

MySQL

杨亮



关系数据库

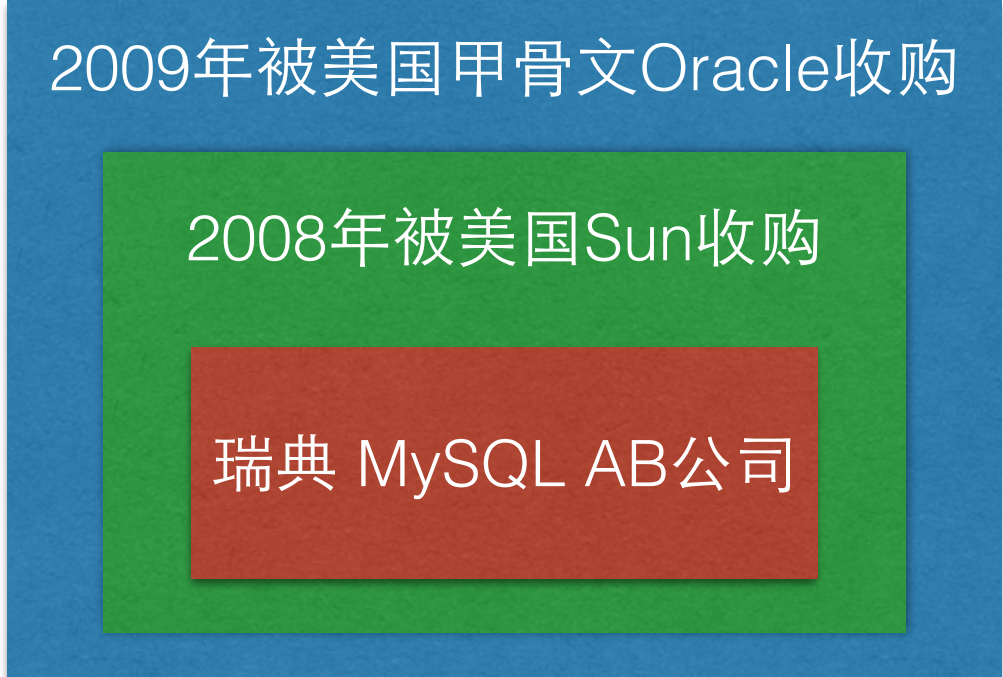


百度对DBA的要求
通读MySQL代码

- 开源：代码面前了无秘密
- 体积小：安装方便，快捷，无严格要求
- 性能足可以与商业数据库（如Oracle）媲美
- 开源社区提供了丰富的扩展功能
- 简单易用，跨平台，支持多用户



关系



id	name	age	gender	dept	phone
1	张三	25	男	开发部	137XXX
2	李四	27	男	开发部	135XXX
3	王五	24	男	市场部	130XXX
4	赵六	25	女	商场部	151XXX

