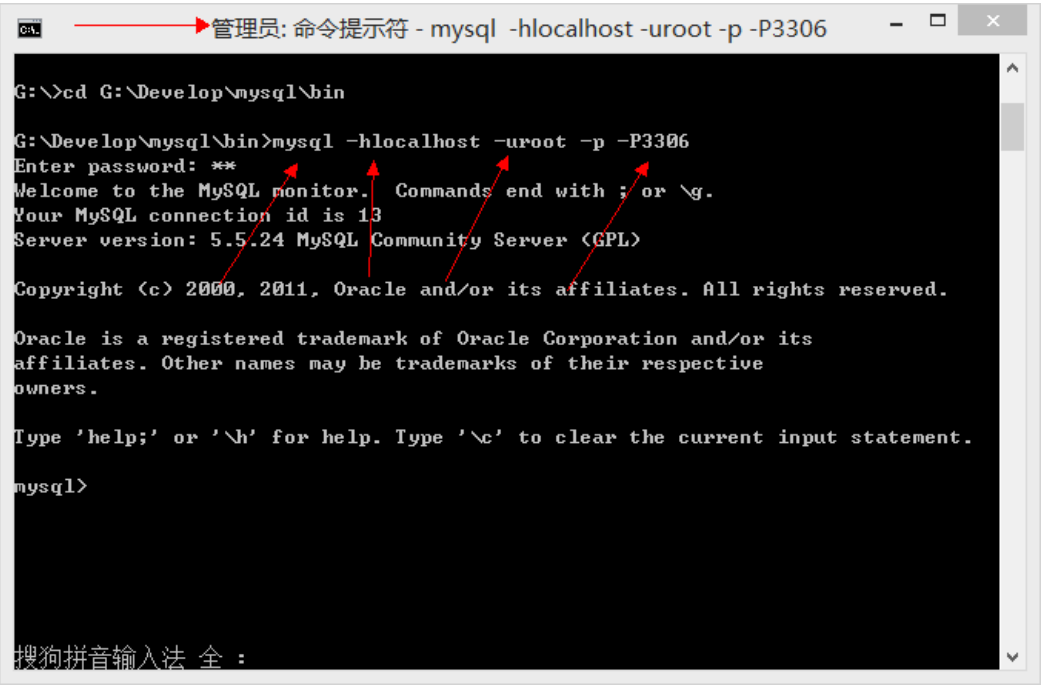
连接MySQL服务器需要的参数：

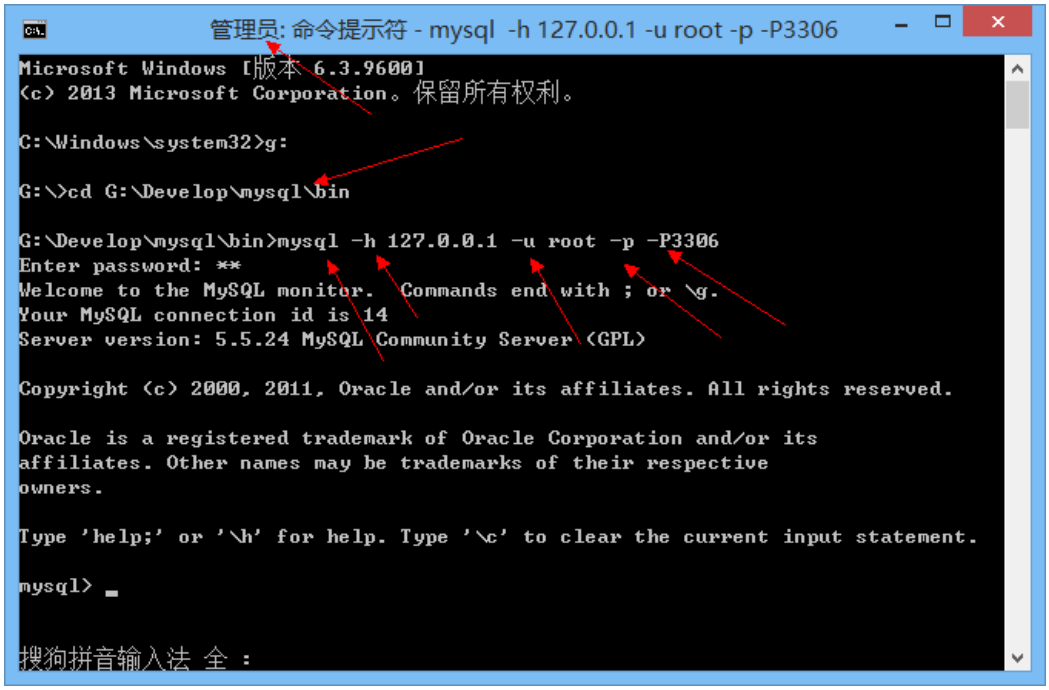
Host: -h

Username: -u

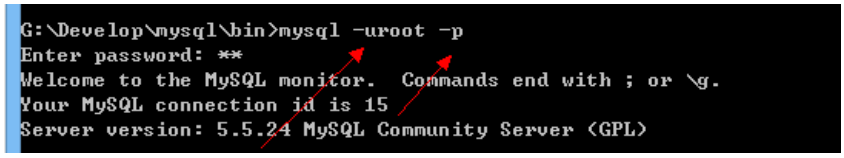
Password : -P (小写p)

