Postman测试Spring Boot后端应用指南

1. 准备工作

首先确保：

- Spring Boot应用已成功启动（默认端口为8080）
- MySQL数据库已配置并运行（应用配置中使用的是 `todoappdb` 数据库）
- Postman已安装

2. 创建Postman集合

1. 打开Postman，创建一个新集合，命名为"TodoList API测试"
2. 不使用变量，直接在每个请求中使用完整URL

3. 用户相关API测试

注册用户

- 方法: `POST`
- URL: `http://localhost:8080/api/users/register`
- Body (raw, JSON):

```
{
  "username": "testuser",
  "password": "password123",
  "email": "test@example.com"
}
```

用户登录

- 方法: `POST`
- URL: `http://localhost:8080/api/users/login`
- Body (raw, JSON):

```
{
  "username": "testuser",
  "password": "password123"
}
```

4. 分类管理API测试

获取用户的所有分类

- 方法: `GET`
- URL: `http://localhost:8080/api/categories/users/testuser`

创建新分类

- 方法: `POST`
- URL: `http://localhost:8080/api/categories`
- Body (raw, JSON):

```
{
  "name": "工作",
  "color": "#FF5733",
  "username": "testuser"
}
```

更新分类

- 方法: `PUT`
- URL: `http://localhost:8080/api/categories/1`
- Body (raw, JSON):

```
{
  "name": "工作任务",
  "color": "#3366FF",
  "username": "testuser"
}
```

删除分类

- 方法: `DELETE`
- URL: `http://localhost:8080/api/categories/1`

批量删除

- 方法：DELETE
- URL： `http://localhost:8080/api/categories/batch`

5. 标签管理API测试

获取用户的所有标签

- 方法：`GET`
- URL: `http://localhost:8080/api/tags/users/testuser`

创建新标签

- 方法: `POST`

- URL: `http://localhost:8080/api/tags`

- Body (raw, JSON):

```
{
  "name": "紧急",
  "color": "#FF0000",
  "username": "testuser"
}
```

更新标签

- 方法: `PUT`

- URL: `http://localhost:8080/api/tags/1`

- Body (raw, JSON):

```
{
  "name": "非常紧急",
  "color": "#CC0000",
  "username": "testuser"
}
```

删除标签

- 方法: `DELETE`

- URL: `http://localhost:8080/api/tags/1`

6. 待办事项API测试

获取用户的所有待办事项

- 方法: `GET`

- URL: `http://localhost:8080/api/todos/users/testuser`

带有过滤条件的待办事项查询

- 方法: `GET`

- URL: `http://localhost:8080/api/todos/users/testuser?done=false&priority=1&categoryId=1&search=会议`

创建新待办事项

- 方法: `POST`

- URL: `http://localhost:8080/api/todos`

- Body (raw, JSON):

```
{
  "description": "完成项目报告",
  "done": false,
  "targetDate": "2025-06-01",
  "username": "testuser",
  "categoryId": 1,
  "priority": 2,
  "reminderTime": "2025-05-31T09:00:00",
  "notes": "需要包含上季度的数据分析",
  "tagIds": [1, 2]
}
```

更新待办事项

- 方法: `PUT`

- URL: `http://localhost:8080/api/todos/10007`

- Body (raw, JSON):

```
{
  "description": "完成项目最终报告",
  "done": true,
  "targetDate": "2025-06-01",
  "username": "testuser",
  "categoryId": 1,
  "priority": 3,
  "reminderTime": "2025-05-31T09:00:00",
  "notes": "已完成所有数据分析",
  "tagIds": [1]
}
```

删除待办事项

- 方法: `DELETE`
- URL: `http://localhost:8080/api/todos/10007`

获取统计信息

- 方法: `GET`
- URL: `http://localhost:8080/api/todos/users/testuser/statistics`

获取即将到期的提醒

- 方法: `GET`
- URL: `http://localhost:8080/api/todos/users/testuser/reminders`