

项目如何存储数据

- 集合可以存储数据，缺点：程序一旦重启，数据全部清空，不能做到持久化。
- XML文件存储数据，缺点：存取很麻烦，需要借助dom4j工具来解析XML文件，数据没有结构化的关联。
- 可以通过IO流将数据保存到本地硬盘，比如txt文档，缺点：需用通过IO流来操作，没有结构化的框架，数据之间的关联关系需要自己设计。

数据库可以解决上述的问题，数据库有非常完善存储，管理数据的结构化设计，同时可以做到持久化。

什么是数据库？

数据库（database）就是一个专门用来存储数据的仓库，为了方便数据的存储和管理，数据库将数据按照特定的规律存储在磁盘中。通过数据库管理系统，可以有效的组织和管理存储在数据库中的数据。

数据库是安装在计算机中的一个服务，启动该服务之后就可以使用它来存储和管理数据了。

MySQL, Oracle, SQLServer, MongoDB, Redis...

数据库的主要功能：1.保存。2.管理数据（修改，删除，查询，添加）

所以我们通常说的数据库全面来讲应该是数据库管理系统（DBMS），DataBase Management System

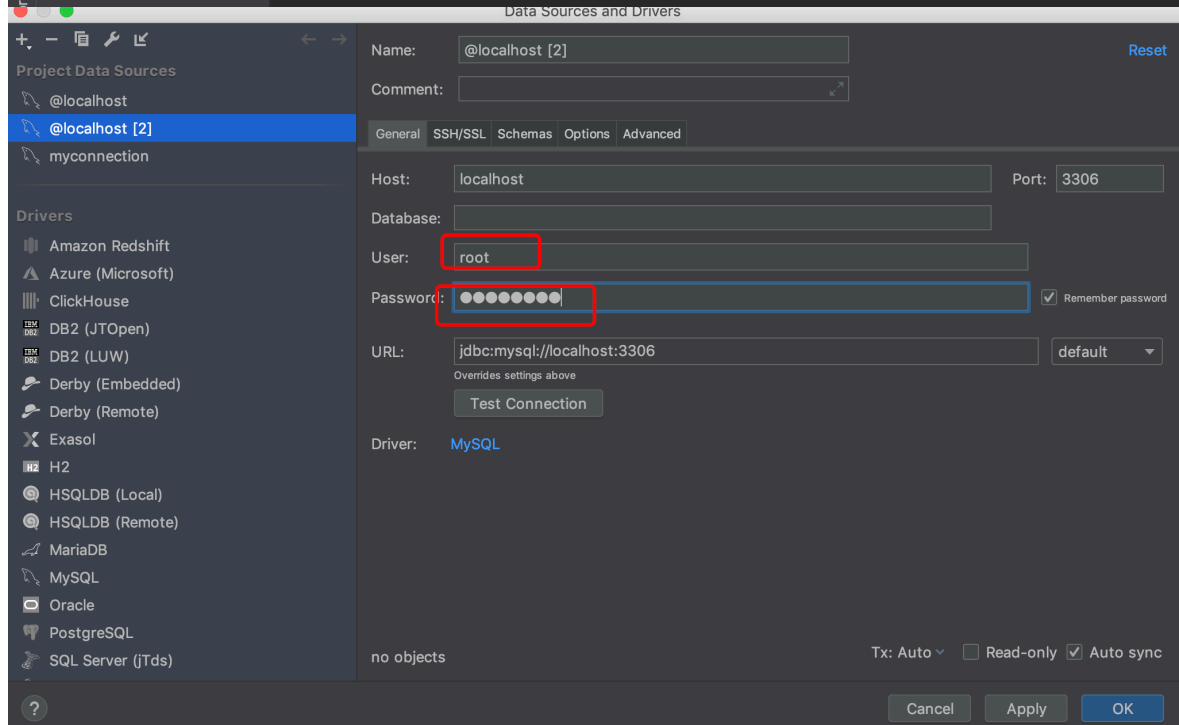
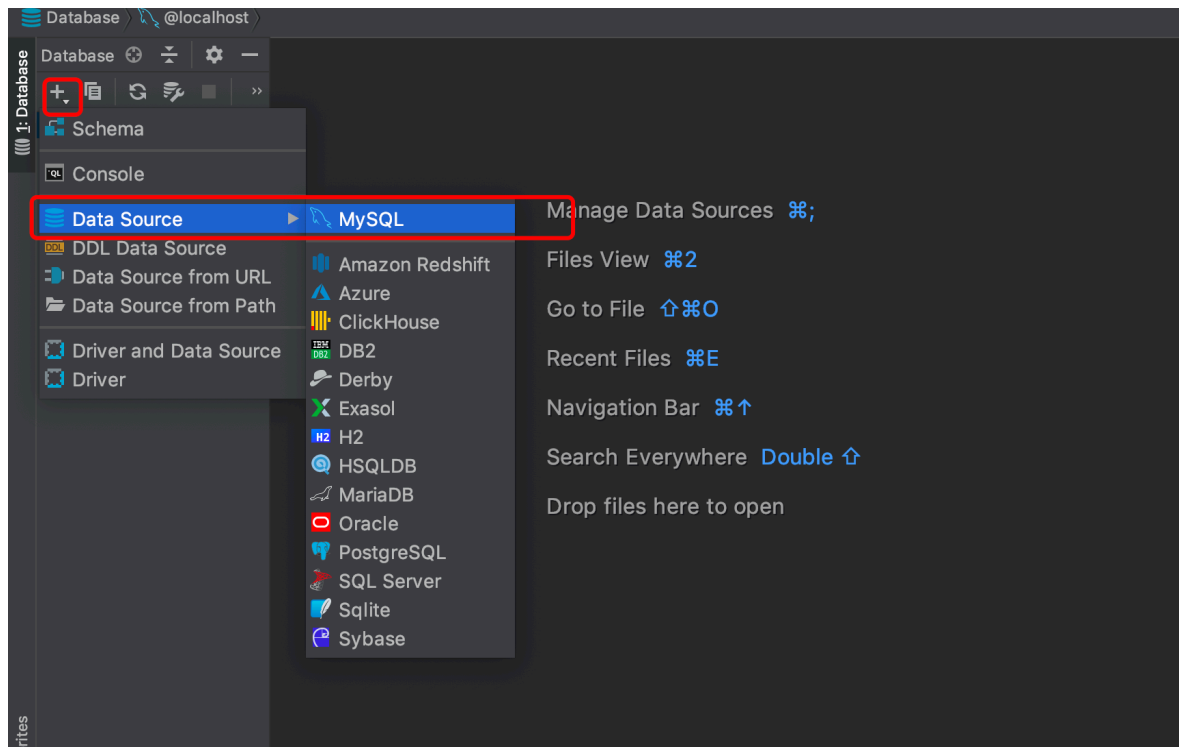
DataGrip, Navicator, IDEA