Port: -p (大写p)

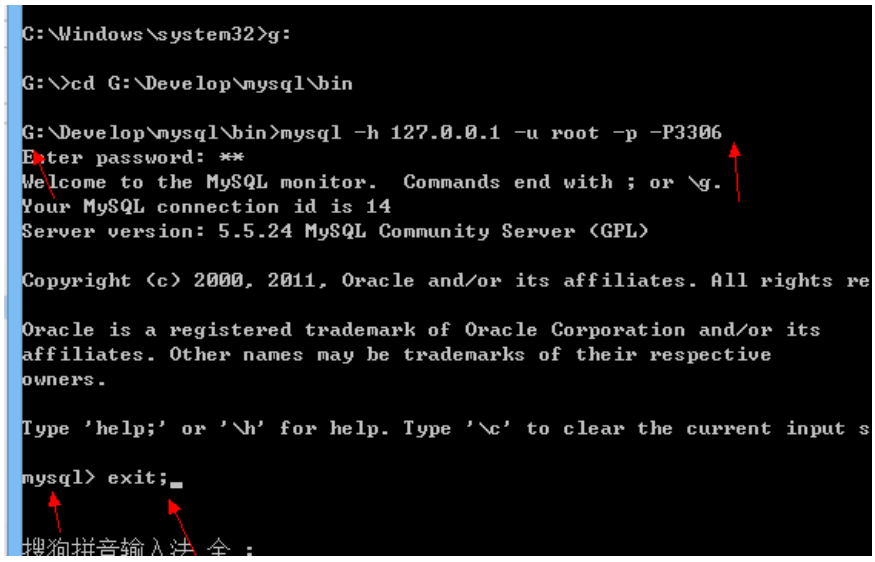




若默认端口是3306和默认的host是本机localhost，则可以省略，只写 **用户名和密码**，也可连接数据库



dos命令句子结尾不用加分号（;），MySQL命令句子结尾要加分号（;）



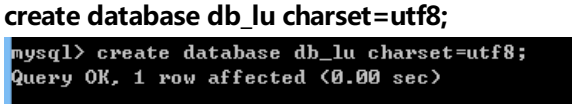
退出数据库： exit; 或 quit; 或 \q;

数据库操作

创建数据库：

创建数据库的时候指定编码：（数据库里的utf-8不用加-，即utf8）

create database db_lu charset=utf8;



create database dbname;

```
mysql> create database db_lu;
Query OK, 1 row affected (0.01 sec)
```

```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| db1                |
| db_lu              |
| mydb               |
| mysql              |
| performance_schema |
| php0507            |
| php34              |
| tempdb2            |
| test               |
+--------------------+
10 rows in set (0.00 sec)
```

```
mysql> create database lu_test;
Query OK, 1 row affected (0.00 sec)

mysql> create database lu_test;
ERROR 1007 (HY000): Can't create database 'lu_test'; database exists
```

不能重复创建
想要创建的数据库名已存在

create database if not exists lu_test;

```
mysql> create database if not exists lu_test;
Query OK, 1 row affected, 1 warning (0.00 sec)
```

要创建的库名是MySQL关键字时，可以用**反引号**包括起来（不是单引号，是键盘1的左边，波浪线这个按键 ）

```
mysql> create database `create`;
Query OK, 1 row affected (0.00 sec)

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| create             |
| db1                |
| lu_test            |
| mydb               |
| mysql              |
| performance_schema |
| php0507            |
| php34              |
| tempdb2            |
| test               |
+--------------------+
11 rows in set (0.00 sec)
```

```
mysql> drop database `create`;
Query OK, 0 rows affected (0.00 sec)
```

删除`create`数据库

查询数据库：

show databases;

```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| db1                |
| db_lu              |
| mydb               |
| mysql              |
| performance_schema |
| php0507            |
| php34              |
| tempdb2            |
| test               |
+--------------------+
10 rows in set (0.00 sec)
```

显示数据库的创建语句：（可以查看到所用的字符编码）

show create database lu_test;

```
mysql> show create database lu_test;
+----------+-----------------------------------------------------------------+
| Database | Create Database                                                 |
+----------+-----------------------------------------------------------------+
| lu_test  | CREATE DATABASE `lu_test` /*!40100 DEFAULT CHARACTER SET utf8 */ |
+----------+-----------------------------------------------------------------+
1 row in set (0.00 sec)
```

