

# 实验 12：调用接口实现数据操作

## 一、实验目的

- 1、掌握 APIJSON 零代码接口的本地部署和配置
- 2、理解 RESTful API 设计原则和 JSON 协议规范
- 3、熟练使用 APIJSON 实现复杂数据查询和关联操作
- 4、培养数据库设计和接口调试能力

## 二、实验学时

2 学时

## 三、实验类型

综合性

## 四、实验需求

### 1、硬件

每人配备计算机 1 台，建议优先使用个人计算机开展实验。

### 2、软件

安装 Visual Studio Code，以及 Edge 浏览器。

### 3、网络

本地主机能够访问互联网和实验中心网络。

### 4、工具

无。

## 五、实验任务

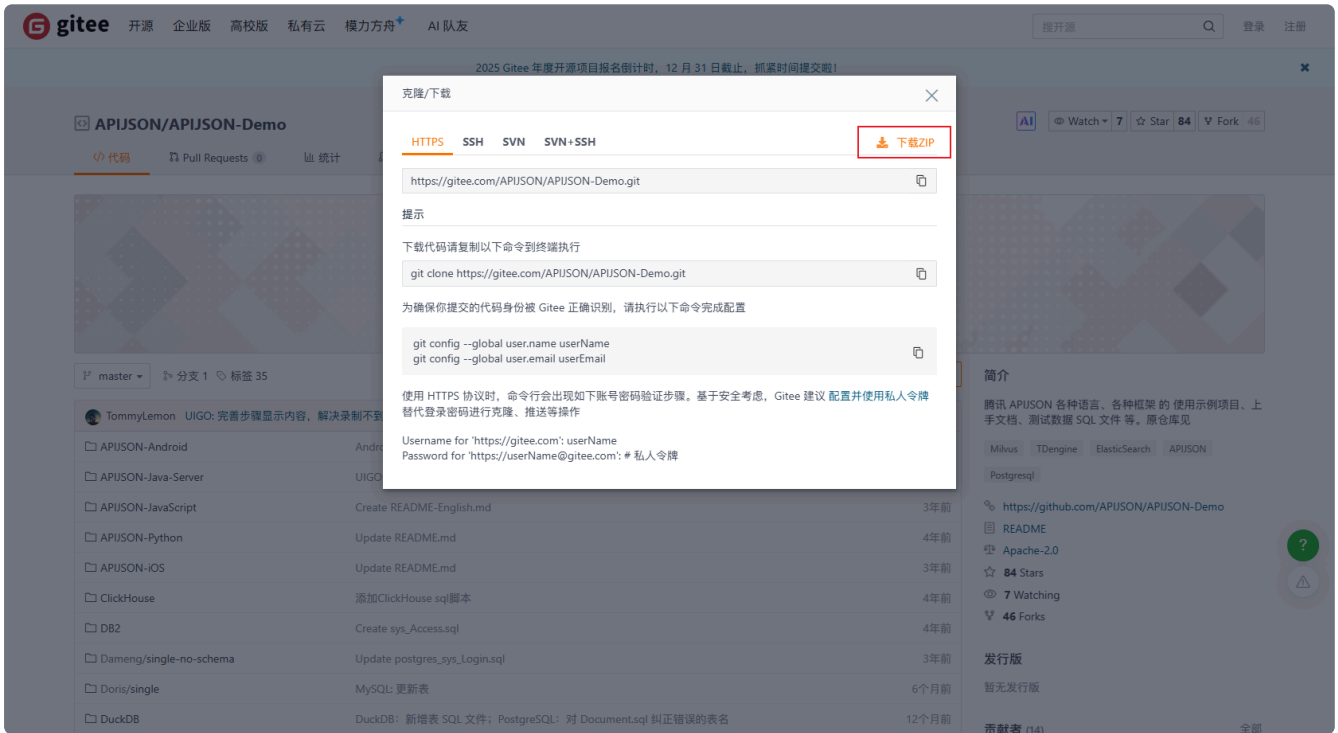
- 1、下载并运行 APIJSON 服务
- 2、设计中医药数据表结构并插入测试数据
- 3、接口测试