数据库存储引擎：

存储引擎其实就是如何存储数据，如何为存储的数据建立索引，如何更新，查询数据等技术的实现方法。关系型数据库中的数据是以表的形式存储的，所以存储引擎也可以成为表类型（存储和操作此表的类型）。

MySQL提供了多种存储引擎。

如何使用DataGrip



!

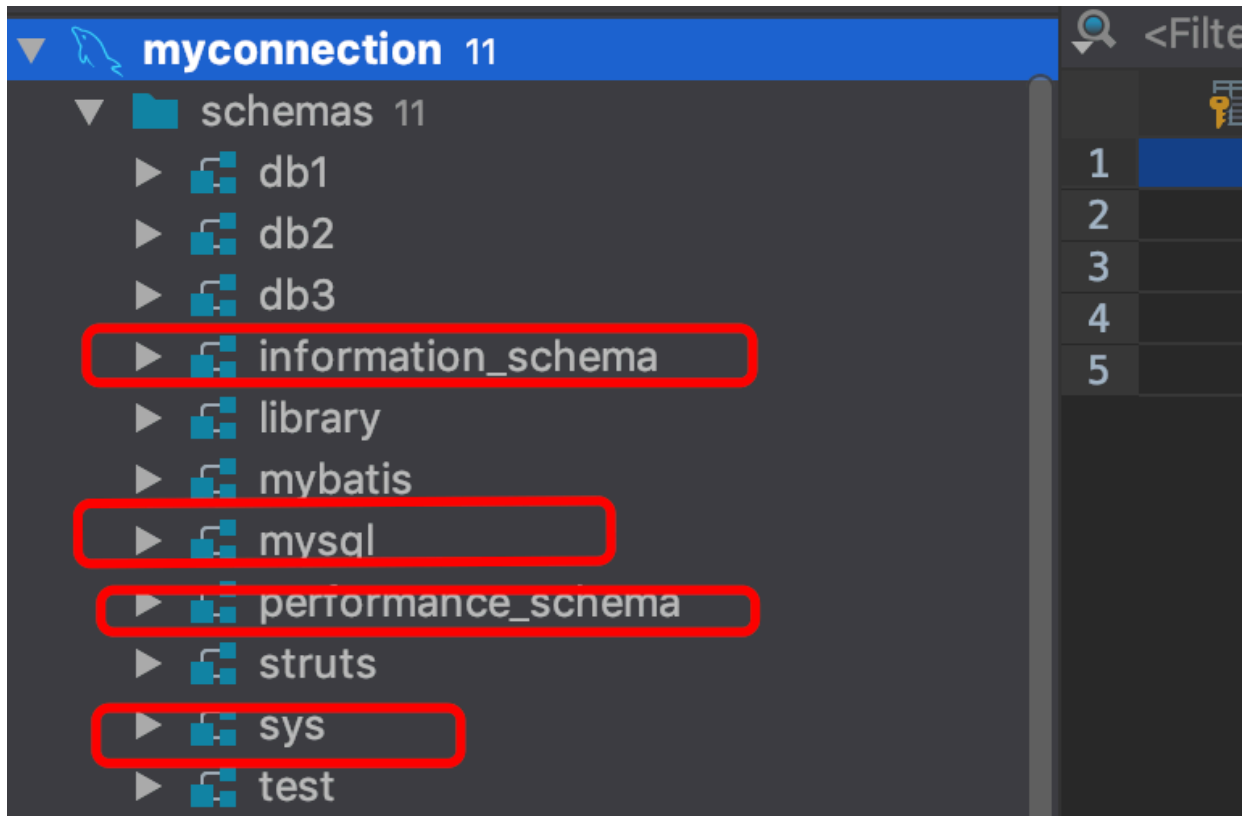
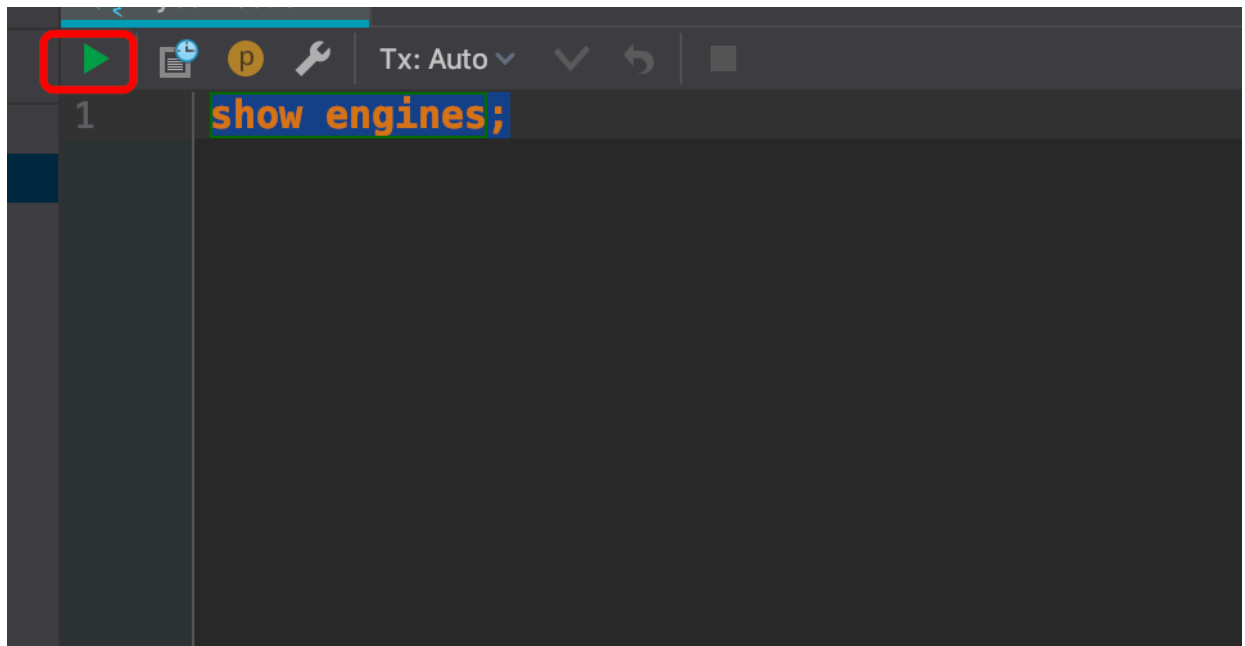