更改数据库：

更改数据库的字符编码：

alter database db_lu charset=gbk;

```
mysql> alter database db_lu charset=gbk;
Query OK, 1 row affected (0.00 sec)

mysql> show create database db_lu;
+----------+-----------------------------------------------------------------+
| Database | Create Database                                                 |
+----------+-----------------------------------------------------------------+
| db_lu    | CREATE DATABASE `db_lu` /*!40100 DEFAULT CHARACTER SET gbk */ |
+----------+-----------------------------------------------------------------+
1 row in set (0.00 sec)
```

删除数据库

drop database if exists db_lu;

```
mysql> drop database db_lu;
ERROR 1008 (HY000): Can't drop database 'db_lu'; database doesn't exist
mysql> drop database if exists db_lu;
Query OK, 0 rows affected, 1 warning (0.00 sec)
```

drop database db_lu;

```
mysql> drop database db_lu;
Query OK, 0 rows affected (0.02 sec)

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| db1                |
| mydb               |
| mysql              |
| performance_schema |
| php0507            |
| php34              |
| tempdb2            |
| test               |
+--------------------+
9 rows in set (0.00 sec)
```

已删除db_lu数据库

选择数据库

已存在很多数据库，创建表时要先选择数据库

use lu_test;

```
mysql> use lu_test;
Database changed
mysql> _
```

行也叫**记录**，一行就是一条记录

列也叫**字段**，一列就是一个字段。 字段也叫属性

创建表：

```
语法:
Create table 表名(
字段1  数据类型  [null|not null] [default] [auto_increment] [primary key],
字段2  数据类型
...
)

not null:不为空
default:默认值
auto_increment:自动增长
primary key:主键
```

create table stu(stuid int,stuname varchar(10));

```
mysql> use lu_test;
Database changed
mysql> create table stu(stuid int,stuname varchar(10));
Query OK, 0 rows affected (0.01 sec)
```

```
表名:stu
字段名:
Stuid  int,
Stuname  varchar(10)
```

```
mysql> use data;
Database changed
mysql> create table stu(
    -> stuid int,
    -> stuname varchar(10)
    -> );
Query OK, 0 rows affected (0.06 sec)

mysql>
```

查看表：

show tables;

```
mysql> show tables;
+------------------+
| Tables_in_lu_test |
+------------------+
| stu              |
+------------------+
1 row in set (0.00 sec)
```

显示创建表的SQL语句：

show create table stu \G;

（\G ：表示table字段和create table 字段竖排）

```
mysql> show create table stu \G;
*************************** 1. row ***************************
       Table: stu
Create Table: CREATE TABLE `stu` (
  `stuid` int(11) DEFAULT NULL,
  `stuname` varchar(10) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8
1 row in set (0.00 sec)
```

显示表结构：

desc stu;

```
mysql> desc stu;
+---------+-------------+------+-----+---------+-------+
| Field   | Type        | Null | Key | Default | Extra |
+---------+-------------+------+-----+---------+-------+
| stuid   | int(11)     | YES  |     | NULL    |       |
| stuname | varchar(10) | YES  |     | NULL    |       |
+---------+-------------+------+-----+---------+-------+
2 rows in set (0.01 sec)
```

或

describe stu;

```
mysql> describe stu;
+---------+-------------+------+-----+---------+-------+
| Field   | Type        | Null | Key | Default | Extra |
+---------+-------------+------+-----+---------+-------+
| stuid   | int(11)     | YES  |     | NULL    |       |
| stuname | varchar(10) | YES  |     | NULL    |       |
+---------+-------------+------+-----+---------+-------+
2 rows in set (0.01 sec)
```

删除表：

drop table stu;

```
mysql> drop table stu;
Query OK, 0 rows affected (0.01 sec)

mysql> show tables;
Empty set (0.00 sec)
```

create table stu1(id int, name varchar(10));create table stu2(id int, name varchar(10));create table stu3(id int, name varchar(10));

```
mysql> create table stu1(id int, name varchar(10));create table stu2(id int, nam
e varchar(10));create table stu3(id int, name varchar(10));
Query OK, 0 rows affected (0.01 sec)

Query OK, 0 rows affected (0.01 sec)

Query OK, 0 rows affected (0.02 sec)
```

```
mysql> show tables;
+-------------------+
| Tables_in_lu_test |
+-------------------+
| stu1              |
| stu2              |
| stu3              |
+-------------------+
3 rows in set (0.00 sec)
```

drop table stu1,stu2,stu3;　（同时删除多个表）

```
mysql> drop table stu1,stu2,stu3;
Query OK, 0 rows affected (0.01 sec)

mysql> show tables;
Empty set (0.00 sec)
```