SQL: Structured Query Language

3000+练习

Data Definition Language

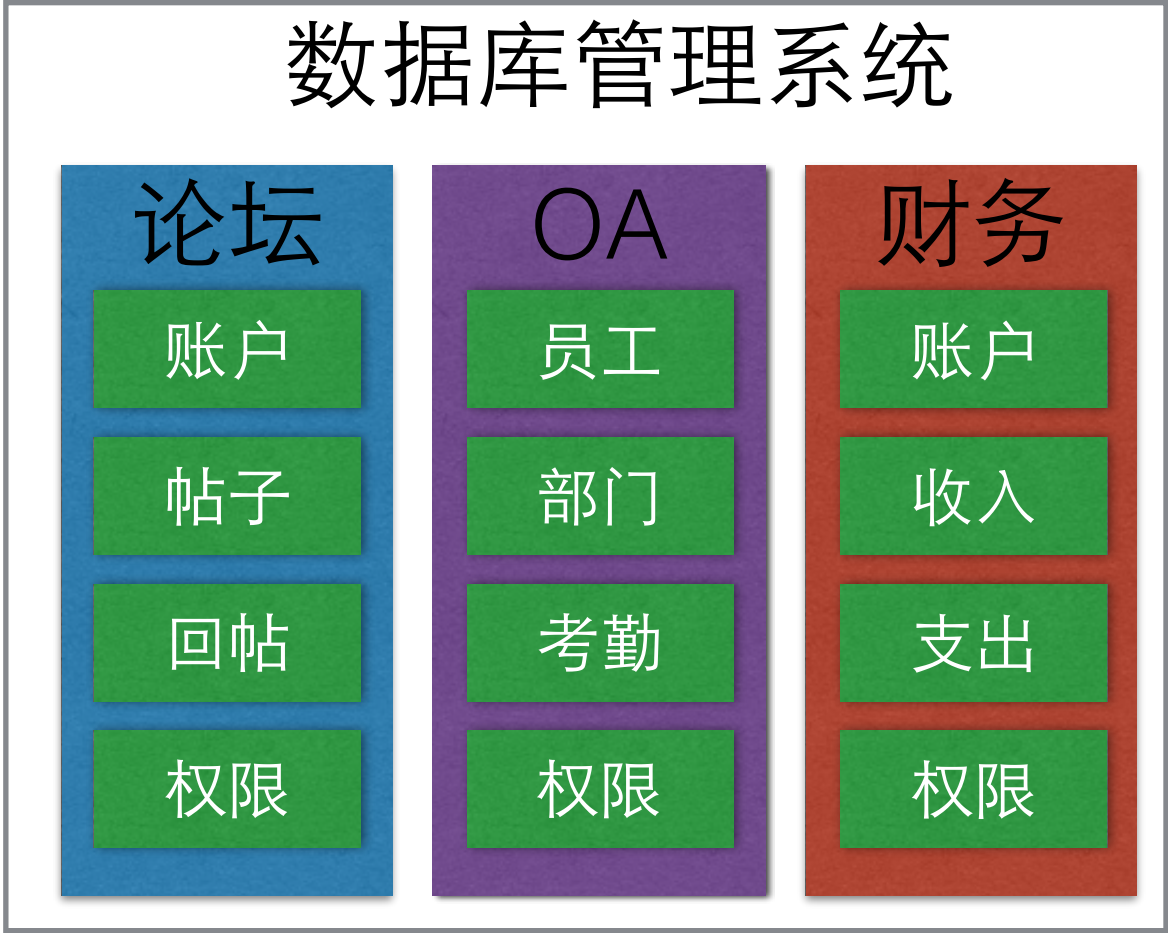
进行与数据库或表结构有关的操作

Data Manipulation Language

进行数据的增、删、改、查

Data Control Language

与数据库管理系统相关的设置



一个数据库中放置与仅一个系统相关的所有表

id	name	age	gender	dept	phone
1	张三	25	男	开发部	137XXX
2	李四	27	男	开发部	135XXX
3	王五	24	男	市场部	130XXX
4	赵六	25	女	商场部	151XXX

访问MySQL的三种方式

命令行 (强烈推荐)

用户名 (Username) 密码 (Password)

mysql -h127.0.0.1 -P3306 -uroot -p123456

IP地址 (host)

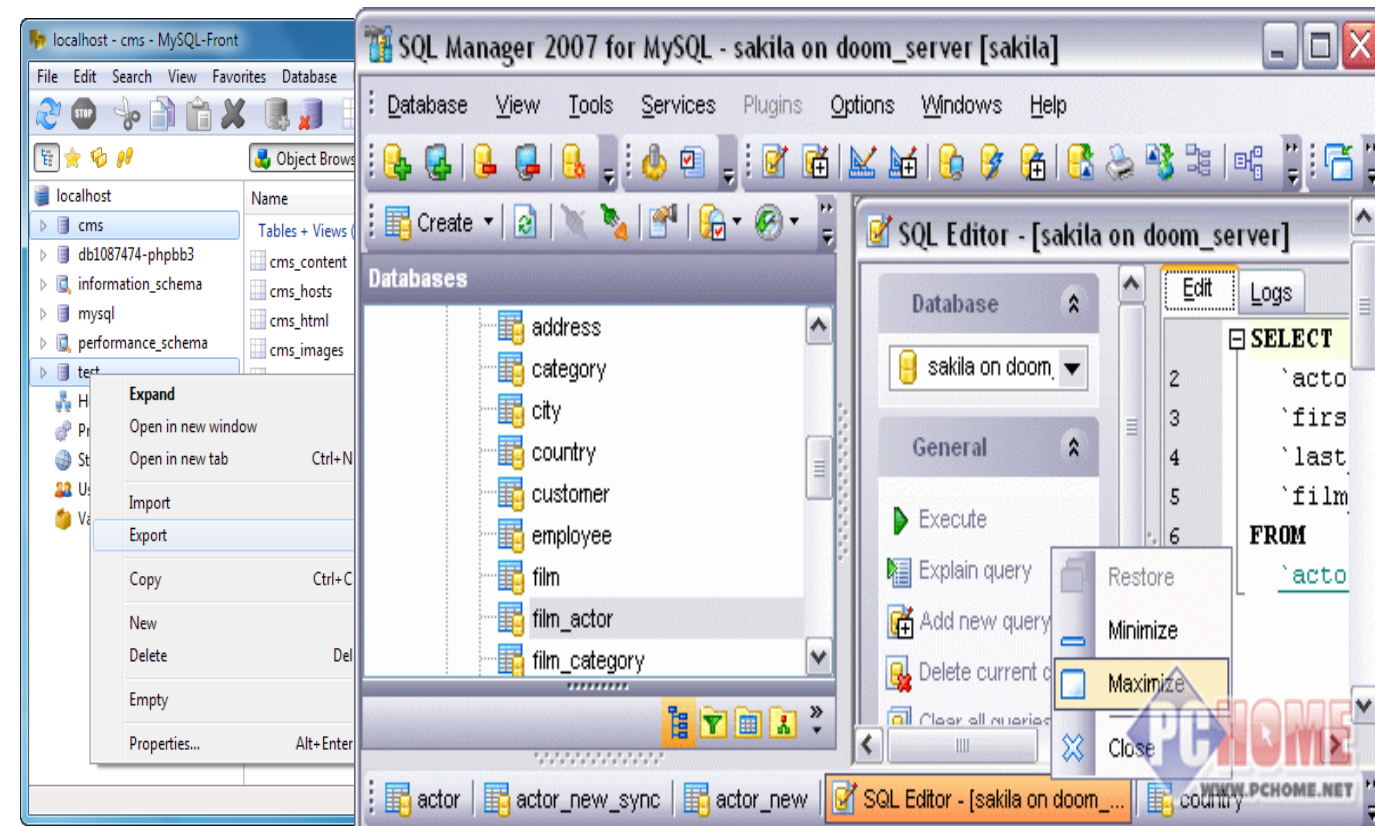
端口 (Port)

如果是服务器上的命令行可使用localhost

3306是默认端口，可以省略

浏览器 (phpMyAdmin)

客户端 (不推荐)