## 六、实验内容及步骤

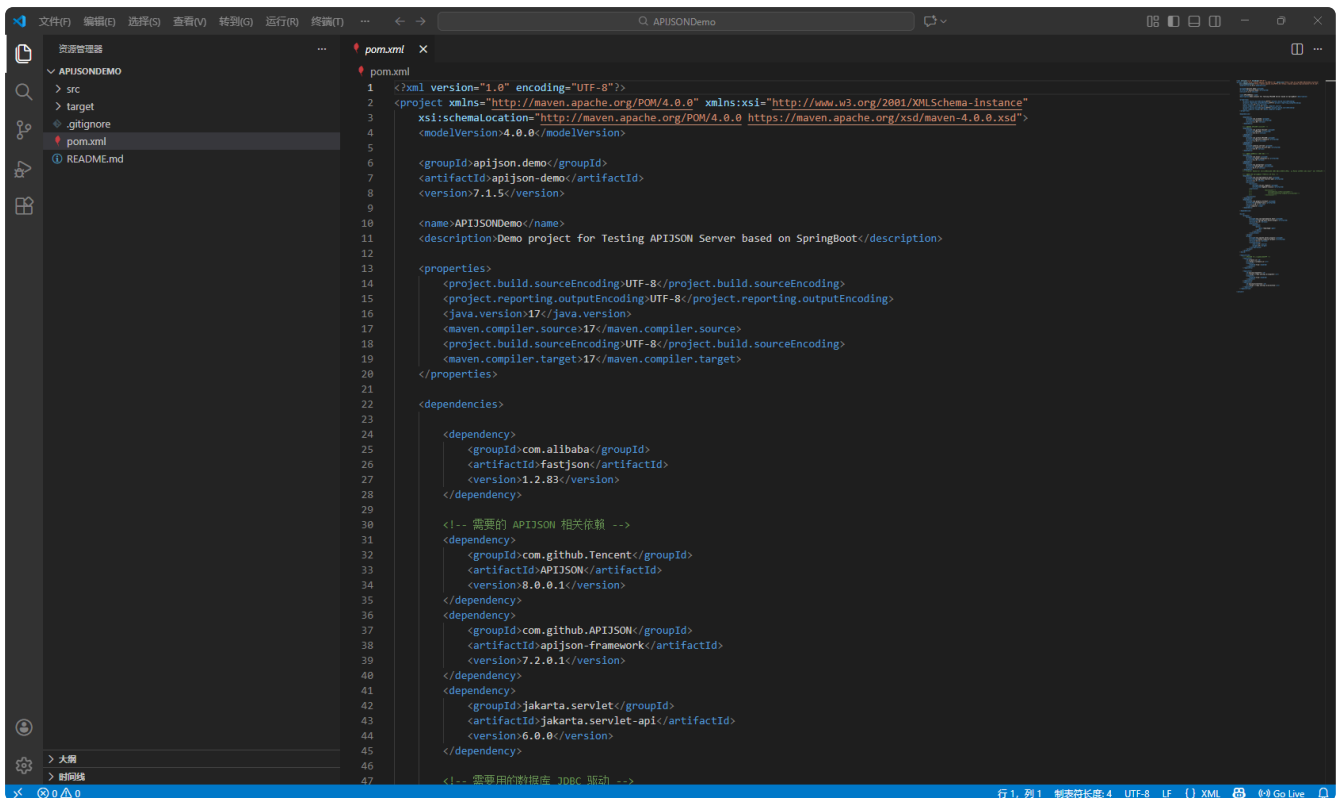
### 1、下载并运行 APIJSON 服务

步骤 1：环境准备与项目下载

访问 APIJSON 的官网 (<http://apijson.cn/>) 下载 APIJSON 项目 (<https://gitee.com/APIJSON/APIJSON-Demo>)。



步骤 2：使用 Visual Code 打开 Java 项目：APIJSON-Demo-master\APIJSON-Java-Server\APIJSONDemo



步骤 3: 自行完成 MySQL 数据库的安装并创建相关数据库, 参考命令如下。

#### Bash

```
1 -- 创建数据库
2 CREATE DATABASE tcm_api CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_c
  i;
3
4 -- 创建专用用户
5 CREATE USER 'tcm_user'@'localhost' IDENTIFIED BY 'Tcm123456';
6 GRANT ALL PRIVILEGES ON tcm_api.* TO 'tcm_user'@'localhost';
7 FLUSH PRIVILEGES;
8
9 -- 使用数据库
10 USE tcm_api;
```

