

MyBatis 分页插件 PageHelper

简介

官网: <https://pagehelper.github.io/>

优势:

自动分页: PageHelper 会自动解析你执行的 SQL 语句, 并根据你提供的页码和每页显示的记录数, 自动添加分页相关的 SQL 片段, 从而返回正确的分页结果。无需在 SQL 语句中手动编写复杂的分页逻辑。

配置简单: 在 Spring Boot 项目中通过添加依赖和简单配置就可以启用它。

易于集成: PageHelper 可以与 MyBatis 很好地集成, 无论你是使用 XML 映射文件还是注解方式, 都可以方便地使用 PageHelper。

性能优化: PageHelper 采用了物理分页的方式, 相比于内存分页, 它可以减少内存消耗并提高查询性能。

引入相关依赖

```
1 <dependency>
2     <groupId>com.github.pagehelper</groupId>
3     <artifactId>pagehelper-spring-boot-starter</artifactId>
4     <version>${pagehelper.boot.version}</version>
5 </dependency>
6
7
8 <pagehelper.boot.version>2.0.0</pagehelper.boot.version>
```

编码

- 提出公共的分页参数

```
1 /**
2  * 分页参数
3  */
4 @Getter
5 @Setter
6 public class PageDomain {
```

```
7    /**
8     * 当前记录起始索引
9     */
10   private Integer pageNum = 1;
11
12   /**
13    * 每页显示记录数
14    */
15   private Integer pageSize = 10;
16 }
```

- BaseController中提供getDataTable方法

```
1 public TableDataInfo getDataTable(List<?> list) {
2     if (CollectionUtil.isEmpty(list)) {
3         return TableDataInfo.empty();
4     }
5     return TableDataInfo.success(list, new PageInfo(list).getTotal());
6 }
```

- 修改controller和service

```
1 @GetMapping("/list")
2 public TableDataInfo list(QuestionQueryDTO questionQueryDTO) {
3     return getDataTable(questionService.list(questionQueryDTO));
4 }
5
6 @Override
7 public List<QuestionVO> list(QuestionQueryDTO questionQueryDTO) {
8     PageHelper.startPage(questionQueryDTO.getPageNum(),
9         questionQueryDTO.getPageSize());
10    return questionMapper.selectQuestionList(questionQueryDTO);
11 }
```