字符集

int -> character

latin1: 西欧希腊字符

gbk: 简体中文

big5: 繁体中文

utf8: 所有国家的字符

各种乱码的解决方法

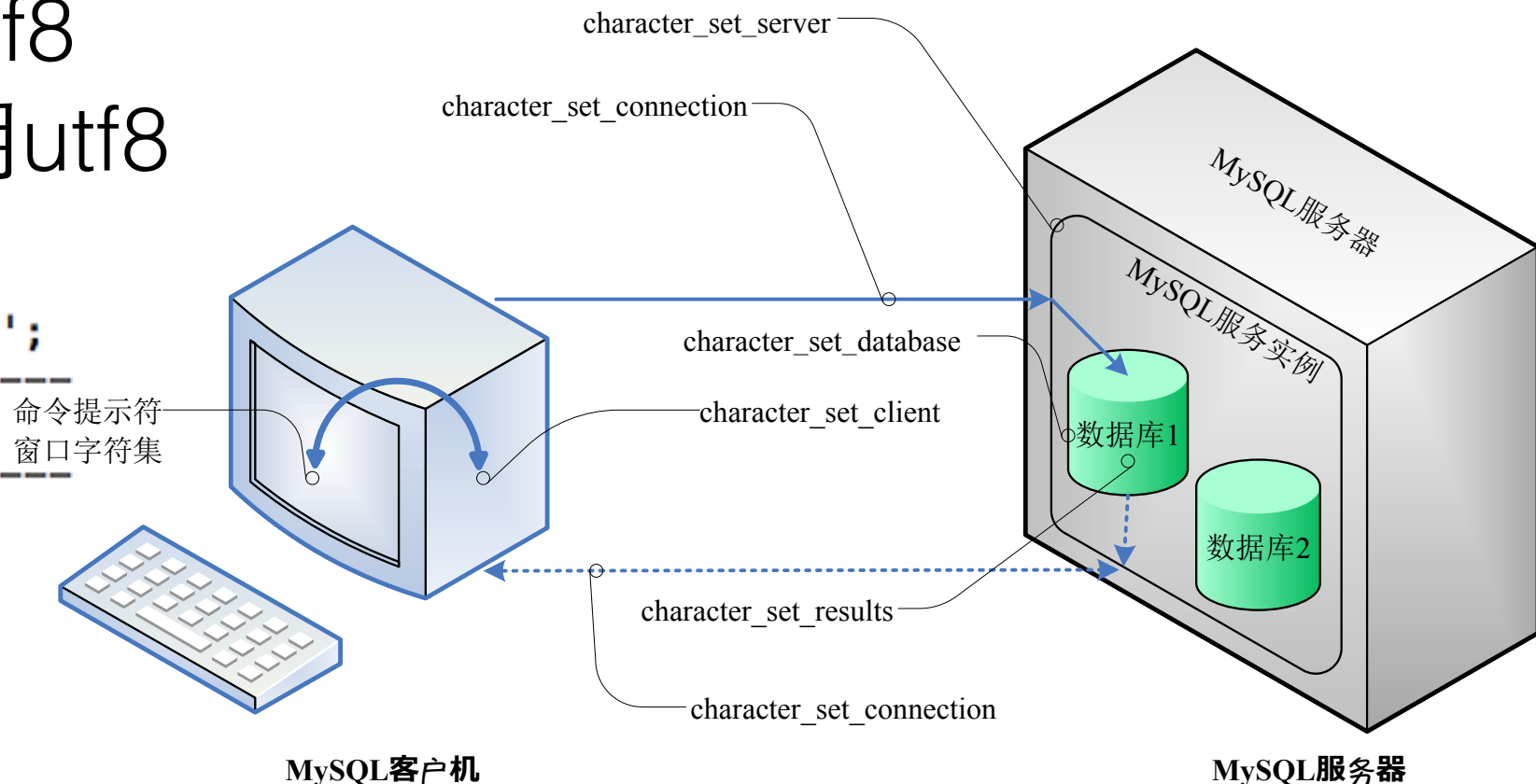
- 1: 所有数据库、表、字段都用utf8
- 2: 所有html页面都用utf8
- 3: 所有的PHP页面都用utf8

ASCII TABLE

Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char
0	0	[NULL]	32	20	[SPACE]	64	40	@	96	60	`
1	1	[START OF HEADING]	33	21	!	65	41	A	97	61	a
2	2	[START OF TEXT]	34	22	"	66	42	B	98	62	b
3	3	[END OF TEXT]	35	23	#	67	43	C	99	63	c
4	4	[END OF TRANSMISSION]	36	24	\$	68	44	D	100	64	d
5	5	[ENQUIRY]	37	25	%	69	45	E	101	65	e
6	6	[ACKNOWLEDGE]	38	26	&	70	46	F	102	66	f
7	7	[BELL]	39	27	'	71	47	G	103	67	g
8	8	[BACKSPACE]	40	28	(72	48	H	104	68	h
9	9	[HORIZONTAL TAB]	41	29)	73	49	I	105	69	i
10	A	[LINE FEED]	42	2A	*	74	4A	J	106	6A	j
11	B	[VERTICAL TAB]	43	2B	+	75	4B	K	107	6B	k
12	C	[FORM FEED]	44	2C	,	76	4C	L	108	6C	l
13	D	[CARRIAGE RETURN]	45	2D	-	77	4D	M	109	6D	m
14	E	[SHIFT OUT]	46	2E	.	78	4E	N	110	6E	n
15	F	[SHIFT IN]	47	2F	/	79	4F	O	111	6F	o
16	10	[DATA LINK ESCAPE]	48	30	0	80	50	P	112	70	p
17	11	[DEVICE CONTROL 1]	49	31	1	81	51	Q	113	71	q
18	12	[DEVICE CONTROL 2]	50	32	2	82	52	R	114	72	r
19	13	[DEVICE CONTROL 3]	51	33	3	83	53	S	115	73	s
20	14	[DEVICE CONTROL 4]	52	34	4	84	54	T	116	74	t
21	15	[NEGATIVE ACKNOWLEDGE]	53	35	5	85	55	U	117	75	u
22	16	[SYNCHRONOUS IDLE]	54	36	6	86	56	V	118	76	v
23	17	[ENG OF TRANS. BLOCK]	55	37	7	87	57	W	119	77	w
24	18	[CANCEL]	56	38	8	88	58	X	120	78	x
25	19	[END OF MEDIUM]	57	39	9	89	59	Y	121	79	y
26	1A	[SUBSTITUTE]	58	3A	:	90	5A	Z	122	7A	z
27	1B	[ESCAPE]	59	3B	;	91	5B	[123	7B	{
28	1C	[FILE SEPARATOR]	60	3C	<	92	5C	\	124	7C	
29	1D	[GROUP SEPARATOR]	61	3D	=	93	5D]	125	7D	}
30	1E	[RECORD SEPARATOR]	62	3E	>	94	5E	^	126	7E	~
31	1F	[UNIT SEPARATOR]	63	3F	?	95	5F	_	127	7F	[DEL]