步骤 4: 导入 APIJSON 默认数据, SQL 文件目录为: APIJSON-Demo-master\MySQL\sys.sql。

步骤 5: 修订 DemoSQLConfig.java 的配置

## Java

```
1  /*Copyright ©2016 TommyLemon(https://github.com/TommyLemon/APIJSON)
2
3  Licensed under the Apache License, Version 2.0 (the "License");
4  you may not use this file except in compliance with the License.
5  You may obtain a copy of the License at
6
7      http://www.apache.org/licenses/LICENSE-2.0
8
9  Unless required by applicable law or agreed to in writing, software
10 distributed under the License is distributed on an "AS IS" BASIS,
11 WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
12 See the License for the specific language governing permissions and
13 limitations under the License.*/
14
15 package apijson.demo;
16
17 import com.alibaba.fastjson.JSONArray;
18 import com.alibaba.fastjson.JSONObject;
19 import com.alibaba.fastjson.annotation.JSONField;
20
21 import apijson.framework.APIJSONSQLConfig;
22
23
24 /**SQL 配置
25  * TiDB 用法和 MySQL 一致
26  * 具体见详细的说明文档 C.开发说明 C-1-1.修改数据库链接
27  * https://github.com/Tencent/APIJSON/blob/master/%E8%AF%A6%E7%BB%86%E7%9A%84%E8%AF%B4%E6%98%8E%E6%96%87%E6%A1%A3.md#c-1-1%E4%BF%AE%E6%94%B9%E6%95%B0%E6%8D%AE%E5%BA%93%E9%93%BE%E6%8E%A5
28  * @author Lemon
29  */
30 public class DemoSQLConfig extends APIJSONSQLConfig<Long, JSONObject, JSONArray> {
31
32     static {
33         DEFAULT_DATABASE = DATABASE_MYSQL; // TODO 默认数据库类型,改成你自己的
34         DEFAULT_SCHEMA = "tcm_api"; // TODO 默认数据库名/模式,改成你自己的,默认情况是 MySQL: sys, PostgreSQL: public, SQL Server: dbo, Oracle:
35
36         // 表名和数据库不一致的,需要配置映射关系。只使用 APIJSONORM 时才需要;
```

```

37         // 如果用了 apijson-framework 且调用了 APIJSONApplication.init 则不
        需要
38         // (间接调用 DemoVerifier.init 方法读取数据库 Access 表来替代手动输入
        配置)。
39         // 但如果 Access 这张表的对外表名与数据库实际表名不一致，仍然需要这里注
        册。例如
40         //         TABLE_KEY_MAP.put(Access.class.getSimpleName(), "acce
        ss");
41
42         //表名映射，隐藏真实表名，对安全要求很高的表可以这么做
43         // TABLE_KEY_MAP.put("User", "apijson_user");
44         // TABLE_KEY_MAP.put("Privacy", "apijson_privacy");
45     }
46
47     @Override
48     public String gainDBVersion() {
49         return "9.1.0"; // "8.0.11"; // TODO 改成你自己的 MySQL 或 Postg
        reSQL 数据库版本号 // MYSQL 8 和 7 使用的 JDBC 配置不一样
50     }
51
52     @JSONField(serialize = false) // 不在日志打印 账号/密码 等敏感信息
53     @Override
54     public String gainDBUri() {
55         return "jdbc:mysql://1.95.11.68:33067?serverTimezone=GMT%2B8&us
        eUnicode=true&characterEncoding=UTF-8"; // TODO 改成你自己的，TiDB 可以当
        成 MySQL 使用，默认端口为 4000
56     }
57
58     @JSONField(serialize = false) // 不在日志打印 账号/密码 等敏感信息
59     @Override
60     public String gainDBAccount() {
61         return "tcm_user"; // TODO 改成你自己的
62     }
63
64     @JSONField(serialize = false) // 不在日志打印 账号/密码 等敏感信息
65     @Override
66     public String gainDBPassword() {
67         return "Tcm123456"; // TODO 改成你自己的，TiDB 可以当成 MySQL 使
        用，默认密码为空字符串 ""
68     }
69
70 }
71

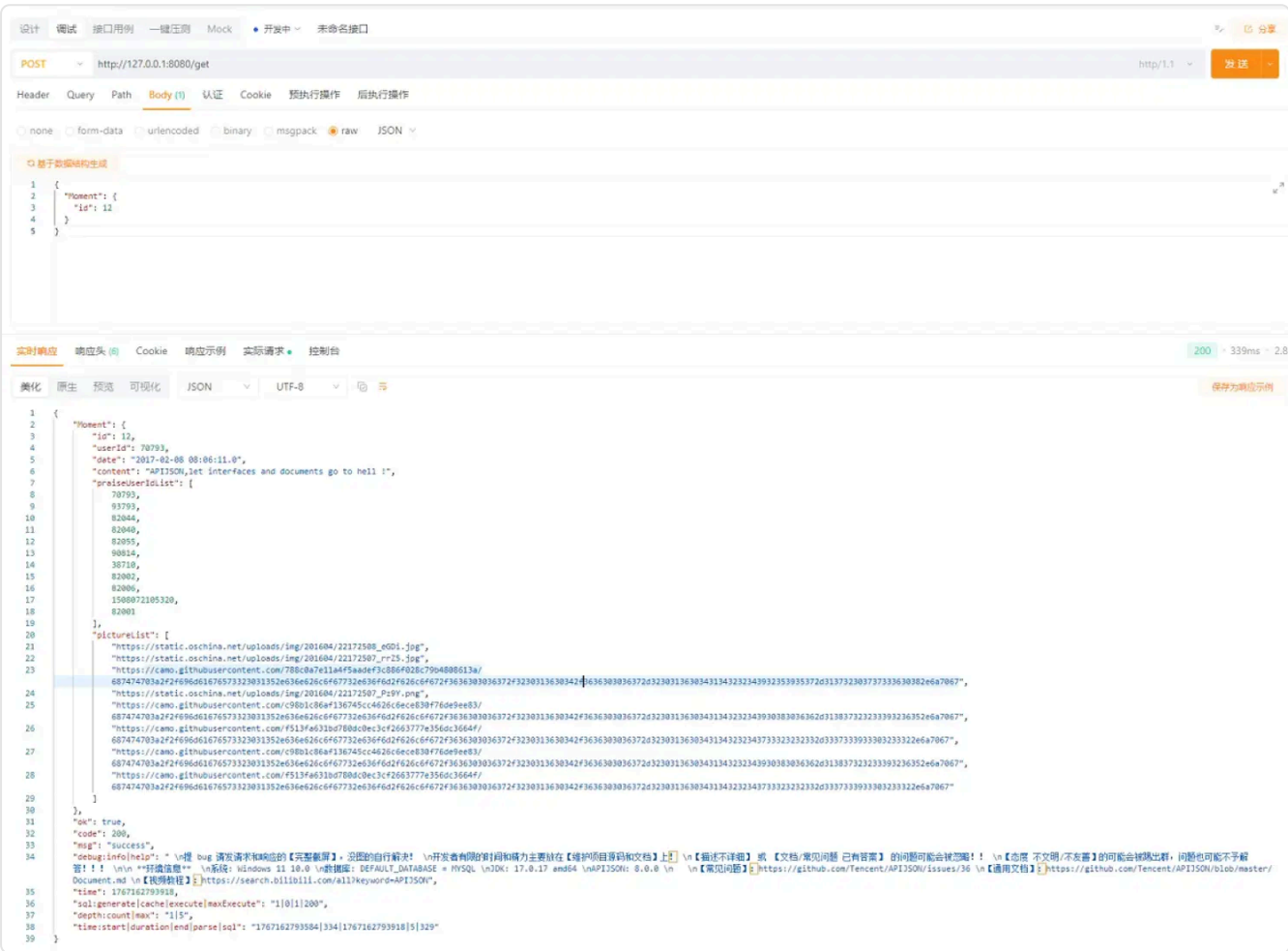
```