创建复杂表：

```
create table stu(
    id int auto_increment primary key,
    name varchar(10) not null,
    sex char(1) not null,
    `add` varchar(50) default '地址不详',
    score decimal(3,1)
);
```

关键字用 反引号

```
create table stu(
  id int auto_increment primary key,
  name varchar(10) not null,
  sex char(1) not null,
  `add` varchar(50) default 'address',
  score decimal(3,1)
);
```

```
mysql> create table stu(
    ->  id int auto_increment primary key,
    ->  name varchar(10) not null,
    ->  sex char(1) not null,
    ->  `add` varchar(50) default '地址不详',
    ->  score decimal(3,1)
    -> );
ERROR 1067 (42000): Invalid default value for 'add'
mysql>
```

存在中文，编码关系

```
mysql> create table stu(
    ->  id int auto_increment primary key,
    ->  name varchar(10) not null,
    ->  sex char(1) not null,
    ->  `add` varchar(50) default 'address',
    ->  score decimal(3,1)
    -> );
Query OK, 0 rows affected (0.01 sec)
```

show variables like 'char_%';

```
mysql> show variables like 'char_%';
+--------------------------+-----------------------------------+
| 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       | G:\Develop\mysql\share\charsets\ |
+--------------------------+-----------------------------------+
```

```
mysql> desc stu;
+-------+--------------+------+-----+---------+----------------+
| Field | Type         | Null | Key | Default | Extra          |
+-------+--------------+------+-----+---------+----------------+
| id    | int(11)      | NO   | PRI | NULL    | auto_increment |
| name  | varchar(10)  | NO   |     | NULL    |                |
| sex   | char(1)      | NO   |     | NULL    |                |
| add   | varchar(50)  | YES  |     | address |                |
| score | decimal(3,1) | YES  |     | NULL    |                |
+-------+--------------+------+-----+---------+----------------+
5 rows in set (0.01 sec)
```

数据类型：

数据库里面**没有字符串string类型**，只有**字符类型char**

int： 整型

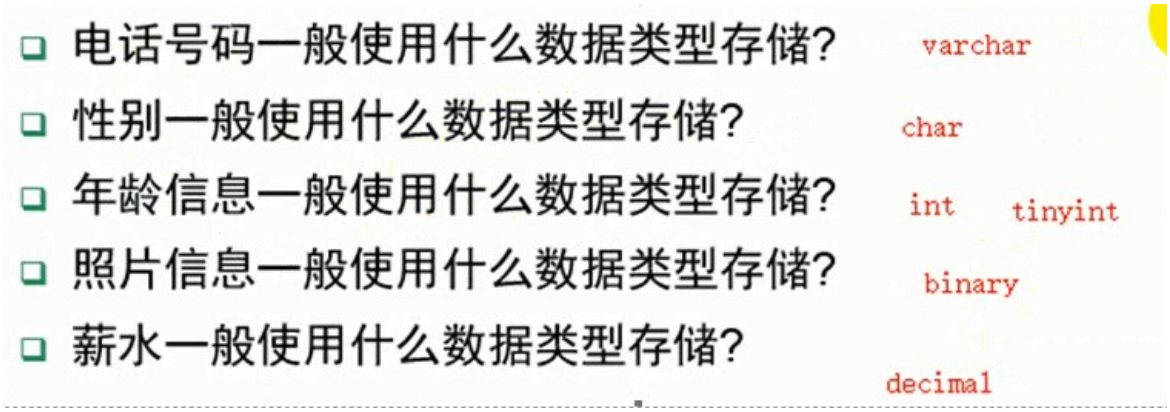
tinyint：微型整数

decimal(总位数，小数位数)：小数

char(10)：定长

varchar(10)：变长

text：大段文本



qq号手机号没有起到数字的作用，不会拿qq手机号作加减运算

数据操作：

插入数据：

```
mysql> desc stu;
+-------+--------------+------+-----+---------+----------------+
| Field | Type         | Null | Key | Default | Extra          |
+-------+--------------+------+-----+---------+----------------+
| id    | int(11)      | NO   | PRI | NULL    | auto_increment |
| name  | varchar(10)  | NO   |     | NULL    |                |
| sex   | char(1)      | NO   |     | NULL    |                |
| add   | varchar(50)  | YES  |     | address |                |
| score | decimal(3,1) | YES  |     | NULL    |                |
+-------+--------------+------+-----+---------+----------------+
```

insert into stu (id,name,sex,`add`) values(1,'libai','m','baijing');

```
mysql> insert into stu (id,name,sex,`add`) values(1,'libai','m','baijing');
Query OK, 1 row affected (0.00 sec)
```

insert into stu(id,name,sex,`add`,score) values(2,'lisi','n','shanghai',99);

```
mysql> insert into stu(id,name,sex,`add`,score) values(2,'lisi','n','shanghai',9
9);
Query OK, 1 row affected (0.00 sec)
```

select * from stu;

```
mysql> select * from stu;
+----+--------+-----+-----------+-------+
| id | name   | sex | add       | score |
+----+--------+-----+-----------+-------+
|  1 | libai  | m   | baijing   |  NULL |
|  2 | lisi   | n   | shanghai  |  99.0 |
+----+--------+-----+-----------+-------+
2 rows in set (0.00 sec)
```