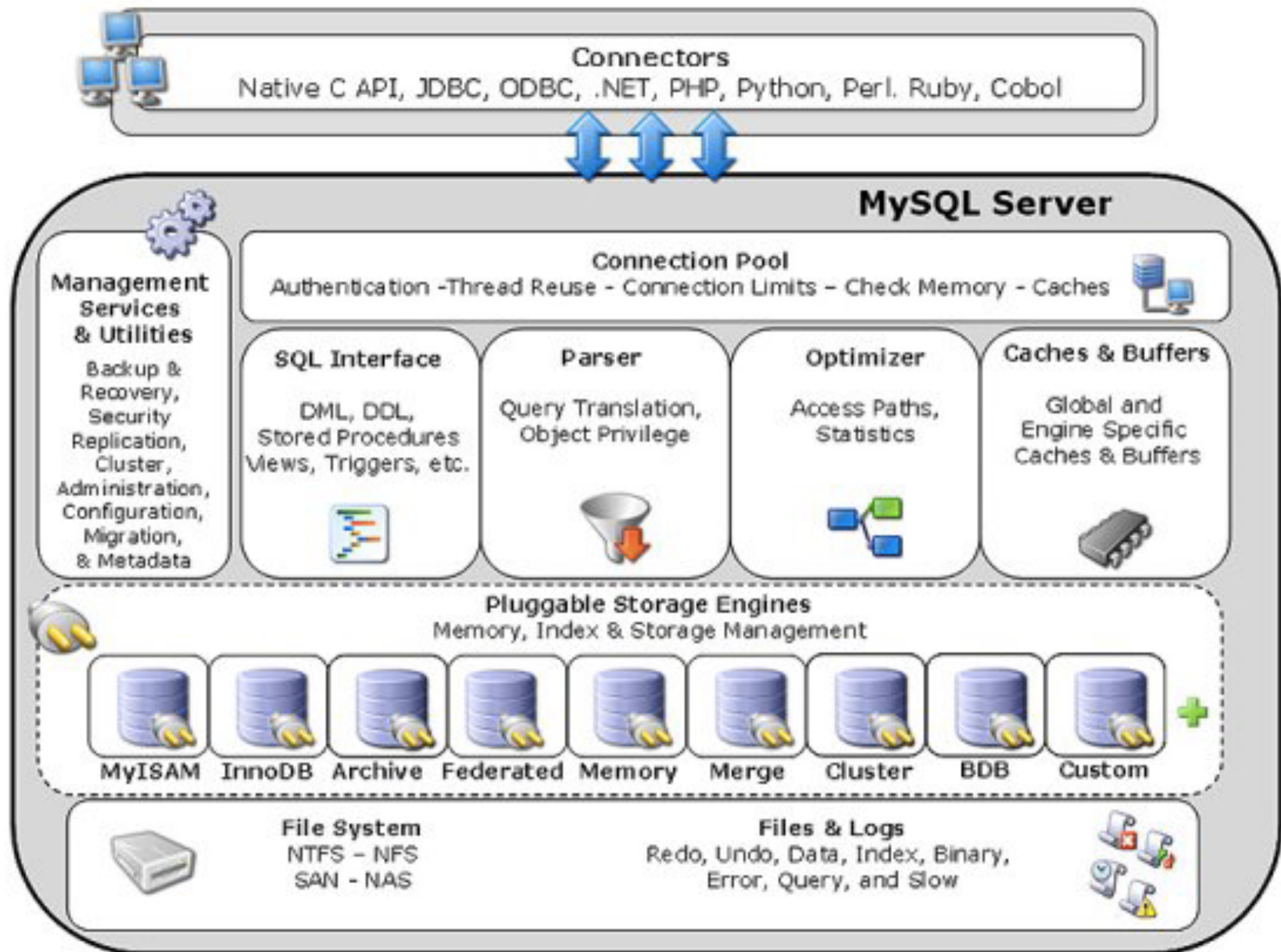
```
mysql> show variables like 'character%';
```

