

CLAYS POD Order Processing Tool

A web-based application for processing print-on-demand orders for Clays Ltd, designed to streamline the conversion of Excel order files into CSV format suitable for POD fulfillment.

<https://tjintranet.github.io/CLAYS-POD-Order-Processing-Tool/>

Overview

This tool allows users to upload Excel files containing book orders, validate them against available inventory, and generate properly formatted CSV files for Clays POD processing. The application provides real-time validation, order management, and export capabilities.

Features

- **Excel File Processing:** Upload and process `.xlsx` and `.xls` order files
- **Real-time Validation:** Check ISBNs against available inventory database
- **ISBN Search:** Quick lookup tool to check individual title availability
- **Order Management:** View, edit, and delete individual order lines
- **Batch Operations:** Select and delete multiple rows at once
- **CSV Export:** Generate properly formatted CSV files for Clays POD
- **Template Download:** Get a pre-formatted Excel template for orders
- **Copy to Clipboard:** Copy order data as formatted HTML table
- **Filtering Options:** Toggle to show only unavailable items for quick issue identification
- **Security Enhanced:** Input validation, XSS protection, and file security checks

File Structure

```
project-root/
├── index.html           # Main application interface
├── script.js           # Core application logic
├── data.json           # Book inventory database (required)
├── order_template.xlsx # Excel template file (required)
├── favicon-32x32.png  # Favicon (optional)
└── apple-touch-icon.png # Apple touch icon (optional)
```

Prerequisites

Required Files

1. **data.json:** A JSON file containing the book inventory database with the following structure:

```
[
  {
    "code": "9781234567890",
    "description": "Book Title",
```

```
    "setupdate": "2024-01-01"  
  }  
]
```

2. **order_template.xlsx**: An Excel template file with columns for ISBN and Qty

Browser Requirements

- Modern web browser with JavaScript enabled
- Support for File API and Clipboard API
- Internet connection (for loading external libraries)

Installation

1. **Download/Clone** all project files to a local directory
2. **Ensure** `data.json` contains your current book inventory
3. **Place** `order_template.xlsx` in the same directory
4. **Open** `index.html` in a web browser

Usage Instructions

Getting Started

1. Open the Application

- Double-click `index.html` or open it in your web browser
- The tool will automatically load the book inventory from `data.json`

2. Download Template (First Time)

- Click "Download Template" to get the Excel order format
- Use this template for creating your orders

Quick ISBN Lookup

1. Check Individual Titles

- Use the "ISBN Lookup" panel on the right side
- Enter any ISBN (10 or 13 digits, with or without dashes)
- Click "Search" or press Enter
- View availability status with book details

2. Search Results

- **Green (Available)**: Shows ISBN, title, and setup date
- **Red (Not Available)**: Shows ISBN and unavailable message
- Results auto-hide after 10 seconds for available items

Processing Orders

Processing Orders

1. Enter Order Reference

- Input a unique order reference in the "Order Reference" field (left panel)
- This field is required before uploading files

2. Upload Excel File

- Click "Choose File" and select your Excel order file
- Supported formats: **.xlsx, .xls**
- The file should contain columns: **ISBN** and **Qty**
- Maximum file size: 10MB

3. Review Processed Data

- Orders appear in the preview table with validation status
- Green "Available" badges indicate books found in inventory
- Red "Not Found" badges indicate missing ISBNs
- Use the "Show only unavailable items" toggle to filter and focus on problem orders

Managing Orders

Individual Row Actions:

- **Delete Single Row:** Click the trash icon in the Action column
- **View Details:** All order information is displayed in the table

Batch Operations:

- **Select Rows:** Use checkboxes to select specific rows
- **Select All:** Use the header checkbox to select/deselect all rows
- **Delete Selected:** Click "Delete Selected Rows" to remove checked items

View and Filter Options:

- **Toggle Filter:** Use the "Show only unavailable items" switch to display only books not found in inventory
- **Quick Problem Identification:** When filtered, easily see which ISBNs need attention
- **Status Messages:** Get feedback on how many items are being displayed
- **All Items View:** Toggle off to return to showing all processed orders

ISBN Search:

- **Individual Lookup:** Use the right panel to check single ISBNs instantly
- **Format Flexibility:** Accepts ISBNs with or without dashes/spaces
- **Detailed Results:** Shows title, availability, and setup date information
- **Keyboard Shortcut:** Press Enter in search field to trigger lookup

Data Export:

- **Download CSV:** Generate Clays POD-formatted CSV file
- **Copy to Clipboard:** Copy table as HTML for pasting into other applications

View Options:

- **Filter Toggle:** Use "Show only unavailable items" switch to display only books not found in inventory
- **Full View:** Toggle off to see all processed orders

Excel File Format

Your Excel file should contain the following columns:

Column	Description	Example
ISBN	13-digit book identifier	9781234567890
Qty	Quantity to order	5

Notes:

- ISBNs can include dashes or spaces (automatically cleaned)
- Scientific notation in Excel is automatically converted
- Quantities must be whole numbers

CSV Output Format

The generated CSV follows Clays POD specifications:

```
HDR,ORDER123,20240605,POD,Clays Ltd,Popson Street,,Bungay,Suffolk,UK,NR35
1ED,GB,01986 893 211,
DTL,ORDER123,001,9781234567890,5,,,,,,,,,
DTL,ORDER123,002,9780987654321,3,,,,,,,,,
```

Structure:

- **HDR Line:** Header with order details and Clays Ltd address
- **DTL Lines:** Detail lines with ISBN and quantity for each book

Troubleshooting

Common Issues

"Please enter an order reference" Warning

- Ensure the Order Reference field is filled before uploading files

"Error loading book data"

- Check that `data.json` exists and is properly formatted
- Verify JSON syntax is valid

"Error processing file"

- Ensure Excel file has ISBN and Qty columns

- Check that ISBNs are in a recognizable format
- Verify quantities are numeric values
- Check file size is under 10MB limit
- Ensure file type is .xlsx or .xls

Books showing "Not Found" status

- Check that ISBNs match exactly with codes in `data.json`
- Verify the inventory database is up to date
- Use the "Show only unavailable items" toggle to quickly identify problem ISBNs
- Use the ISBN Search feature to double-check individual titles

ISBN Search not working

- Ensure ISBN format is valid (10 or 13 digits)
- Check that `data.json` has loaded successfully
- Try removing dashes or spaces from the ISBN
- Verify the ISBN exists in your inventory database

Filter not working properly

- Refresh the page and try again
- Ensure JavaScript is enabled in your browser
- Check browser console for any error messages

File Upload Issues

- **File won't upload:** Check file extension is `.xlsx` or `.xls`
- **Data not processing:** Ensure column headers match expected format
- **ISBN recognition:** Remove any special characters or formatting

Browser Compatibility

- **Clipboard features:** May not work on older browsers
- **File processing:** Requires modern browser with JavaScript support
- **CSV download:** Should work on all modern browsers

Technical Details

Dependencies

External libraries loaded via CDN:

- Bootstrap 5.3.2 (UI framework)
- Font Awesome 6.4.0 (icons)
- SheetJS 0.18.5 (Excel processing)
- PapaParse 5.4.1 (CSV generation)

Data Processing

1. **Excel Parsing:** Uses SheetJS to read Excel files (max 10MB)

2. **ISBN Normalization:** Removes non-digits and pads to 13 digits
3. **Inventory Matching:** Compares against loaded JSON database
4. **CSV Generation:** Creates Clays POD-compliant format
5. **Input Validation:** Validates file types, ISBNs, and quantities
6. **XSS Protection:** Sanitizes all user inputs and prevents script injection

Security Features

- **Content Security Policy:** Prevents XSS attacks and unauthorized resource loading
- **File Validation:** Checks file types, sizes, and content before processing
- **Input Sanitization:** Cleans all user inputs to prevent injection attacks
- **Secure Headers:** Implements multiple security headers for enhanced protection
- **Error Handling:** Provides safe error messages without information disclosure

Security Considerations

- All processing happens client-side (no data sent to servers)
- Files are processed in browser memory only
- No persistent storage of order data

Customization

Modifying Customer Details

Edit the `customerConfig` object in `script.js`:

```
const customerConfig = {
  "name": "Your Company Name",
  "address": {
    "street": "Your Street",
    "city": "Your City",
    // ... other address fields
  },
  "phone": "Your Phone",
  "type": "Your Type"
};
```

Updating Inventory

Replace or update `data.json` with current book inventory:

```
[
  {
    "code": "9781234567890",
    "description": "New Book Title",
    "setupdate": "2024-06-05"
  }
]
```

Support

For technical issues:

1. Check browser console for error messages
2. Verify all required files are present
3. Ensure `data.json` format matches specification
4. Test with the provided template file

Version History

- **v2.0:** Enhanced security and ISBN search features
 - Added individual ISBN lookup functionality
 - Implemented comprehensive security enhancements (CSP, input validation, XSS protection)
 - Added file size limits and enhanced error handling
 - Improved user interface with responsive design
- **v1.0:** Initial release with core functionality
 - Supports Excel upload, validation, and CSV export
 - Includes batch operations and clipboard integration