当插入的值和数据表中的字段顺序和个数都一致时，插入字段是可以省略的

insert into stu values(4,'lu','m','shanghai',87);

```
mysql> insert into stu values(4,'lu','m','shanghai',87);
Query OK, 1 row affected (0.00 sec)

mysql> select * from stu;
+----+--------+-----+-----------+-------+
| id | name   | sex | add       | score |
+----+--------+-----+-----------+-------+
|  1 | libai  | m   | baijing   |  NULL |
|  2 | lisi   | n   | shanghai  |  99.0 |
|  4 | lu     | m   | shanghai  |  87.0 |
+----+--------+-----+-----------+-------+
3 rows in set (0.00 sec)
```

自动增长的插入：

insert into stu values(null,'jianli','m','shenzheng',97);

```
mysql> insert into stu values(null,'jianli','m','shenzheng',97);
Query OK, 1 row affected (0.01 sec)

mysql> select * from stu;
+----+---------+-----+------------+-------+
| id | name    | sex | add        | score |
+----+---------+-----+------------+-------+
|  1 | libai   | m   | baijing    |  NULL |
|  2 | lisi    | n   | shanghai   |  99.0 |
|  4 | lu      | m   | shanghai   |  87.0 |
|  5 | jianli  | m   | shenzheng  |  97.0 |
+----+---------+-----+------------+-------+
4 rows in set (0.00 sec)
```

默认值的插入：

insert into stu values(null,'lii','m',default,77);

```
mysql> insert into stu values(null,'lii','m',default,77);
Query OK, 1 row affected (0.00 sec)

mysql> select * from stu;
+----+---------+-----+------------+-------+
| id | name    | sex | add        | score |
+----+---------+-----+------------+-------+
|  1 | libai   | m   | baijing    |  NULL |
|  2 | lisi    | n   | shanghai   |  99.0 |
|  4 | lu      | m   | shanghai   |  87.0 |
|  5 | jianli  | m   | shenzheng  |  97.0 |
|  6 | lii     | m   | address    |  77.0 |
+----+---------+-----+------------+-------+
5 rows in set (0.00 sec)
```

修改数据：

将libai的性别改为女（n）：

update stu set sex='n' where name='libai';

```
mysql> update stu set sex='n' where name='libai';
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from stu;
+----+--------+-----+-----------+-------+
| id | name   | sex | add       | score |
+----+--------+-----+-----------+-------+
|  1 | libai  | n   | baijing   |  NULL |
|  2 | lisi   | n   | shanghai  |  99.0 |
|  4 | lu     | m   | shanghai  |  87.0 |
|  5 | jianli | m   | shenzheng |  97.0 |
|  6 | lii    | m   | address   |  77.0 |
+----+--------+-----+-----------+-------+
5 rows in set (0.00 sec)
```

将所有字段的sex都设置为女（n）

update stu set sex='n';

将6号的学生性别改为女（n），地址改为tianjin:

update stu set sex='n',`add`='tianjing' where id='6';

```
mysql> update stu set sex='n',`add`='tianjing' where id='6';
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from stu;
+----+--------+-----+-----------+-------+
| id | name   | sex | add       | score |
+----+--------+-----+-----------+-------+
|  1 | libai  | n   | baijing   |  NULL |
|  2 | lisi   | n   | shanghai  |  99.0 |
|  4 | lu     | m   | shanghai  |  87.0 |
|  5 | jianli | m   | shenzheng |  97.0 |
|  6 | lii    | n   | tianjing  |  77.0 |
+----+--------+-----+-----------+-------+
5 rows in set (0.00 sec)
```

删除数据：

(原始数据)

```
mysql> select * from stu;
+----+--------+-----+-----------+-------+
| id | name   | sex | add       | score |
+----+--------+-----+-----------+-------+
|  1 | libai  | n   | baijing   |  NULL |
|  2 | lisi   | n   | shanghai  |  99.0 |
|  4 | lu     | m   | shanghai  |  87.0 |
|  5 | jianli | m   | shenzheng |  97.0 |
|  6 | lii    | n   | tianjing  |  77.0 |
+----+--------+-----+-----------+-------+
5 rows in set (0.00 sec)
```