## 步骤 6: 指定 mvn 命令启动项目

```
Bash

1 mvn spring-boot:run
```

## 步骤 7: 使用接口测试工具测试接口服务



## 2、数据库部署与配置

步骤 1: 创建中医药数据表结构，参考 SQL 如下。

## SQL

```
1 -- 药材分类表
2 CREATE TABLE `HerbCategory` (
3   `id` bigint(15) NOT NULL AUTO_INCREMENT,
4   `name` varchar(50) NOT NULL COMMENT '分类名称',
5   `description` text COMMENT '分类描述',
6   `status` tinyint(1) DEFAULT '1' COMMENT '状态: 0-禁用, 1-启用',
7   `create_time` timestamp DEFAULT CURRENT_TIMESTAMP,
8   `update_time` timestamp DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_T
  IMESTAMP,
9   PRIMARY KEY (`id`),
10  UNIQUE KEY `uk_name` (`name`)
11 ) ENGINE=InnoDB COMMENT='药材分类表';
12
13 -- 中药材表
14 CREATE TABLE `ChineseHerb` (
15   `id` bigint(15) NOT NULL AUTO_INCREMENT,
16   `name` varchar(50) NOT NULL COMMENT '药材名称',
17   `pinyin` varchar(100) COMMENT '拼音名称',
18   `latin_name` varchar(100) COMMENT '拉丁名',
19   `category_id` bigint(15) NOT NULL COMMENT '分类ID',
20   `nature` enum('寒','热','温','凉','平') COMMENT '药性',
21   `flavor` json COMMENT '药味数组, 如: ["甘","苦"]',
22   `channel_tropism` json COMMENT '归经数组, 如: ["肺","胃"]',
23   `efficacy` text COMMENT '功效主治',
24   `usage_dosage` varchar(200) COMMENT '用法用量',
25   `contraindications` text COMMENT '禁忌注意事项',
26   `processing_method` varchar(100) COMMENT '炮制方法',
27   `storage_method` varchar(100) COMMENT '贮藏方法',
28   `status` tinyint(1) DEFAULT '1' COMMENT '状态',
29   `create_time` timestamp DEFAULT CURRENT_TIMESTAMP,
30   `update_time` timestamp DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_T
  IMESTAMP,
31   PRIMARY KEY (`id`),
32   UNIQUE KEY `uk_name` (`name`),
33   KEY `idx_category` (`category_id`),
34   KEY `idx_nature` (`nature`),
35   CONSTRAINT `fk_herb_category` FOREIGN KEY (`category_id`) REFERENCES
  `herb_category` (`id`)
36 ) ENGINE=InnoDB COMMENT='中药材表';
37
38 -- 方剂表
39 CREATE TABLE `Prescription` (
```

```

40 `id` bigint(15) NOT NULL AUTO_INCREMENT,
41 `name` varchar(100) NOT NULL COMMENT '方剂名称',
42 `pinyin` varchar(150) COMMENT '拼音名称',
43 `source` varchar(200) COMMENT '方剂来源',
44 `composition` text COMMENT '组成',
45 `preparation` text COMMENT '制法',
46 `efficacy` text COMMENT '功效',
47 `indication` text COMMENT '主治',
48 `usage` text COMMENT '用法',
49 `precautions` text COMMENT '注意事项',
