

Table of Contents

数据库课程	1.1
数据库介绍	1.2
数据库是什么	1.2.1
数据库分类	1.2.2
SQL介绍	1.2.3
MySQL介绍	1.2.4
MySQL安装与使用	1.2.5

为什么学习数据库？

1、90%以上的软件都需要操作数据，比如游戏、社交、新闻、商城、财务等，这些软件都在不停的展示、存储数据，它们的数据都存储在数据库，数据库是软件的基础。

2、测试工程师在测试软件的过程中，不仅需要在界面上操作，还需要检查数据库中的数据是否正确。从而在软件出问题，测出更深层的问题。

例如：测试注册登录功能，在输入了注册信息后，提示注册成功，但是使用刚才注册的信息登录不成功。这时需要检查数据库中是否保存了正确的注册信息，如果数据库中没有保存数据，那么使用刚才注册的账号肯定登录不了，这样就能定位问题：注册时，没有把数据存储起来。

登录 · 注册

你的昵称

手机号

设置密码

注册

数据库阶段知识点

1. 数据库的基本概念

2

黑马程序员-软件测试

2. Navicat 操作数据库
 3. SQL 语言（重点）
 4. MySQL 高级（了解）
 5. Redis
-

学习目标

- 熟练编写 SQL 语言中的查询语句

数据库介绍

目标

- 了解数据库是什么？
- 了解 RDBMS 是什么？
- 了解 SQL 是什么？