删除lii:

delete from stu where name='lii';

```
mysql> delete from stu where name='lii';
Query OK, 1 row affected (0.00 sec)

mysql> select * from stu;
+----+--------+-----+-----------+-------+
| id | name   | sex | add       | score |
+----+--------+-----+-----------+-------+
|  1 | libai  | n   | baijing   |  NULL |
|  2 | lisi   | n   | shanghai  |  99.0 |
|  4 | lu     | m   | shanghai  |  87.0 |
|  5 | jianli | m   | shenzheng |  97.0 |
+----+--------+-----+-----------+-------+
4 rows in set (0.00 sec)
```

delete from stu;

删除stu表中所有数据

（现在stu表中数据）

```
mysql> select * from stu;
+----+--------+-----+-----------+-------+
| id | name   | sex | add       | score |
+----+--------+-----+-----------+-------+
|  1 | libai  | n   | baijing   |  NULL |
|  2 | lisi   | n   | shanghai  |  99.0 |
|  4 | lu     | m   | shanghai  |  87.0 |
|  5 | jianli | m   | shenzheng |  97.0 |
+----+--------+-----+-----------+-------+
4 rows in set (0.00 sec)
```

插入一条新数据：（6号刚才被删除，6号已被占用，自动增长所以是7）

insert into stu values(null,'aa','m','meizhou'86);

```
mysql> insert into stu values(null,'aa','m','meizhou',86);
Query OK, 1 row affected (0.00 sec)

mysql> select * from stu;
+----+--------+-----+-----------+-------+
| id | name   | sex | add       | score |
+----+--------+-----+-----------+-------+
|  1 | libai  | n   | baijing   |  NULL |
|  2 | lisi   | n   | shanghai  |  99.0 |
|  4 | lu     | m   | shanghai  |  87.0 |
|  5 | jianli | m   | shenzheng |  97.0 |
|  7 | aa     | m   | meizhou   |  86.0 |
+----+--------+-----+-----------+-------+
5 rows in set (0.00 sec)
```

查询数据：

查询所有学生的姓名和性别：

select name,sex from stu;

```
mysql> select name,sex from stu;
+--------+-----+
| name   | sex |
+--------+-----+
| libai  | n   |
| lisi   | n   |
| lu     | m   |
| jianli | m   |
| lii    | n   |
+--------+-----+
5 rows in set (0.00 sec)
```

查询所有男生(male)的信息：

select * from stu where sex='m';

```
mysql> select * from stu where sex='m';
+----+--------+-----+-----------+-------+
| id | name   | sex | add       | score |
+----+--------+-----+-----------+-------+
|  4 | lu     | m   | shanghai  |  87.0 |
|  5 | jianli | m   | shenzheng |  97.0 |
+----+--------+-----+-----------+-------+
2 rows in set (0.00 sec)
```

查询深圳的男生：

select * from stu where `add`='shenzheng' and sex='m';

```
mysql> select * from stu where `add`='shenzheng' and sex='m';
+----+--------+-----+-----------+-------+
| id | name   | sex | add       | score |
+----+--------+-----+-----------+-------+
|  5 | jianli | m   | shenzheng |  97.0 |
+----+--------+-----+-----------+-------+
1 row in set (0.00 sec)
```

查询所有女生和上海的男生：

select * from stu where sex='n' or (sex='m' and `add`='shenzheng');

```
mysql> select * from stu where sex='n' or (sex='m' and `add`='shenzheng');
+----+--------+-----+-----------+-------+
| id | name   | sex | add       | score |
+----+--------+-----+-----------+-------+
|  1 | libai  | n   | baijing   |  NULL |
|  2 | lisi   | n   | shanghai  |  99.0 |
|  5 | jianli | m   | shenzheng |  97.0 |
|  6 | lii    | n   | tianjing  |  77.0 |
+----+--------+-----+-----------+-------+
```

查询baijing 和 shanghai 的学生：

select * from stu where `add`='baijing' or `add`='shanghai';

```
mysql> select * from stu where `add`='baijing' or `add`='shanghai';
+----+-------+-----+----------+-------+
| id | name  | sex | add      | score |
+----+-------+-----+----------+-------+
|  1 | libai | n   | baijing  |  NULL |
|  2 | lisi  | n   | shanghai |  99.0 |
|  4 | lu    | m   | shanghai |  87.0 |
+----+-------+-----+----------+-------+
3 rows in set (0.00 sec)
```

查询成绩大于90分的学生：

select * from stu where score>=90;

```
mysql> select * from stu where score>=90;
+----+--------+-----+-----------+-------+
| id | name   | sex | add       | score |
+----+--------+-----+-----------+-------+
|  2 | lisi   | n   | shanghai  |  99.0 |
|  5 | jianli | m   | shenzheng |  97.0 |
+----+--------+-----+-----------+-------+
2 rows in set (0.00 sec)
```