50 `modern_application` text COMMENT '现代应用',
51 `status` tinyint(1) DEFAULT '1' COMMENT '状态',
52 `create_time` timestamp DEFAULT CURRENT_TIMESTAMP,
53 `update_time` timestamp DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_T
    IMESTAMP,
54 PRIMARY KEY (`id`),
55 UNIQUE KEY `uk_name` (`name`)
56 ) ENGINE=InnoDB COMMENT='方剂表';
57
58 -- 方剂组成表 (关联表)
59 CREATE TABLE `PrescriptionHerb` (
60 `id` bigint(15) NOT NULL AUTO_INCREMENT,
61 `prescription_id` bigint(15) NOT NULL COMMENT '方剂ID',
62 `herb_id` bigint(15) NOT NULL COMMENT '药材ID',
63 `dosage` varchar(50) COMMENT '用量',
64 `processing_note` varchar(100) COMMENT '炮制说明',
65 `role` enum('君','臣','佐','使') COMMENT '在方剂中的角色',
66 `sequence` int COMMENT '排序序号',
67 `notes` text COMMENT '备注',
68 PRIMARY KEY (`id`),
69 UNIQUE KEY `uk_prescription_herb` (`prescription_id`, `herb_id`),
70 KEY `idx_herb` (`herb_id`),
71 CONSTRAINT `fk_ph_prescription` FOREIGN KEY (`prescription_id`) REFER
    ENCES `prescription` (`id`),
72 CONSTRAINT `fk_ph_herb` FOREIGN KEY (`herb_id`) REFERENCES `chinese_h
    erb` (`id`)
73 ) ENGINE=InnoDB COMMENT='方剂组成表';
74
75 -- 功效分类表
76 CREATE TABLE `EfficacyCategory` (
77 `id` bigint(15) NOT NULL AUTO_INCREMENT,
78 `name` varchar(50) NOT NULL COMMENT '功效分类名称',
79 `parent_id` bigint(15) DEFAULT '0' COMMENT '父级ID',
80 `level` int DEFAULT '1' COMMENT '层级',

```

```

81 `description` text COMMENT '描述',
82 `status` tinyint(1) DEFAULT '1' COMMENT '状态',
83 `create_time` timestamp DEFAULT CURRENT_TIMESTAMP,
84 PRIMARY KEY (`id`),
85 KEY `idx_parent` (`parent_id`)
86 ) ENGINE=InnoDB COMMENT='功效分类表';
87
88 -- 药材功效关联表
89 CREATE TABLE `HerbEfficacy` (
90 `id` bigint(15) NOT NULL AUTO_INCREMENT,
91 `herb_id` bigint(15) NOT NULL COMMENT '药材ID',
92 `efficacy_id` bigint(15) NOT NULL COMMENT '功效ID',
93 `intensity` enum('强','中','弱') DEFAULT '中' COMMENT '功效强度',
94 `notes` text COMMENT '备注说明',
95 PRIMARY KEY (`id`),
96 UNIQUE KEY `uk_herb_efficacy` (`herb_id`, `efficacy_id`),
97 KEY `idx_efficacy` (`efficacy_id`),
98 CONSTRAINT `fk_he_herb` FOREIGN KEY (`herb_id`) REFERENCES `chinese_h
  erb` (`id`),
99 CONSTRAINT `fk_he_efficacy` FOREIGN KEY (`efficacy_id`) REFERENCES `e
  fficacy_category` (`id`)
100 ) ENGINE=InnoDB COMMENT='药材功效关联表';

