

HoweScape CSV Boy Scout Label Print

This application takes the CSV file created from ScoutBook and creates a label PDF file. The Interface to the application is a dialog box. See Below.

HoweScape.com CSV Boy Scout Label Printer

File

Scout Book File: ScoutBook Invoice CSV File

Label Style: Avery 6570

Font Size: 9 Label Position: 1

Pack 0000 Council Na... Abraham Lincoln Council

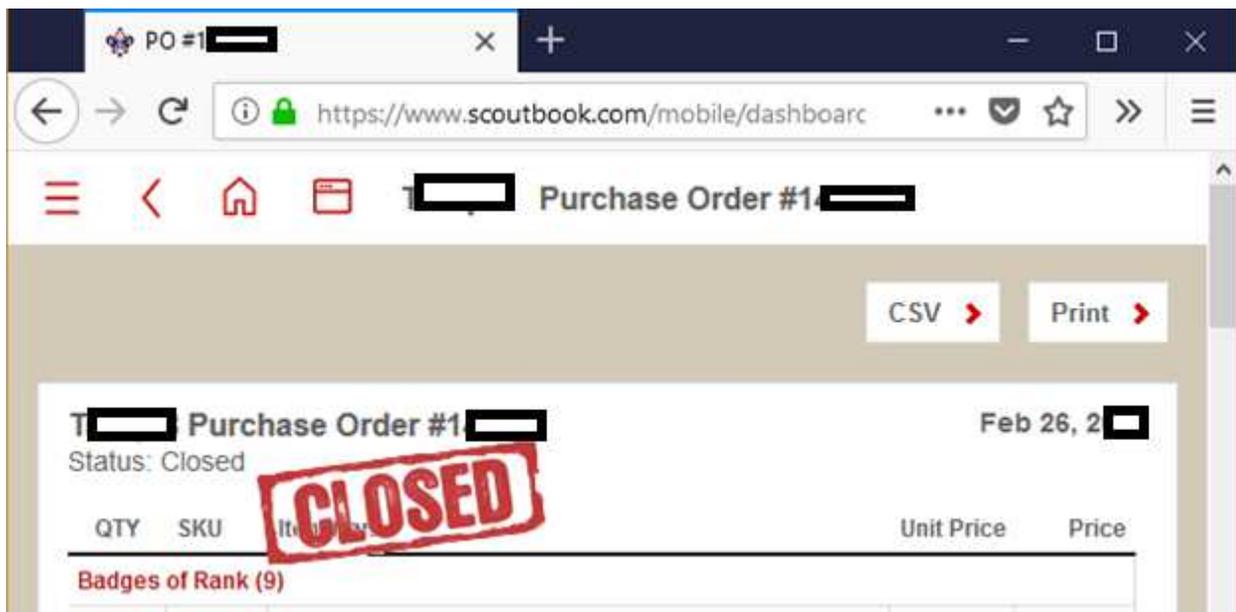
Rank Msg Has met the Req'ts for the<n> <rank>

MB Msg Has met the Req'ts for the<n> <mb_name>

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16
17	18	19	20
21	22	23	24
25	26	27	28
29	30	31	32

Store Settings Generate Labels

This tool is designed to create a PDF file which can be used to print labels. This is similar to the output from TroopMaster. The input file is a CSV file created by ScoutBook.



The above screen capture from ScoutBook shows the CSV button. This button exports the information on the invoice. This is an input file used by the HoweScape CSV Boy Scout Label Printer.

There are a few pieces of information which are missing from the CSV file. These fields are provided by the application.