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



MySQL Architecture

分层



存储引擎

负责具体的数据存储与查询

不同表可以使用不同的存储引擎

MyISAM	InnoDB
不支持外键	支持外键
不支持事务	支持事务
select性能较高	频繁增删改查
MySQL原生	第三方扩展

年	制造商	型号	说明	价值
1997	Ford	E350	ac, abs, moon	3000
1999	Chevy	Venture "Extended Edition"		4900
1999	Chevy	Venture "Extended Edition, Very Large"		5000
1996	Jeep	Grand Cherokee	MUST SELL! air, moon roof, loaded	4799

年,制造商,型号,说明,价值
1997,Ford,E350,"ac, abs, moon",3000
1999,Chevy,"Venture ""Extended Edition""",4900
1999,Chevy,"Venture ""Extended Edition, Very Large""",5000
1996,Jeep,Grand Cherokee,"MUST SELL! air, moon roof, loaded",4799

.CSV

Engine	Support	Comment	Transactions	XA	Savepoints
FEDERATED	NO	Federated MySQL storage engine	NULL	NULL	NULL
MRG_MYISAM	YES	Collection of identical MyISAM tables	NO	NO	NO
MyISAM	YES	MyISAM storage engine	NO	NO	NO
BLACKHOLE	YES	/dev/null storage engine (anything you write to it disappears)	NO	NO	NO
CSV	YES	CSV storage engine	NO	NO	NO
MEMORY	YES	Hash based, stored in memory, useful for temporary tables	NO	NO	NO
ARCHIVE	YES	Archive storage engine	NO	NO	NO
InnoDB	DEFAULT	Supports transactions, row-level locking, and foreign keys	YES	YES	YES
PERFORMANCE_SCHEMA	YES	Performance Schema	NO	NO	NO

索引

如果我们把一个表的内容认为是一本字典
那索引就相当于字典的目录

加速查找 (Select)

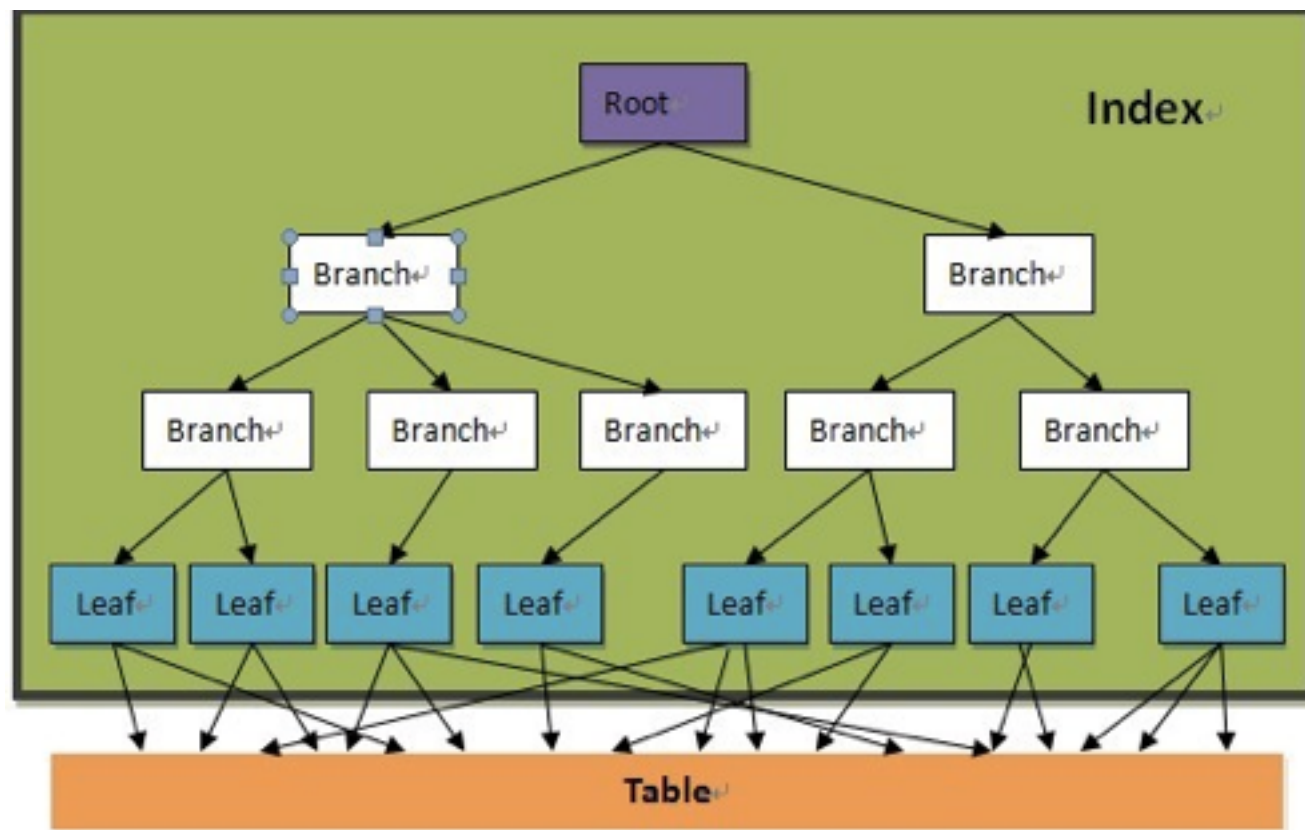
唯一索引

普通索引

复合索引



实现方式: B-Tree (Balance Tree)



根节点

分支节点1

分支节点3

叶子节点

叶子节点内到数据块号的指针

心部		示部	
心	542	示	451
一	24	示	440
必	639	示	354
志	483	示	354
志	483	示	470
志	483	示	382
志	483	示	217
志	483	示	455
志	483	示	240
志	483	示	243
志	483	示	34
志	483	示	22
志	483	示	196
志	483	示	601
志	483	示	85