```

步骤2：插入测试数据，参考SQL如下。

## SQL

```
1 -- 插入药材分类数据
2 INSERT INTO `HerbCategory` (`name`, `description`) VALUES
3 ('解表药', '主要用于治疗外感表证的药物'),
4 ('清热药', '主要用于治疗里热证的药物'),
5 ('泻下药', '主要用于通利大便、攻逐水饮的药物'),
6 ('祛风湿药', '主要用于治疗风湿痹痛的药物'),
7 ('化湿药', '主要用于化湿运脾的药物'),
8 ('利水渗湿药', '主要用于通利水道、渗泄水湿的药物'),
9 ('温里药', '主要用于温散里寒的药物'),
10 ('理气药', '主要用于疏理气机的药物'),
11 ('消食药', '主要用于消化食积的药物'),
12 ('驱虫药', '主要用于驱除或杀灭寄生虫的药物'),
13 ('止血药', '主要用于制止体内外出血的药物'),
14 ('活血化瘀药', '主要用于通行血脉、消散瘀血的药物'),
15 ('化痰止咳平喘药', '主要用于祛痰或消痰、减轻或制止咳嗽气喘的药物'),
16 ('安神药', '主要用于安定神志的药物'),
17 ('平肝息风药', '主要用于治疗肝阳上亢或肝风内动病证的药物'),
18 ('开窍药', '主要用于治疗闭证神昏的药物'),
19 ('补虚药', '主要用于补益正气、增强体质的药物'),
20 ('收涩药', '主要用于收敛固涩的药物'),
21 ('涌吐药', '主要用于促使呕吐的药物'),
22 ('攻毒杀虫止痒药', '主要用于治疗疮痍疥癣等病证的药物');
23
24 -- 插入功效分类数据
25 INSERT INTO `EfficacyCategory` (`name`, `parent_id`, `level`, `description`) VALUES
26 ('解表', 0, 1, '发发表邪、解除表证'),
27 ('发散风寒', 1, 2, '性味辛温, 发散风寒'),
28 ('发散风热', 1, 2, '性味辛凉, 发散风热'),
29 ('清热', 0, 1, '清解里热'),
30 ('清热泻火', 4, 2, '清气分热, 泻火解毒'),
31 ('清热燥湿', 4, 2, '清热燥湿, 用于湿热证'),
32 ('清热解毒', 4, 2, '清热解毒, 用于热毒证'),
33 ('清热凉血', 4, 2, '清营分、血分热邪'),
34 ('清虚热', 4, 2, '清虚热, 用于阴虚发热'),
35 ('补虚', 0, 1, '补益人体气血阴阳'),
36 ('补气', 9, 2, '补益脾气、肺气等'),
37 ('补血', 9, 2, '补益血液'),
38 ('补阴', 9, 2, '滋补阴液'),
39 ('补阳', 9, 2, '温补肾阳');
40
41 -- 插入中药材数据
```

```

42 INSERT INTO `ChineseHerb` (`name`, `pinyin`, `category_id`, `nature`, `
    flavor`, `channel_tropism`, `efficacy`, `usage_dosage`) VALUES
43 ('麻黄', 'Mahuang', 1, '温', '['辛', '微苦']', '['肺', '膀胱']', '发汗解表,
    宣肺平喘, 利水消肿', '煎服, 2-10g'),
44 ('桂枝', 'Guizhi', 1, '温', '['辛', '甘']', '['心', '肺', '膀胱']', '发汗解
    肌, 温通经脉, 助阳化气', '煎服, 3-10g'),
45 ('金银花', 'Jinyinhua', 2, '寒', '['甘']', '['肺', '心', '胃']', '清热解毒,
    疏散风热', '煎服, 6-15g'),
46 ('黄连', 'Huanglian', 2, '寒', '['苦']', '['心', '脾', '胃', '肝', '胆',
    '大肠']', '清热燥湿, 泻火解毒', '煎服, 2-5g'),
47 ('人参', 'Renshen', 17, '温', '['甘', '微苦']', '['脾', '肺', '心']', '大补
    元气, 复脉固脱, 补脾益肺, 生津养血, 安神益智', '煎服, 3-9g; 挽救虚脱可用15-30g'),
48 ('黄芪', 'Huangqi', 17, '温', '['甘']', '['肺', '脾']', '补气升阳, 固表止汗,
    利水消肿, 生津养血', '煎服, 9-30g'),
49 ('当归', 'Danggui', 17, '温', '['甘', '辛']', '['肝', '心', '脾']', '补血活
    血, 调经止痛, 润肠通便', '煎服, 6-12g');
50
51 -- 插入方剂数据
52 INSERT INTO `Prescription` (`name`, `pinyin`, `source`, `composition`,
    `efficacy`, `indication`) VALUES
53 ('麻黄汤', 'Mahuang Tang', '《伤寒论》', '麻黄、桂枝、杏仁、甘草', '发汗解表, 宣
    肺平喘', '外感风寒表实证。恶寒发热, 头身疼痛, 无汗而喘, 舌苔薄白, 脉浮紧'),
54 ('银翘散', 'Yinqiao San', '《温病条辨》', '金银花、连翘、桔梗、薄荷、竹叶、生甘
    草、荆芥穗、淡豆豉、牛蒡子', '辛凉透表, 清热解毒', '温病初起。发热, 微恶风寒, 无汗或
    有汗不畅, 头痛口渴, 咳嗽咽痛, 舌尖红, 苔薄白或薄黄, 脉浮数'),
55 ('四君子汤', 'Sijunzi Tang', '《太平惠民和剂局方》', '人参、白术、茯苓、甘草',
    '益气健脾', '脾胃气虚证。面色萎白, 语声低微, 气短乏力, 食少便溏, 舌淡苔白, 脉虚弱');
56
57 -- 插入方剂组成数据
58 INSERT INTO `PrescriptionHerb` (`prescription_id`, `herb_id`, `dosage
    `, `role`, `sequence`) VALUES
59 (1, 1, '9g', '君', 1),
60 (1, 2, '6g', '臣', 2),
61 (2, 3, '15g', '君', 1),
62 (3, 5, '9g', '君', 1);
63
64 -- 插入药材功效关联数据
65 INSERT INTO `HerbEfficacy` (`herb_id`, `efficacy_id`, `intensity`, `not
    es`) VALUES
66 (1, 2, '强', '发汗解表要药'),
67 (3, 7, '强', '清热解毒良药'),
68 (5, 10, '强', '大补元气要药'),
69 (6, 10, '中', '补气固表');