升序：asc

降序：desc

按成绩降序排：

select * from stu order by score desc;

```
mysql> select * from stu order by score desc;
+----+--------+-----+-----------+-------+
| id | name   | sex | add       | score |
+----+--------+-----+-----------+-------+
|  2 | lisi   | n   | shanghai  |  99.0 |
|  5 | jianli | m   | shenzheng |  97.0 |
|  4 | lu     | m   | shanghai  |  87.0 |
|  6 | lii    | n   | tianjing  |  77.0 |
|  1 | libai  | n   | baijing   |  NULL |
+----+--------+-----+-----------+-------+
5 rows in set (0.00 sec)
```

男生按成绩升序排：

select * from stu where `sex`='m' order by score asc;

```
mysql> select * from stu where `sex`='m' order by score asc;
+----+--------+-----+-----------+-------+
| id | name   | sex | add       | score |
+----+--------+-----+-----------+-------+
|  4 | lu     | m   | shanghai  |  87.0 |
|  5 | jianli | m   | shenzheng |  97.0 |
+----+--------+-----+-----------+-------+
2 rows in set (0.00 sec)
```

取前3条数据：

select * from stu limit 3;

```
mysql> select * from stu limit 3;
+----+-------+-----+----------+-------+
| id | name  | sex | add      | score |
+----+-------+-----+----------+-------+
|  1 | libai | n   | baijing  |  NULL |
|  2 | lisi  | n   | shanghai |  99.0 |
|  4 | lu    | m   | shanghai |  87.0 |
+----+-------+-----+----------+-------+
3 rows in set (0.00 sec)
```

或：

```
mysql> select * from stu limit 0,3;
+----+-------+-----+----------+-------+
| id | name  | sex | add      | score |
+----+-------+-----+----------+-------+
|  1 | libai | n   | baijing  |  NULL |
|  2 | lisi  | n   | shanghai |  99.0 |
|  4 | lu    | m   | shanghai |  87.0 |
+----+-------+-----+----------+-------+
3 rows in set (0.00 sec)
```

从1开始取3条数据：(0是最上面一条)

select * from stu limit 1,3;

```
mysql> select * from stu limit 1,3;
+----+--------+-----+-----------+-------+
| id | name   | sex | add       | score |
+----+--------+-----+-----------+-------+
|  2 | lisi   | n   | shanghai  |  99.0 |
|  4 | lu     | m   | shanghai  |  87.0 |
|  5 | jianli | m   | shenzheng |  97.0 |
+----+--------+-----+-----------+-------+
3 rows in set (0.00 sec)
```

取 lu 和 jianli 2条数据：（从2开始取2条数据）

select * from stu limit 2,2;

```
mysql> select * from stu;
+----+--------+-----+-----------+-------+
| id | name   | sex | add       | score |
+----+--------+-----+-----------+-------+
|  1 | libai  | n   | baijing   |  NULL |
|  2 | lisi   | n   | shanghai  |  99.0 |
|  4 | lu     | m   | shanghai  |  87.0 |
|  5 | jianli | m   | shenzheng |  97.0 |
|  6 | lii    | n   | tianjing  |  77.0 |
+----+--------+-----+-----------+-------+
5 rows in set (0.00 sec)

mysql> select * from stu limit 2,2;
+----+--------+-----+-----------+-------+
| id | name   | sex | add       | score |
+----+--------+-----+-----------+-------+
|  4 | lu     | m   | shanghai  |  87.0 |
|  5 | jianli | m   | shenzheng |  97.0 |
+----+--------+-----+-----------+-------+
2 rows in set (0.00 sec)
```

找出前三名学生：

select * from stu order by score desc limit 3;

```
mysql> select * from stu order by score desc limit 3;
+----+--------+-----+-----------+-------+
| id | name   | sex | add       | score |
+----+--------+-----+-----------+-------+
|  2 | lisi   | n   | shanghai  |  99.0 |
|  5 | jianli | m   | shenzheng |  97.0 |
|  4 | lu     | m   | shanghai  |  87.0 |
+----+--------+-----+-----------+-------+
3 rows in set (0.00 sec)
```

聚合函数：

sum()
avg()
max()
min()
count()

求最高分：
select max(score) from stu;

```
mysql> select max(score) from stu;
+-------------+
| max(score) |
+-------------+
|       99.0 |
+-------------+
1 row in set (0.00 sec)
```

男生人数：
select count(*) from stu where sex='m';

```
mysql> select count(*) from stu where sex='m';
+----------+
| count(*) |
+----------+
|        3 |
+----------+
1 row in set (0.00 sec)
```

//

通过php代码连接操作MySQL数据库

连接数据库

通过 mysql_connect(数据库地址,用户名,密码)，返回一个数据库的连接对象

```
$link = mysql_connect('localhost','root','lu');     //连接数据库
if($link)
{
    echo '连接成功';
}
else
{
    //echo '连接失败';
    die('连接失败');     //终止执行 或  exit();
}
```