SQL 语句

Data Definition Language

进行与数据库或表结构有关的操作

数据库

```
CREATE {DATABASE | SCHEMA} [IF NOT EXISTS] db_name  
[create_specification [, create_specification] ...]
```

create_specification:

```
    [DEFAULT] CHARACTER SET charset_name  
    | [DEFAULT] COLLATE collation_name
```

```
SHOW {DATABASES | SCHEMAS} [LIKE 'pattern']
```

```
SHOW CREATE {DATABASE | SCHEMA} db_name
```

```
USE db_name
```

```
DROP {DATABASE | SCHEMA} [IF EXISTS] db_name
```

create
show
show create
use
drop

IF EXISTS is used to prevent an error from occurring if the database does not exist.

Data Definition Language

进行与数据库或表结构有关的操作

表结构

CREATE [TEMPORARY] TABLE [IF NOT EXISTS] tbl_name
 [(create_definition,...)]
 [table_options] [select_statement]

create_definition:
 column_definition
 | [CONSTRAINT [symbol]] PRIMARY KEY [index_type] (index_col_name,...)
 | KEY [index_name] [index_type] (index_col_name,...)
 | INDEX [index_name] [index_type] (index_col_name,...)
 | [CONSTRAINT [symbol]] UNIQUE [INDEX]
 [index_name] [index_type] (index_col_name,...)
 | [FULLTEXT|SPATIAL] [INDEX] [index_name] (index_col_name,...)
 | [CONSTRAINT [symbol]] FOREIGN KEY
 [index_name] (index_col_name,...) [reference_definition]
 | CHECK (expr)

create
show
show create
desc
drop
alter

column_definition:
 col_name type [NOT NULL | NULL] [DEFAULT default_value]
 [AUTO_INCREMENT] [UNIQUE [KEY] | [PRIMARY] KEY]
 [COMMENT 'string'] [reference_definition]

CREATE [TEMPORARY] TABLE [IF NOT EXISTS]
 tbl_name [(] LIKE old_tbl_name [)];

Data Definition Language

进行与数据库或表结构有关的操作

表结构

type:

TINYINT[(length)] [UNSIGNED] [ZEROFILL]
SMALLINT[(length)] [UNSIGNED] [ZEROFILL]
MEDIUMINT[(length)] [UNSIGNED] [ZEROFILL]
INT[(length)] [UNSIGNED] [ZEROFILL]
INTEGER[(length)] [UNSIGNED] [ZEROFILL]
BIGINT[(length)] [UNSIGNED] [ZEROFILL]

整形

REAL[(length,decimals)] [UNSIGNED] [ZEROFILL]
DOUBLE[(length,decimals)] [UNSIGNED] [ZEROFILL]
FLOAT[(length,decimals)] [UNSIGNED] [ZEROFILL]
DECIMAL(length,decimals) [UNSIGNED] [ZEROFILL]
NUMERIC(length,decimals) [UNSIGNED] [ZEROFILL]

浮点

DATE
TIME
TIMESTAMP
DATETIME
YEAR

日期

字符串

CHAR(length) [BINARY | ASCII | UNICODE]
VARCHAR(length) [BINARY]

BINARY(length)
VARBINARY(length)
TINYBLOB
BLOB
MEDIUMBLOB
LONGBLOB

二进制

TINYTEXT [BINARY]
TEXT [BINARY]
MEDIUMTEXT [BINARY]
LONGTEXT [BINARY]

长文本

ENUM(value1,value2,value3,...)
SET(value1,value2,value3,...)
spatial_type

复合类型

create
show
show create
desc
drop
alter

table_option:

{ENGINE|TYPE} [=] engine_name
AUTO_INCREMENT [=] value
AVG_ROW_LENGTH [=] value
[DEFAULT] CHARACTER SET charset_name [COLLATE collation]
CHECKSUM [=] {0 | 1}
COMMENT [=] 'string'
CONNECTION [=] 'connect_string'
MAX_ROWS [=] value
MIN_ROWS [=] value
PACK_KEYS [=] {0 | 1 | DEFAULT}
PASSWORD [=] 'string'
DELAY_KEY_WRITE [=] {0 | 1}
ROW_FORMAT [=] {DEFAULT|DYNAMIC|FIXED|COMPRESSED|REDUNDANT}
UNION [=] (tbl_name[,tbl_name]...)
INSERT_METHOD [=] { NO | FIRST | LAST }
DATA DIRECTORY [=] 'absolute path to directory'
INDEX DIRECTORY [=] 'absolute path to directory'

Data Definition Language

进行与数据库或表结构有关的操作

表结构

ALTER [IGNORE] TABLE tbl_name
alter_specification [, alter_specification] ...