```

步骤 3: 配置角色访问权限: Access 表

## Bash

```
1 -- 为每个表配置 Access 权限
2 -- 使用 ON DUPLICATE KEY UPDATE 避免重复插入
3 INSERT INTO `Access` (`name`, `alias`, `get`, `head`, `gets`, `heads`,
  `post`, `put`, `delete`) VALUES
4 -- 药材分类表
5 ('herb_category', 'herb_category',
6  ["UNKNOWN", "LOGIN", "CONTACT", "CIRCLE", "OWNER", "ADMIN"]',
7  ["UNKNOWN", "LOGIN", "CONTACT", "CIRCLE", "OWNER", "ADMIN"]',
8  ["LOGIN", "CONTACT", "CIRCLE", "OWNER", "ADMIN"]',
9  ["LOGIN", "CONTACT", "CIRCLE", "OWNER", "ADMIN"]',
10 ["OWNER", "ADMIN"]',
11 ["OWNER", "ADMIN"]',
12 ["OWNER", "ADMIN"]'),
13
14 -- 中药材表
15 ('chinese_herb', 'chinese_herb',
16 ["UNKNOWN", "LOGIN", "CONTACT", "CIRCLE", "OWNER", "ADMIN"]',
17 ["UNKNOWN", "LOGIN", "CONTACT", "CIRCLE", "OWNER", "ADMIN"]',
18 ["LOGIN", "CONTACT", "CIRCLE", "OWNER", "ADMIN"]',
19 ["LOGIN", "CONTACT", "CIRCLE", "OWNER", "ADMIN"]',
20 ["OWNER", "ADMIN"]',
21 ["OWNER", "ADMIN"]',
22 ["OWNER", "ADMIN"]'),
23
24 -- 方剂表
25 ('prescription', 'prescription',
26 ["UNKNOWN", "LOGIN", "CONTACT", "CIRCLE", "OWNER", "ADMIN"]',
27 ["UNKNOWN", "LOGIN", "CONTACT", "CIRCLE", "OWNER", "ADMIN"]',
28 ["LOGIN", "CONTACT", "CIRCLE", "OWNER", "ADMIN"]',
29 ["LOGIN", "CONTACT", "CIRCLE", "OWNER", "ADMIN"]',
30 ["OWNER", "ADMIN"]',
31 ["OWNER", "ADMIN"]',
32 ["OWNER", "ADMIN"]'),
33
34 -- 方剂组成表
35 ('prescription_herb', 'prescription_herb',
36 ["UNKNOWN", "LOGIN", "CONTACT", "CIRCLE", "OWNER", "ADMIN"]',
37 ["UNKNOWN", "LOGIN", "CONTACT", "CIRCLE", "OWNER", "ADMIN"]',
38 ["LOGIN", "CONTACT", "CIRCLE", "OWNER", "ADMIN"]',
39 ["LOGIN", "CONTACT", "CIRCLE", "OWNER", "ADMIN"]',
40 ["OWNER", "ADMIN"]',
41 ["OWNER", "ADMIN"]',
```

```

42  '["OWNER", "ADMIN"]'),
43
44  -- 功效分类表
45  ('efficacy_category', 'efficacy_category',
46  '["UNKNOWN", "LOGIN", "CONTACT", "CIRCLE", "OWNER", "ADMIN"]',
47  '["UNKNOWN", "LOGIN", "CONTACT", "CIRCLE", "OWNER", "ADMIN"]',
48  '["LOGIN", "CONTACT", "CIRCLE", "OWNER", "ADMIN"]',
49  '["LOGIN", "CONTACT", "CIRCLE", "OWNER", "ADMIN"]',
50  '["OWNER", "ADMIN"]',
51  '["OWNER", "ADMIN"]',
52  '["OWNER", "ADMIN"]'),
53
54  -- 药材功效关联表
55  ('herb_efficacy', 'herb_efficacy',
56  '["UNKNOWN", "LOGIN", "CONTACT", "CIRCLE", "OWNER", "ADMIN"]',
57  '["UNKNOWN", "LOGIN", "CONTACT", "CIRCLE", "OWNER", "ADMIN"]',
58  '["LOGIN", "CONTACT", "CIRCLE", "OWNER", "ADMIN"]',
59  '["LOGIN", "CONTACT", "CIRCLE", "OWNER", "ADMIN"]',
60  '["OWNER", "ADMIN"]',
61  '["OWNER", "ADMIN"]',
62  '["OWNER", "ADMIN"]')
63
64  ON DUPLICATE KEY UPDATE
65  alias = VALUES(alias),
66  `get` = VALUES(`get`),
67  `head` = VALUES(`head`),
68  `gets` = VALUES(`gets`),
69  `heads` = VALUES(`heads`),
70  `post` = VALUES(`post`),
71  `put` = VALUES(`put`),
72  `delete` = VALUES(`delete`);

```

### 3、测试新增的接口服务

测试1: 查询清热药材

JSON

```
1 {
2   "ChineseHerb[]": {
3     "ChineseHerb": {
4       "nature": "寒",
5       "@column": "id,name,pinyin,efficacy"
6     },
7     "HerbCategory": {
8       "id@": "ChineseHerb/category_id",
9       "@column": "name:categoryName"
10    }
11  }
12 }
```

测试2: 查询方剂详细信息

JSON

```
1 {
2   "Prescription": {
3     "id": 1
4   },
5   "PrescriptionHerb[]": {
6     "PrescriptionHerb": {
7       "prescription_id@": "Prescription/id",
8       "@column": "dosage,role"
9     },
10    "ChineseHerb": {
11      "id@": "PrescriptionHerb/herb_id",
12      "@column": "name,pinyin,nature"
13    }
14  }
15 }
```

测试3: 复杂统计查询

## JSON

```
1 {
2   "herbByNature": {
3     "ChineseHerb": {
4       "@column": "nature, count(id):count",
5       "@group": "nature"
6     }
7   },
8   "prescriptionCount": {
9     "Prescription": {
10      "@column": "count(id):total"
11    }
12  }
13 }
```

更多使用方法，参考官网：<http://apijson.cn/doc/zh/grammar.html#%E7%99%BB%E5%B D%95>

## 七、实验考核

本实验考核采用【实验随堂查】方式开展。

每个实验完成后，在实验课上通过现场演示的方式向实验指导教师进行汇报，并完成现场问答交流。

每个实验考核满分 100 分，其中实验成果汇报 60 分，现场提问交流 40 分。

实验考核流程：

- (1) 学生演示汇报实验内容的完成情况，实验指导老师现场打分。
- (2) 指导老师结合实验内容进行提问，每位学生提问 2-3 个问题，根据回答的情况现场打分。
- (3) 实验考核结束后，进行公布成绩。