用or优化上面代码：

```
$link = mysql_connect('localhost', 'root', 'lu') or die('连接数据库失败');
```

继续优化，加@屏蔽错误信息

```
$link = @mysql_connect('localhost', 'root', 'lu') or die('连接数据库失败');
```

选择数据库：（db_lu是数据库名）

方法一：通过‘use dbname’来选择数据库

mysql_query('use db_lu') or die('数据库选择失败');

方法二：直接用php函数

mysql_select_db('db_lu') or die('数据库选择失败');

设置mysql客户端的字符编码：防止出现乱码

mysql_query('set names utf8');

执行SQL语句，获得表中数据：（stu是数据库表名）

$resource = mysql_query('select * from stu'); //结果集是一个资源类型数据

取出结果集中数据一（mysql_fetch_row）：

$rows = mysql_fetch_row($reource);

```php
<?php
$link = @mysql_connect('localhost', 'root', 'lu') or die('连接数据库失败');

mysql_select_db('db_lu') or die('数据库选择失败');

mysql_query('set names utf8');

$resource = mysql_query('select * from stu');  //结果集是一个资源类型数据

$rows = mysql_fetch_row($resource);

echo $rows[0].' --- '.$rows[1].' --- '.$rows[2];
?>
```

浏览器输出：

1 --- 李白 --- 男

数据库中的数据：

取第二条数据：

```php
$rows = mysql_fetch_row($resource);
echo $rows[0].' --- '.$rows[1].' --- '.$rows[2].'<br/>';

$rows = mysql_fetch_row($resource);
echo $rows[0].' --- '.$rows[1].' --- '.$rows[2];
```

输出下一条记录的内容：

mysql link test

1 --- 李白 --- 男
2 --- 杜甫 --- 女

取出表中全部数据：

```
while($rows = mysql_fetch_row($resource)
{
    echo $rows[0].' --- '.$rows[1].' --- '. $rows[2].' --- '. $rows[3].' --- '. $rows[4].' --- '. $rows[5].'<br/>';
}
```

循环取出：

```
while($rows = mysql_fetch_row($resource))
{
    echo $rows[0].' --- '.$rows[1].' --- '. $rows[2].' --- '. $rows[3].' --- '. $rows[4].'<br/>';
}
```

mysql link test

1 --- 李白 --- 男 --- 22 --- 北京
2 --- 杜甫 --- 女 --- 20 --- 上海
3 --- 张三 --- 男 --- 23 --- 深圳

匹配**索引数组**缺点：增加或删除列（字段）时，会影响程序中数组的索引编号。

关联数组：

取出结果集数据 二（mysql_fetch_assoc ）：

循环取出，取出一条记录匹配成**关联数组**：

```
while($rows = mysql_fetch_assoc($resource))
{
    echo $rows['stuid'].' -- '.$rows['stuname'].' -- '. $rows['stusex'].' -- '. $rows['stuage'].' -- '. $rows['stuaddr'].'<br/>';
}
```

数组里面要加引号 '' ；

取出结果集数据 三（mysql_fetch_array ）：

//既支持关联数组又支持索引数组

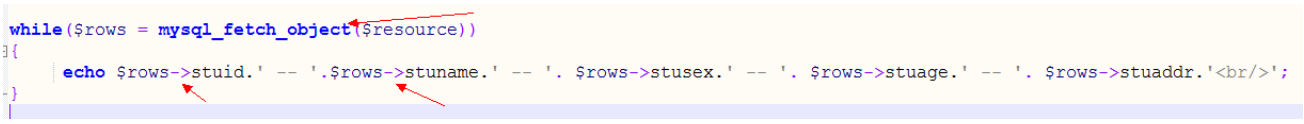
```
while($rows = mysql_fetch_array($resource))
{
    echo $rows[0].' -- '.$rows['stuname'].' -- '. $rows[2].' -- '. $rows['stuage'].' -- '. $rows['stuaddr'].'<br/>';
}
```

```
while($rows = mysql_fetch_array($resource))
{
    echo $rows[0].' -- '.$rows['stuname'].' -- '. $rows[2].' -- '. $rows['stuage'].' -- '. $rows['stuaddr'].'<br/>';
}
```

取出结果集数据 四（mysql_fetch_object ）：

通过 -> 访问对象属性

```
while($rows = mysql_fetch_object($resource))
{
    echo $rows->stuid.' -- '.$rows->stuname.' -- '. $rows->stusex.' -- '. $rows->stuage.' -- '. $rows->stuaddr.'<br/>';
}
```



释放资源（mysql_free_result($resource)）：

mysql_free_result($resource);

关闭连接（myql_close($link)）：

mysql_close($link);