

多表查询及分页查询

教学内容

- 第一节 多表查询
- 第二节 分页查询



多表查询

- 实现复杂关系映射，可以使用@Results注解，@Result注解，@One注解，@Many注解组合完成复杂关系的配置。

注解	说明
@Results	代替<resultMap>标签，该注解中可以加入单个或多个@Result注解
@Result	代替<id>标签和<Result>标签，@Result中可以使用以下属性： <ul style="list-style-type: none">- column: 数据表的字段名称
- property: 类中对应的属性名
- one: 与@One注解配合，进行一对一的映射
- many: 与@Many注解配合，进行一对多的映射
@One	代替<association>标签，用于指定查询中返回的单一对象 通过select属性指定用于多表查询的方法 使用格式：@Result(column="",property="",one=@One(select=""))
@Many	代替<collection>标签,用于指定查询中返回的集合对象 使用格式：@Result(column="",property="",many=@Many(select=""))

分页查询

■ 编写配置文件

```
@Configuration
public class MyBatisPlusConfig {
    @Bean
    public MybatisPlusInterceptor paginationInterceptor() {
        MybatisPlusInterceptor interceptor = new MybatisPlusInterceptor();
        PaginationInnerInterceptor paginationInterceptor = new PaginationInnerInterceptor(DbType.MYSQL);
        interceptor.addInnerInterceptor(paginationInterceptor);
        return interceptor;
    }
}
```

分页查询

■ 测试

```
// 查询所有用户: localhost:8080/findAll
@GetMapping("findAll")
public IPage findAll(){
    //设置起始值及每页条数
    Page<User> page = new Page<>(0,2);
    return userMapper.selectPage(page,null);
}
```