- 了解 MySQL 是什么？

佐智播客-黑马程序员

数据库是什么

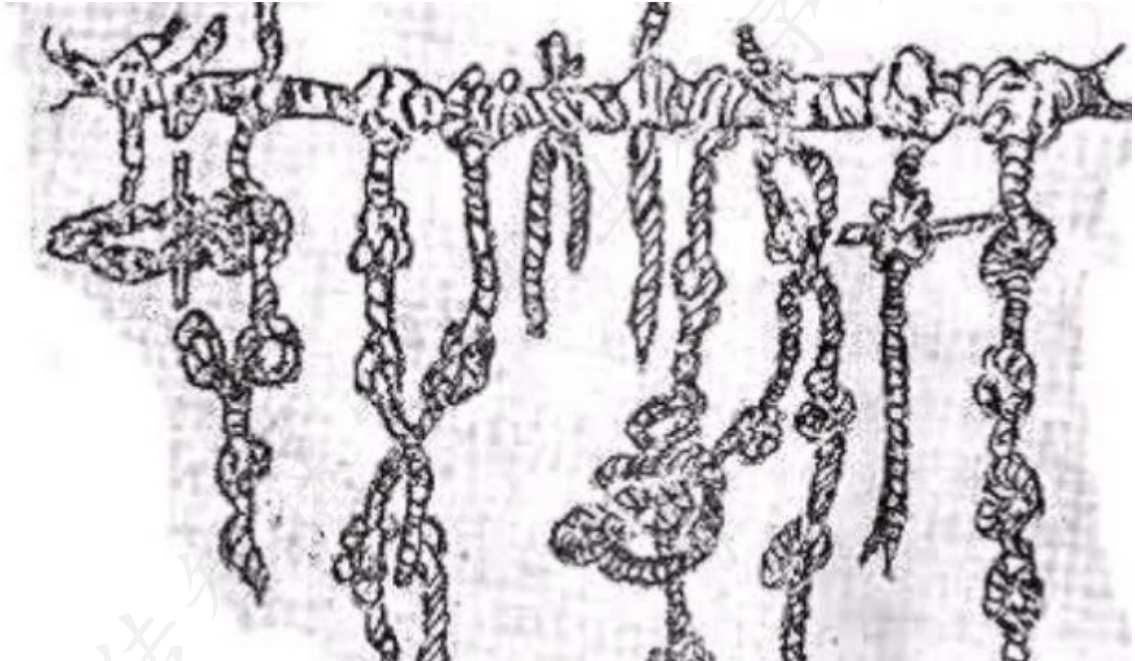
目标

- 知道什么是数据库
- 了解数据库的优点

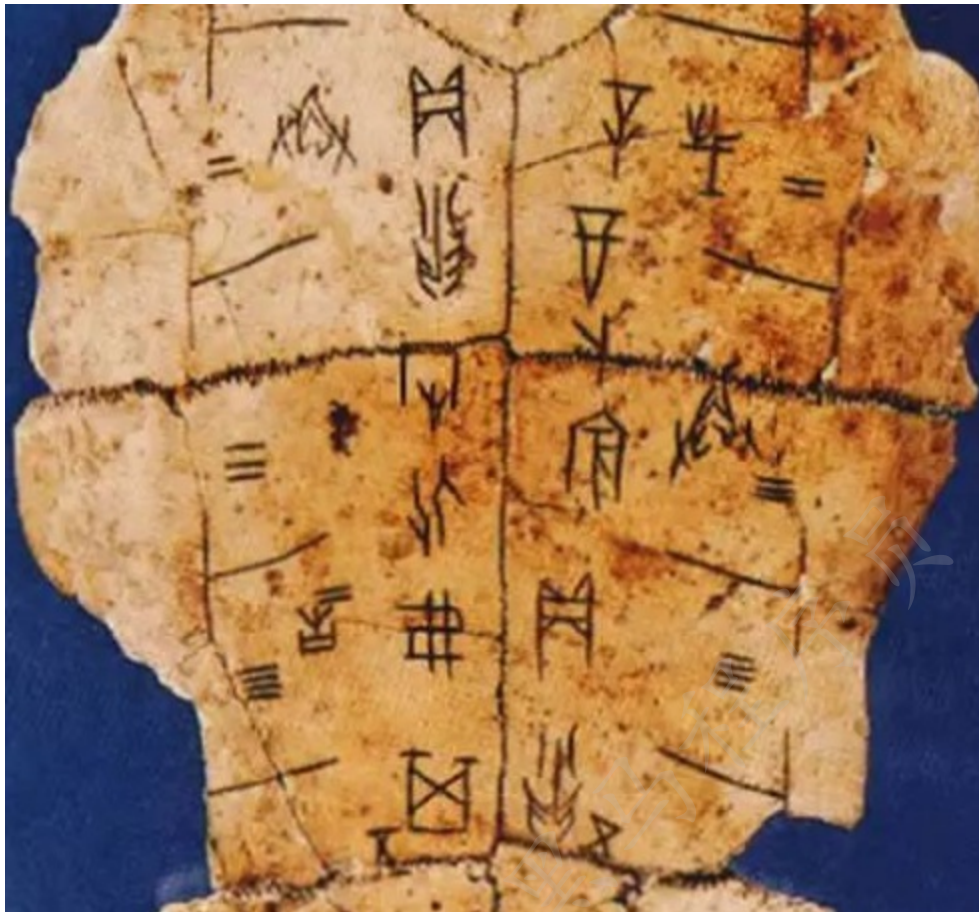
1. 数据存储的阶段发展

1.1 人工管理阶段

- 结绳记事



- 甲骨



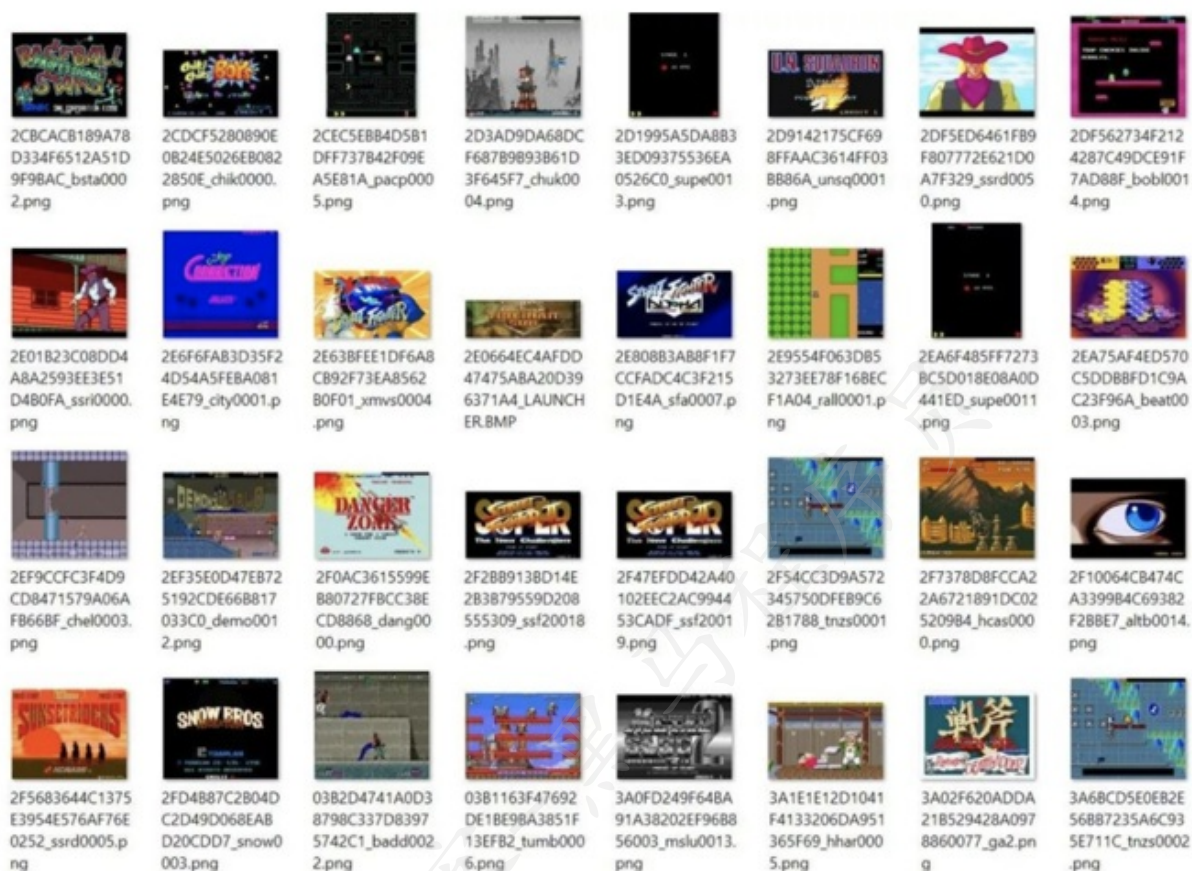
- 纸



人工管理阶段，数据都是靠人工进行整理和保存的，使用起来很不方便。不便于查询、共享、保存。

1.2 文件系统阶段

- ## ● 磁盘



文件系统阶段，将数据存储在磁盘上，数据都是以文件的形式出现，通过文件系统管理。

相对于人工管理阶段方便的多，但是在查询数据内容时，还不是很方便。

1.3 数据库系统阶段

数据库：是指长期存储在计算机内、有组织的数据集合。简而言之，数据库就是一个存储数据的地方。

表是数据库中存储数据的基本单位，数据按照分类存储到不同的表中，能够非常高效的查询其中的数据。

网页中显示的商品信息

优品推荐

大屏手机

双卡双待

畅销千元

三星 Galaxy C7 (SM-C7000)
¥1999.00

努比亚(nubia) [6+64GB] Z17mini
¥1999.00

荣耀8青春版 全网通 标配版
¥1099.00

中兴(ZTE) A2 珍珠银 全网通4G双卡
¥649.00

vivo Xplay6 全网通 6GB+128GB 玫
¥4498.00

小辣椒 红辣椒 Note4X 高配版 全网
¥849.00

数据库中存储的商品数据

```
mysql> select * from goods;
```

id	name	cate	price	is_saleoff	count	img_url
1	三星 Galaxy G7	双卡双待	1999.00		10	https://item.jd.com/4746242.html
2	努比亚	双卡双待	1999.00		20	https://item.jd.com/4746242.html
3	荣耀8青春版	双卡双待	1099.00		30	https://item.jd.com/4746242.html
4	中兴(ZTE) A2	双卡双待	649.00		15	https://item.jd.com/4746242.html
5	vivo Xplay	双卡双待	4498.00		20000	https://item.jd.com/4746242.html
6	小辣椒 红辣椒	双卡双待	849.00		200	https://item.jd.com/4746242.html