alter_specification:
ADD [COLUMN] column_definition [FIRST | AFTER col_name]
| ADD [COLUMN] (column_definition,...)
| ADD INDEX [index_name] [index_type] (index_col_name,...)
| ADD [CONSTRAINT [symbol]]
PRIMARY KEY [index_type] (index_col_name,...)
| ADD [CONSTRAINT [symbol]]
UNIQUE [INDEX] [index_name] [index_type] (index_col_name,...)
| ADD [FULLTEXT|SPATIAL] [INDEX] [index_name] (index_col_name,...)
| ADD [CONSTRAINT [symbol]]
FOREIGN KEY [index_name] (index_col_name,...)
[reference_definition]
| ALTER [COLUMN] col_name {SET DEFAULT literal | DROP DEFAULT}
| CHANGE [COLUMN] old_col_name column_definition
[FIRST|AFTER col_name]
| MODIFY [COLUMN] column_definition [FIRST | AFTER col_name]
| DROP [COLUMN] col_name
| DROP PRIMARY KEY
| DROP INDEX index_name
| DROP FOREIGN KEY fk_symbol
| DISABLE KEYS
| ENABLE KEYS
| RENAME [TO] new_tbl_name
| ORDER BY col_name
| CONVERT TO CHARACTER SET charset_name [COLLATE collation_name]
| [DEFAULT] CHARACTER SET charset_name [COLLATE collation_name]
| DISCARD TABLESPACE
| IMPORT TABLESPACE
| table_options

create
show
show create
desc
drop
alter

Data Definition Language

进行与数据库或表结构有关的操作

表结构

SHOW [FULL] TABLES [FROM db_name] [LIKE 'pattern']

SHOW CREATE TABLE tbl_name

{DESCRIBE | DESC} tbl_name [col_name]

Field	Type	Null	Key	Default	Extra
Host	char(60)	NO	PRI		
Db	char(64)	NO	PRI		
User	char(16)	NO	PRI		
Select_priv	enum('N','Y')	NO		N	
Insert_priv	enum('N','Y')	NO		N	
Update_priv	enum('N','Y')	NO		N	
Delete_priv	enum('N','Y')	NO		N	

create
show
show create
desc
drop
alter

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

Data Definition Language 进行与数据库或表结构有关的操作

索引

区分索引与约束

```
CREATE [UNIQUE|FULLTEXT|SPATIAL] INDEX index_name  
    [USING index_type]  
    ON tbl_name (index_col_name,...)
```

index_col_name:

col_name [(length)] [ASC | DESC]

```
SHOW INDEX FROM tbl_name [FROM db_name]
```

```
DROP INDEX index_name ON tbl_name
```

create
show
drop

创建(create table)数据库表时也可以增加索引

```
| [CONSTRAINT [symbol]] PRIMARY KEY [index_type]  
(index_col_name,...)  
| KEY [index_name] [index_type] (index_col_name,...)  
| INDEX [index_name] [index_type] (index_col_name,...)  
| [FULLTEXT|SPATIAL] [INDEX] [index_name]  
(index_col_name,...)
```

索引 基本原则

出现在where, group, order中的字段才有必要建索引

某个字段离散度越高，越适合做索引的关键字

组合索引字段顺序应与where中字段顺序相一致

对于字符串类型的列的索引，应使用前缀索引

两个表连接的字段应该建索引

占用空间小的字段使用做索引（int）



Data Manipulation Language 进行数据的增、删、改、查

```
INSERT [LOW_PRIORITY | DELAYED | HIGH_PRIORITY] [IGNORE]
      [INTO] tbl_name [(col_name,...)]
      VALUES ({expr | DEFAULT},...),(...),...
      [ ON DUPLICATE KEY UPDATE col_name=expr, ... ]
```

```
INSERT [LOW_PRIORITY | DELAYED | HIGH_PRIORITY] [IGNORE]
      [INTO] tbl_name SET col_name={expr | DEFAULT}, ...
      [ ON DUPLICATE KEY UPDATE col_name=expr, ... ]
```

```
INSERT [LOW_PRIORITY | HIGH_PRIORITY] [IGNORE]
      [INTO] tbl_name [(col_name,...)] SELECT ...
      [ ON DUPLICATE KEY UPDATE col_name=expr, ... ]
```

```
UPDATE [LOW_PRIORITY] [IGNORE] tbl_name
      SET col_name1=expr1 [, col_name2=expr2 ...]
      [WHERE where_condition][ORDER BY ...]
      [LIMIT row_count]
```

```
DELETE [LOW_PRIORITY] [QUICK] [IGNORE]
      FROM tbl_name [WHERE where_condition]
      [ORDER BY ...] [LIMIT row_count]
```

insert
update
delete
select

Data Manipulation Language 进行数据的增、删、改、查

单表查询

insert
update
delete
select

SELECT

[ALL | DISTINCT | DISTINCTROW]
[HIGH_PRIORITY]
[STRAIGHT_JOIN]
[SQL_SMALL_RESULT] [SQL_BIG_RESULT] [SQL_BUFFER_RESULT]
[SQL_CACHE | SQL_NO_CACHE] [SQL_CALC_FOUND_ROWS]

select_expr, ...

[FROM table_references
[WHERE where_condition]
[GROUP BY {col_name | expr | position}
[ASC | DESC], ... [WITH ROLLUP]]
[HAVING where_condition]
[ORDER BY {col_name | expr | position}
[ASC | DESC], ...]
[LIMIT {[offset,] row_count | row_count OFFSET offset}]]

[PROCEDURE procedure_name(argument_list)]
[INTO OUTFILE 'file_name' export_options
| INTO DUMPFILE 'file_name']
[FOR UPDATE | LOCK IN SHARE MODE]]

Data Manipulation Language 进行数据的增、删、改、查

单表查询

insert
update
delete
select

SELECT

[ALL | DISTINCT | DISTINCTROW]
[HIGH_PRIORITY]
[STRAIGHT_JOIN]
[SQL_SMALL_RESULT] [SQL_BIG_RESULT] [SQL_BUFFER_RESULT]
[SQL_CACHE | SQL_NO_CACHE] [SQL_CALC_FOUND_ROWS]

select_expr, ...