Items on the dialog:

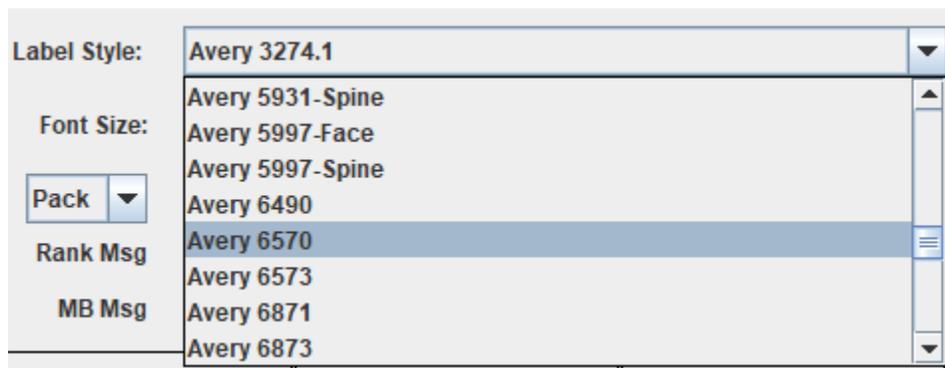
- Scout Book CSV file and path
- Label Style (Avery 6570)
- Font Size
- Label Position on label sheet
- Unit type
- Unit Number
- Council name
- Rank Message
- Merit Badge Message
- Label sheet preview with position numbers
- Store Settings button
- Generate Labels button

The application has a file pulldown menu. This menu has 3 items, File, About and Exit. The File opens a file open dialog and allows navigation to the location of the CSV file. The second is the About item.

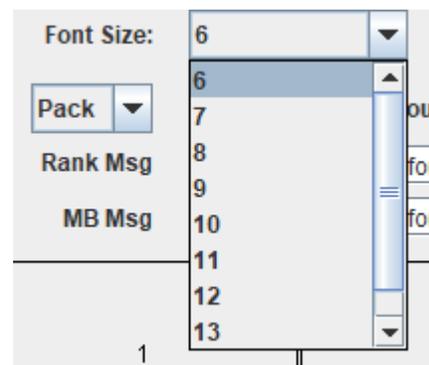


This opens a dialog box with the application version number and the website where you can get the application. The third Exit, leaves the application.

The dialog contains several dropdown lists. These lists are loaded from XML files stored in the jar or are a part of the application. The label style information is being loaded from an XML file stored in the Jar. The only style which was tested was Avery 6570.



The font size list is a fixed list.

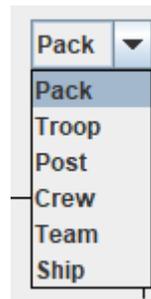


The Label position is a dynamic item. It is updated based on the label style selected. This item is used to select the position on the label sheet where the first label from the CSV file should be positioned. To assist you in selecting this value a layout of the labels is presented at the bottom section of the dialog

box. If you have a sheet of labels which has been used previously you can insert blank labels to get the new items to the proper position.

The unit type is the next choice.

The unit number is a text field. You can type in any value



you like.

The council list is a selection from an XML file which is stored in the jar. This list was compiled from a web site. Any updates will require changing CouncilList.xml file in the application.

The next field is a message to be included on the label. The tag “<rank>” is replaced with the rank of the label. Next field is “MB Msg” which is the message which will be used for a merit badge label. The tag “<mb_name>” is replaced with the merit badge name which is in the CSV input file. The other tag you will see is “<n>”. This will be replaced with a carriage return. This helps with alignment on the labels.

The next item is a layout of the labels on a page. This is used to know what will be considered the first label position.

There are 2 buttons at the bottom. Save settings will create a Config_ScoutBook.properties file. This file saves all the setting you have made in the dialog. This file is created in the same directory as the application JAR. When the dialog is opened a second time it will have the setting you used restored. The information in the properties file is stored as label and value pairs. One of the issues I encountered was a merit badge name which was too long. I have added the ability to abbreviate a string from the CSV file. These values are in the properties file with the label of “Abbr-List” and values separated with a hash character. The Generate Labels button creates the PDF file. It uses the location and file name of the CSV file. The output is a PDF file. There is also a second file created. The file is a list of the person getting the badge and the badges which they earned in this CSV file. The 2 files created will be in the same directory as the CSV file. One will be the same name as the CSV with an extension of PDF. The second will have the CSV name followed by “_list” with the extension of PDF.

There is still room for improvement in this application. The development is continuing. The source files will be available soon. The first page has the best alignment with the label sheet. Comment, suggestions as well as donations are welcome.