真实的商品



数据库分类

目标

- 能够说出数据库的分类
- 能够说出关系型数据中典型的数据库名称
- 能够说出关系型数据库的构成要素
- 能够说出非关系型数据中典型的数据库名称

1. 常见的数据库分类

当前主要使用两种类型的数据库：

- 关系型数据库
- 非关系型数据库

查看数据库排名：<https://db-engines.com/en/ranking>

Rank			DBMS	Database Model	Score		
Apr 2019	Mar 2019	Apr 2018			Apr 2019	Mar 2019	Apr 2018
1.	1.	1.	Oracle +	Relational, Multi-model i	1279.94	+0.80	-9.85
2.	2.	2.	MySQL +	Relational, Multi-model i	1215.14	+16.89	-11.26
3.	3.	3.	Microsoft SQL Server +	Relational, Multi-model i	1059.96	+12.11	-35.55
4.	4.	4.	PostgreSQL +	Relational, Multi-model i	478.72	+8.91	+83.25
5.	5.	5.	MongoDB +	Document	401.98	+0.64	+60.57
6.	6.	6.	IBM Db2 +	Relational, Multi-model i	176.05	-1.15	-12.89
7.	↑ 8.	↑ 9.	Redis +	Key-value, Multi-model i	146.38	+0.25	+16.27
8.	↑ 9.	8.	Elasticsearch +	Search engine, Multi-model i	146.00	+3.21	+14.64
9.	↓ 7.	↓ 7.	Microsoft Access	Relational	144.65	-1.55	+12.43
10.	10.	↑ 11.	SQLite +	Relational	124.21	-0.66	+8.23
11.	11.	↓ 10.	Cassandra +	Wide column	123.61	+0.81	+4.52
12.	12.	↑ 14.	MariaDB +	Relational, Multi-model i	85.23	+0.92	+20.67
13.	13.	13.	Splunk	Search engine	83.09	-0.01	+18.03
14.	14.	↓ 12.	Teradata +	Relational	75.35	+0.13	+1.67
15.	15.	↑ 18.	Hive +	Relational	74.71	+1.71	+17.31
16.	16.	↓ 15.	Solr	Search engine	60.22	+0.21	-2.99
17.	17.	17.	HBase	Wide column	58.66	-0.13	-1.03
18.	18.	↑ 19.	FileMaker	Relational	58.42	+0.30	+3.42
19.	↑ 21.	↑ 21.	Amazon DynamoDB +	Multi-model i	56.01	+1.52	+12.86
20.	↓ 19.	↓ 16.	SAP Adaptive Server	Relational	55.80	-0.23	-5.83

2. 关系型数据库

RDMS: (Relational Database Management System)关系型数据库系统

将数据间的关系以数据库表的形式加以表达，并将数据存储于表格中，以便于查询。

2.1 关系型数据库的代表产品

- Oracle: 在大型项目中使用，例如：银行、电信等项目

- MySQL: Web 项目中使用最广泛的关系型数据库
- Microsoft SQL Server: 在微软的项目中使用
- SQLite: 轻量级数据库, 主要应用在移动平台

2.2 关系型数据库的核心元素

- 数据行 (一条记录)
- 数据列 (字段)
- 数据表 (数据行的集合)
- 数据库 (数据表的集合, 一个数据库中能够有 n 多个数据表)

3. 非关系型数据库

- Redis
- MongoDB

SQL介绍

目标

- 了解SQL的含作用和分类

1. SQL介绍

SQL: Structured Query Language (结构化查询语言), 通过SQL语言可以对数据库进行操作

主流的关系型数据库都支持使用 SQL 语言进行操作, 也就是说可以通过 SQL 语言操作 Oracle、MySQL、Microsoft SQL Server、SQLite 等关系型的数据库。