[FROM table_references
[WHERE where_condition]
[GROUP BY {col_name | expr | position}
[ASC | DESC], ... [WITH ROLLUP]]
[HAVING where_condition]
[ORDER BY {col_name | expr | position}
[ASC | DESC], ...]
[LIMIT {[offset,] row_count | row_count OFFSET offset}]]

[PROCEDURE procedure_name(argument_list)]
[INTO OUTFILE 'file_name' export_options
| INTO DUMPFILE 'file_name']
[FOR UPDATE | LOCK IN SHARE MODE]]

作业一

学习与MySQL相关的其他细节

PMWD Chapters 8 -10

<http://wenku.baidu.com/view/1acfe579ee06eff9aef80752.html>

PHP访问MySQL

面向对象
Object oriented

基于过程
Procedural

连接数据库

拼接SQL

进行查询

获取结果

释放空间

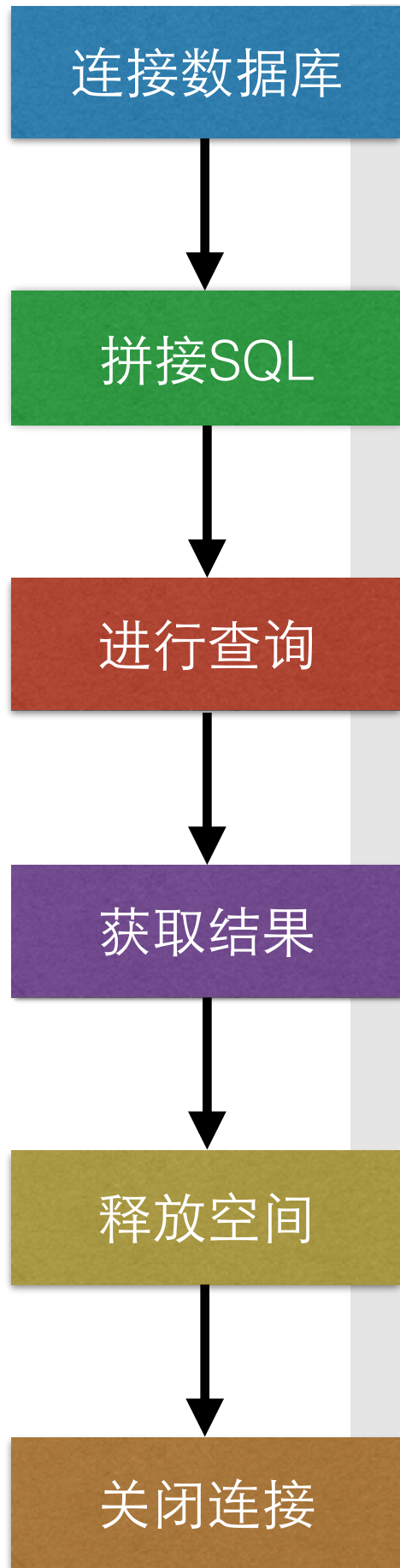
关闭连接

```
1 <?php
2 $link = mysqli_connect('localhost', 'root', 'root', 'php_course');
3 if(!$link) die('Connect Error: '.mysqli_connect_error());
4
5 $sql = "SELECT * FROM user";
6
7 $result = mysqli_query($link, $sql);
8 if(!$result) die('Query Error: '.$SQL);
9
10 $num_of_rows = mysqli_num_rows($result);
11 echo "<table border='1'>";
12 while($row = mysqli_fetch_array($result, MYSQLI_ASSOC)) {
13     echo "<tr>\n";
14     foreach ($row as $no => $value) {
15         echo "<td>".$value."</td>\n";
16     }
17     echo "</tr>\n";
18 }
19 echo "</table>";
20
21 mysqli_free_result($result);
22 mysqli_close($link);
23 ?>
```

基于过程
Procedural

为什么要面向对象

```
1 <?php
2 $link = mysqli_connect('localhost', 'root', 'root', 'php_course');
3 if(!$link) die('Connect Error: ' . mysqli_connect_error());
4
5 $sql = "SELECT * FROM user";
6
7 $result = mysqli_query($link, $sql);
8 if(!$result) die('Query Error: ' . $SQL);
9
10 $num_of_rows = mysqli_num_rows($result);
11 echo "<table border='1'>";
12 while($row = mysqli_fetch_array($result, MYSQLI_ASSOC)) {
13     echo "<tr>\n";
14     foreach ($row as $no => $value) {
15         echo "<td>".$value."</td>\n";
16     }
17     echo "</tr>\n";
18 }
19 echo "</table>";
20
21 mysqli_free_result($result);
22 mysqli_close($link);
23 ?>
```

```
1 <?php
2 $link = new mysqli('localhost', 'root', 'root', 'php_course');
3 if($link->connect_errno) {
4     die("Connection Failed: ". $link->connection_error);
5 }
6
7 $sql = "SELECT * FROM user";
8
9 if($result = $link->query($sql)) {
10     echo "<table border='1'>";
11     while ($row = $result->fetch_assoc()) {
12         echo "<tr>";
13         echo "<td>{$row['id']}</td>";
14         echo "<td>{$row['name']}</td>";
15         echo "<td>{$row['age']}</td>";
16         echo "<td>{$row['class']}</td>";
17         echo "</tr>";
18         echo "</tr>";
19     }
20     echo "</table>";
21     $result->free();
22 }
23
24 $link->close();
25 ?>
```

面向对象
Object oriented

作业二

学习PHP访问MySQL的细节

PMWD Chapters 11

<http://php.net/manual/en/book.mysql.php>

大作业

用PHP完成一个简单的3000行
以上的系统
论坛、商城、信息发布系统...

PMWD Chapters 25-31