

导出Excel

1.添加依赖

```
<!-- EasyExcel -->
<dependency>
    <groupId>com.alibaba</groupId>
    <artifactId>easyexcel</artifactId>
    <version>2.2.6</version>
</dependency>
```

2.创建一个vo，将成员变量和excel表格做映射

```
package com.ishang.vo;

import com.alibaba.excel.annotation.ExcelProperty;
import com.baomidou.mybatisplus.annotation.FieldFill;
import com.baomidou.mybatisplus.annotation.TableField;
import io.swagger.models.auth.In;
import lombok.AllArgsConstructor;
import lombok.Data;

import java.math.BigDecimal;
import java.time.LocalDateTime;

@Data

public class ProductExcelVo {
    @ExcelProperty("商品编号")
    private Integer productId;
    @ExcelProperty("商品名称")
    private String productName;
    @ExcelProperty("商品价格")
    private BigDecimal productPrice;
    @ExcelProperty("商品库存")
    private Integer productStock;
    @ExcelProperty("商品描述")
    private String productDescription;
    @ExcelProperty("商品图标")
    private String productIcon;
    @ExcelProperty("商品状态")
    private String productStatus;
    @ExcelProperty("商品分类")
    private Integer categoryType;
    @ExcelProperty("分类名称")
    private String categoryName;
}
```

3.创建Handler，设置数据布局

```
package com.ishang.handler;

import com.alibaba.excel.enums.CellDataTypeEnum;
```

```

import com.alibaba.excel.metadata.CellData;
import com.alibaba.excel.metadata.Head;
import com.alibaba.excel.util.CollectionUtils;
import com.alibaba.excel.write.metadata.holder.WriteSheetHolder;
import com.alibaba.excel.write.style.column.AbstractColumnWidthStyleStrategy;
import org.apache.poi.ss.usermodel.Cell;

import java.util.HashMap;
import java.util.List;
import java.util.Map;

public class CustomCellwriteHandler extends AbstractColumnWidthStyleStrategy {

    private Map<Integer, Map<Integer, Integer>> CACHE = new HashMap<>();

    @Override
    protected void setColumnWidth(WriteSheetHolder writeSheetHolder,
    List<CellData> cellDataList, Cell cell, Head head, Integer relativeRowIndex,
    Boolean isHead) {
        boolean needSetWidth = isHead || !CollectionUtils.isEmpty(cellDataList);
        if (needSetWidth) {
            Map<Integer, Integer> maxColumnWidthMap =
    CACHE.get(writeSheetHolder.getSheetNo());
            if (maxColumnWidthMap == null) {
                maxColumnWidthMap = new HashMap<>();
                CACHE.put(writeSheetHolder.getSheetNo(), maxColumnWidthMap);
            }

            Integer columnWidth = this.dataLength(cellDataList, cell, isHead);
            if (columnWidth >= 0) {
                if (columnWidth > 255) {
                    columnWidth = 255;
                }

                Integer maxColumnWidth =
    maxColumnWidthMap.get(cell.getColumnIndex());
                if (maxColumnWidth == null || columnWidth > maxColumnWidth) {
                    maxColumnWidthMap.put(cell.getColumnIndex(), columnWidth);

                    writeSheetHolder.getSheet().setColumnWidth(cell.getColumnIndex(), columnWidth *
    256);
                }
            }
        }
    }

    private Integer dataLength(List<CellData> cellDataList, Cell cell, Boolean
    isHead) {
        if (isHead) {
            return cell.getStringCellValue().getBytes().length;
        } else {
            CellData cellData = cellDataList.get(0);
            CellDataTypeEnum type = cellData.getType();
            if (type == null) {
                return -1;
            } else {
                switch (type) {

```

```

        case STRING:
            return cellData.getStringValue().getBytes().length;
        case BOOLEAN:
            return
cellData.getBooleanValue().toString().getBytes().length;
        case NUMBER:
            return
cellData.getNumberValue().toString().getBytes().length;
        default:
            return -1;
    }
}
}
}
}
}

```

4.controller

```

@ApiOperation("导出Excel")
@GetMapping("/export")
public void export(HttpServletResponse response){
    try {
        // 设置数据接收类型
        response.setContentType("application/vnd.ms-excel");
        response.setCharacterEncoding("UTF-8");
        String fileName = URLEncoder.encode("商品信息", "UTF-8");
        response.setHeader("Content-
disposition", "attachment;filename="+fileName+".xlsx");
        // 获取ProductExcelVO类型的list
        List<ProductExcelVO> list = this.productInfoService.excelVOList();
        EasyExcel.write(response.getOutputStream(), ProductExcelVO.class)
            .registerWriteHandler(new CustomCellWriteHandler())
            .sheet("商品信息")
            .dowrite(list);
    } catch (IOException e) {
        e.printStackTrace();
    }
}

```

service

```

public List<ProductExcelVO> excelVOList();

```

serviceImpl

```

@Override
public List<ProductExcelVO> excelVOList() {
    List<ProductInfo> productInfoList = this.productInfoMapper.selectList(null);
    List<ProductExcelVO> productExcelVOList = new ArrayList<>();
    for (ProductInfo productInfo : productInfoList) {
        ProductExcelVO productExcelVO = new ProductExcelVO();
        BeanUtils.copyProperties(productInfo, productExcelVO);
        if (productInfo.getProductStatus() == 1) {
            productExcelVO.setProductStatus("上架");
        } else {
            productExcelVO.setProductStatus("下架");
        }
    }
}

```

```
productExcelVO.setCategoryName(this.productCategoryMapper.findCategoryNameByType(productInfo.getCategoryType()));  
    productExcelVOList.add(productExcelVO);  
}  
return productExcelVOList;  
}
```