1.1 SQL 语言的分类

- DQL: 数据查询语言, 用于对数据进行查询, 例如: **select**
- DML: 数据操作语言, 对数据进行增加、修改、删除, 例如: **insert**、**update**、**delete**
- TPL: 事务处理语言, 对事务进行处理, 例如: **begin transaction**、**commit**、**rollback**
- DCL: 数据控制语言, 进行授权与权限回收, 例如: **grant**、**revoke**
- DDL: 数据定义语言, 进行数据库、表的管理等, 例如: **create**、**drop**
- CCL: 指针控制语言, 通过控制指针完成表的操作, 例如: **declare cursor**

注意:

- 对于测试工程师来说, 重点是数据的查询, 因此需要熟练编写DQL, 其它类型, 如 TPL、DCL、CCL了解即可
- 在 MySQL 中, 默认对 SQL 语法不区分大小写

MySQL介绍

目标

- 知道MySQL的特点

1. MySQL简介

- MySQL 是一个关系型数据库管理系统，由瑞典 MySQL AB 公司开发，后来被 Sun 公司收购，Sun 公司后来又又被 Oracle 公司收购，目前属于 Oracle 旗下产品
- 目前为止，**MySQL** 社区版是可以免费使用的
- 官方网站：<http://www.mysql.com/>

1.1 MySQL特点

开源、社区版免费、支持多平台/多语言、使用范围广泛，是学习数据库开发与使用的首选！

- 使用 C 和 C++ 编写，并使用了多种编译器进行测试，保证源代码的可移植性
- 支持多种操作系统，如 Linux、Windows、AIX、macOS 等
- 为多种编程语言提供了 API，如 C、C++、Python、Java、Perl、PHP、Ruby 等
- 支持多线程，能够充分利用 CPU 资源
- 优化的 SQL 查询算法，有效地提高查询速度
- 提供多语言支持，常见的编码如 GB2312、BIG5、UTF8
- 提供 TCP/IP、ODBC 和 JDBC 等多种数据库连接途径
- 提供用于管理、检查、优化数据库操作的管理工具
- 大型的数据库。可以处理拥有上千万条记录的大型数据库
- 支持多种存储引擎
- MySQL 软件采用了双授权政策，它分为社区版和商业版，由于其体积小、速度快、总体拥有成本低，尤其是开放源码这一特点，一般中小型网站的开发都选择 MySQL 作为网站数据库
- MySQL 使用标准的 SQL 数据语言形式
- MySQL 是可以定制的，采用了 GPL 协议，你可以修改源码来开发自己的 MySQL 系统
- 在线 DDL 更改功能
- 复制全局事务标识
- 复制无崩溃从机
- 复制多线程从机

MySQL安装与使用

目标

- 掌握在 Windows 平台下 MySQL 的安装与使用步骤
- 掌握如何连接远程数据库
- 掌握如何连接本地数据库

1. MySQL 安装

详细安装步骤，请查看MySQL安装手册

2. 客户端与服务端搭配场景

在实际工作中，数据库的服务端是部署在服务器上的，我们要想使用数据库，就需要使用客户端远程连接数据库

- 客户端远程连接服务端（工作场景）
- 客户端本地连接服务端（教学场景）

2.1 客户端远程连接服务端（工作场景）

场景设置

- 两人一组，依次作为对方的 MySQL 服务端；
(A 同学先作为服务端，B 同学使用客户端远程连接 A 同学设备里的 MySQL，完成后互换)
- 使用 Navicat 模拟客户端远程连接服务端的 MySQL