image-20181205203608382](/Users/southwind/Library/Application Support/typora-user-images/image-20181205203608382.png)

information_schema、performance_schema、sys、mysql四个数据库是MySQL自带的默认数据库，不要删除或修改。

查看MySQL支持的存储引擎：show engines; SQL语句



			Tab-se...d (TSV)			View Query
			Transactions	XA	Savepoints	
1	ARCHIVE	YES	Archive storage engine	NO	NO	NO
2	BLACKHOLE	YES	/dev/null storage engine (anything you write to it disa...	NO	NO	NO
3	MRG_MYISAM	YES	Collection of identical MyISAM tables	NO	NO	NO
4	FEDERATED	NO	Federated MySQL storage engine	<null>	<null>	<null>
5	MyISAM	YES	MyISAM storage engine	NO	NO	NO
6	PERFORMANCE_SCHEMA	YES	Performance Schema	NO	NO	NO
7	InnoDB	DEFAULT	Supports transactions, row-level locking, and foreign k...	YES	YES	YES
8	MEMORY	YES	Hash based, stored in memory, useful for temporary tabl...	NO	NO	NO
9	CSV	YES	CSV storage engine	NO	NO	NO

Engine：存储引擎的名称。

Support：MySQL是否支持该引擎。

Comment：引擎的描述。

Transactions：是否支持事务。

XA：是否支持分布式事务处理规范。

Savepoints：是否支持保存点，以便让事务进行回滚。

MySQL数据库默认使用InnoDB存储引擎，InnoDB对事务的处理能力非常强大，是其他引擎所不能比拟。

InnoDB存储引擎支持自增。自增列的值不能为空，且必须唯一，MySQL中规定自增列必须是主键。

InnoDB存储引擎支持外键。

InnoDB优势在于提供了良好的事务管理能力，崩溃修复能力和并发控制。

InnoDB缺点是读写效率较差，占用的数据空间相对比较大。

如何来管理数据库？

通过SQL语言来管理数据库，SQL语言也是一种编程语言，专门用来管理数据库的，包括数据库的创建，删除，数据表的创建，删除，修改，数据的创建，删除，修改，查询。

数据库中包含数据表，数据表中包含数据。

SQL语言功能：

- DML（数据操作语言）操作数据库中存储的数据（insert, update, delete）
- DDL（数据定义语言）创建，删除，修改数据库，数据表（create, drop, alter）
- DQL（数据查询语言）对数据库中存储的数据进行查询（select）
- DCL（数据控制语言）用来控制数据库组件的存取（事务）（commit, rollback）

学习数据库，需要掌握两部分内容

1.掌握数据库的使用（SQL，DataGrip）

2.设计数据库：根据项目的需求，设计数据表与数据表之间的关联关系，数据库设计非常重要，直接决定项目的质量，数据库设计驱动项目的开发。

数据库的组成：

一个数据库由多张数据表组成，数据是保存在数据表中的。

数据表由两部分内容组成

- 表结构：表设计 //列信息
- 表数据：表中所保存的数据 //行信息

数据表与Java的映射关系：

数据表和Java的类关联，数据表中的记录和Java类的实例化对象关联。

--》创建数据库

语法：

create database 数据库名称 default character set utf8 collate utf8_general_ci;

create database 数据库名称 default character set utf8 collate utf8_bin;

排序规则：utf8_general_ci, utf8_bin

utf8_general_ci: 不区分大小写。a,B

utf8_bin: 区分大小写。B,a

a:97 B:66 B,a

```
create database mstest1 default character set utf8
collate utf8_general_ci;
create database mstest2 default character set utf8
collate utf8_bin;
```

--》创建数据表

语法：

create table 数据表名称(

字段 数据类型

字段 数据类型

...

)

```
use mstest1;  
create table user(  
    id int,  
    name varchar(11),  
    password varchar(20),  
    age int  
);
```

字段的数据类型：

整数类型：

- tinyint (1个byte)
- smallint (2个byte)
- mediumint (3个byte)
- int (4个byte)
- bigint (8个byte)

最常用的是int