连接实现步骤

服务端：

1. 作为服务端的同学，打开命令行窗口cmd，输入命令：`ipconfig`，按 回车键，查看自己电脑的 IP 地址
2. 将 以太网适配器 本地连接 下的 IP 地址，告知使用客户端的同学


```
管理员: C:\Windows\system32\cmd.exe

Microsoft Windows [版本 6.1.7601]
版权所有 (c) 2009 Microsoft Corporation。保留所有权利。

C:\Users\Administrator>ipconfig

Windows IP 配置

以太网适配器 Bluetooth 网络连接:

    媒体状态 . . . . . : 媒体已断开
    连接特定的 DNS 后缀 . . . . . :

以太网适配器 本地连接: 1

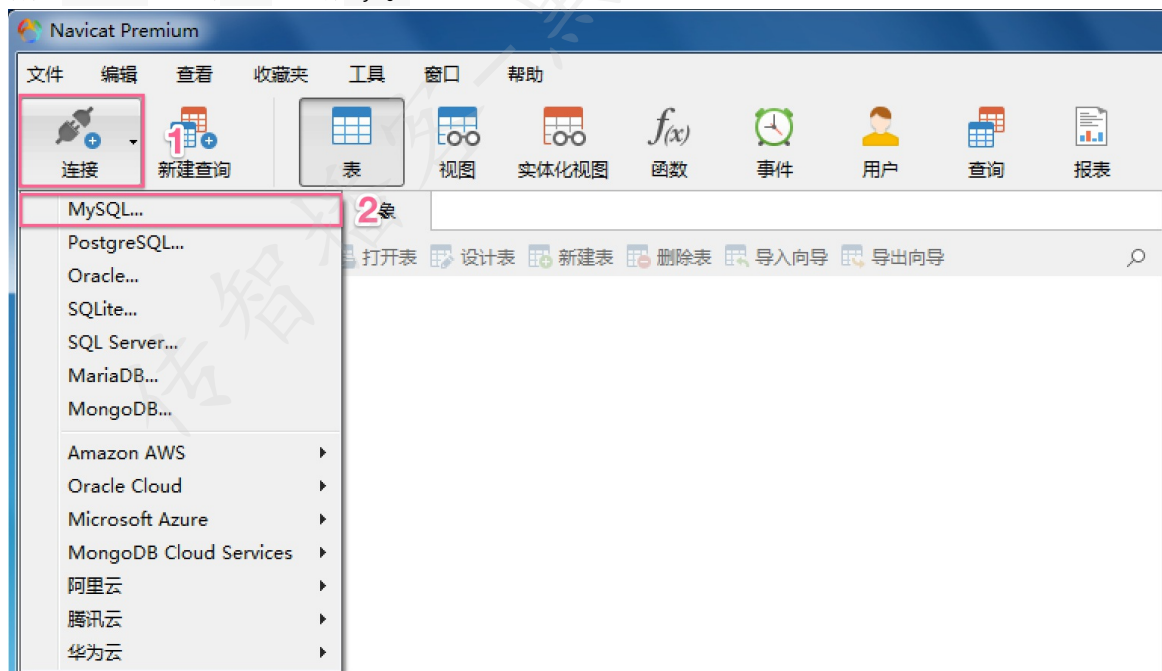
    连接特定的 DNS 后缀 . . . . . : localdomain
    本地连接 IPv6 地址 . . . . . : fe80::e429:382f:105e:6f6b%11
    IPv4 地址 . . . . . : 192.168.180.138 2
    子网掩码 . . . . . : 255.255.255.0
    默认网关 . . . . . : 192.168.180.2

以太网适配器 VMware Network Adapter VMnet1:

    连接特定的 DNS 后缀 . . . . . :
    本地连接 IPv6 地址 . . . . . : fe80::9401:a2b7:bd40:f063%18
```

客户端:

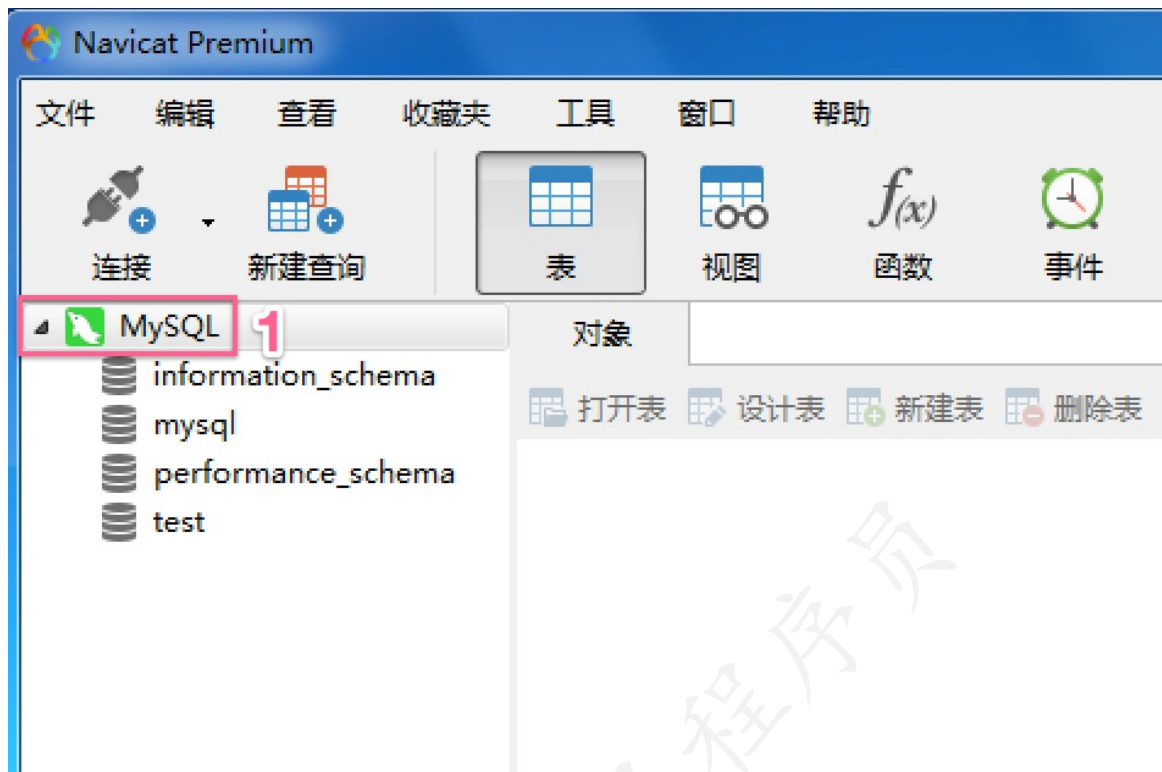
1. 打开 Navicat，点击 连接，选择 MySQL



2. 输入数据库连接信息

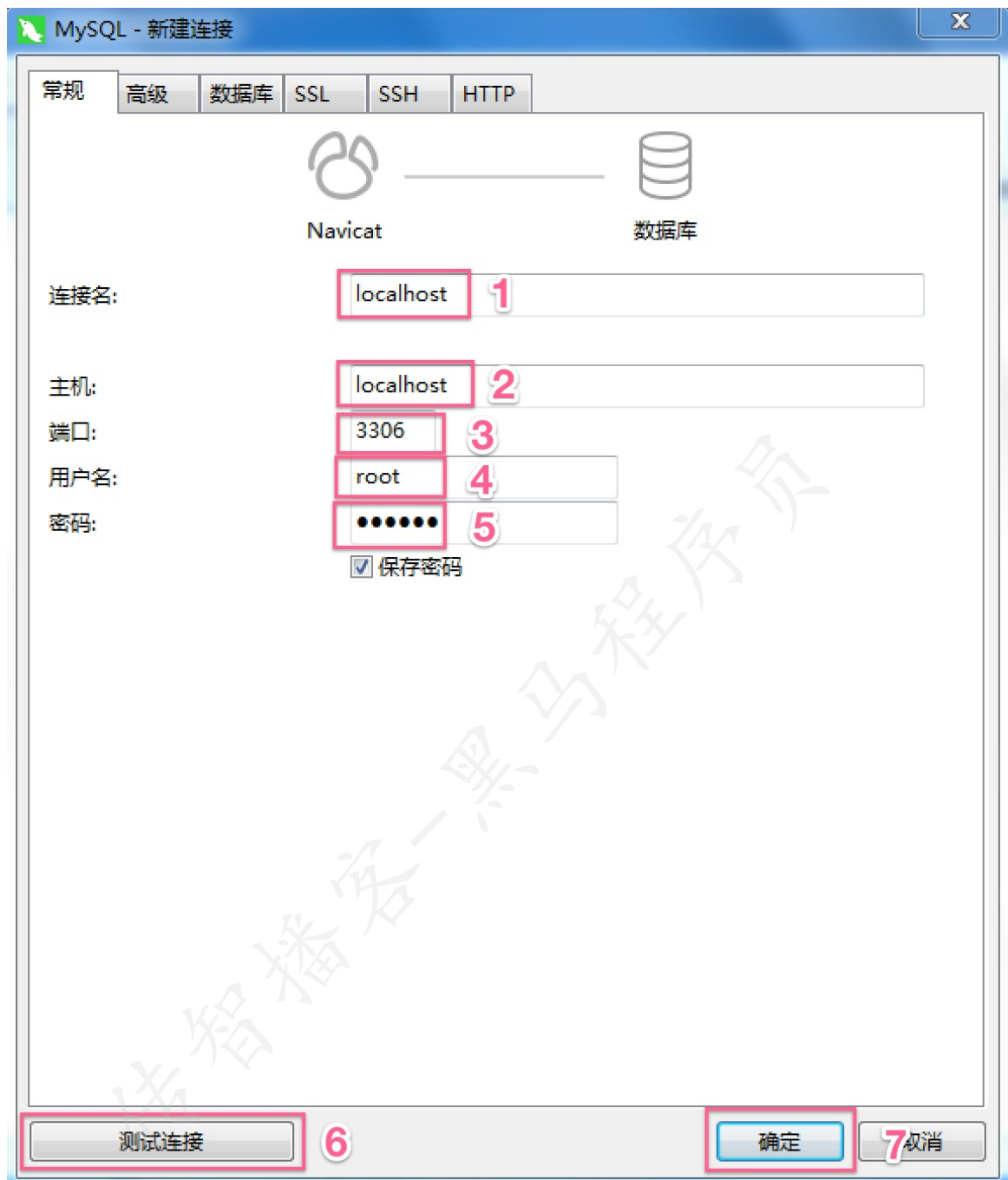


- i. 连接名称（连接目标标识，可自定义，例如：MySQL）
 - ii. MySQL 服务端 IP 地址（从担当服务端的同学那里获取）
 - iii. MySQL 服务端端口号（默认：3306）
 - iv. MySQL 服务端用户名（默认：root）
 - v. MySQL 服务端密码（与担当服务端的同学确认密码内容）
 - vi. 测试连接（点击 测试连接，提示 连接成功，再点击 确定，建立连接）
 - vii. 建立连接
3. 双击建立好的连接名字，如果字体颜色变为绿色，则意为远程连接成功！



2.2 客户端本地连接服务端（教学场景）

1. 打开 Navicat，点击 连接，选择 MySQL
2. 输入数据库连接信息



- i. 连接名称（localhost 意为本地连接）
- ii. 本地 IP 地址（localhost 或 127.0.0.1 都代表本地地址）
- iii. MySQL 本地端口号（默认：3306）
- iv. MySQL 本地用户名（默认：root）
- v. MySQL 本地密码（根据自己设定的填写）
- vi. 测试连接（提示 连接成功 后，点击 确定，建立连接）
- vii. 建立连接

3. 双击建立好的连接名字，如果字体颜色变为绿色，则意为远程连接成功！

3. 扩展-在 Linux 中配置 MySQL 中的 root 用户允许远程连接（了解）

注意：Linux 平台下的 MySQL 服务端，默认是不允许远程连接的。

在 Linux 中打开终端，依次输入下面命令：

```
mysql -uroot  
  
use mysql;  
  
update user set host='%' where host='::1';  
  
flush privileges;
```

```
[admin@localhost ~]$ mysql -u root  
Welcome to the MySQL monitor.  Commands end with ; or \g.  
Your MySQL connection id is 11  
Server version: 5.6.35 MySQL Community Server (GPL)  
  
Copyright (c) 2000, 2016, Oracle and/or its affiliates. All rights reserved.  
  
Oracle is a registered trademark of Oracle Corporation and/or its  
affiliates. Other names may be trademarks of their respective  
owners.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
  
mysql> use mysql;  
Reading table information for completion of table and column names  
You can turn off this feature to get a quicker startup with -A  
  
Database changed  
mysql> update user set host='%' where host='::1';  
Query OK, 1 row affected (0.00 sec)  
Rows matched: 1  Changed: 1  Warnings: 0  
  
mysql> flush privileges;  
Query OK, 0 rows affected (0.00 sec)  
  
mysql> █
```