ABSTRACT
In a compact columnar output, the maximum number of observations should be made to attractively fit either on the width of a single page or on the width of a minimum number of pages without losing any information. This is the neatest most heuristic way to present a data set. Various techniques using PROC REPORT will be shown to accomplish this. Everything was done in SAS® 8.2 batch using the NO WINDOWS (NOWD) option.

INTRODUCTION
You may wish to display a data set to:
• use yourself,
• show others at a meeting
• put in a standardized regulatory report.
• put in a publication

These are ranked in roughly increasing effort required to create the report. The first two reasons and proofreading can be adequately handled using a browser. If a hard copy is required, one would have to use either (1) PROC PRINT; (2) PROC REPORT with or without the Output Delivery System (ODS) STYLE option⁴; (3) ODS as a stand alone; or (4) PUT statement formatting through the use of a DATA _NULL_. PROC PRINT cannot be sufficiently customized. DATA _NULL_ can do anything but it is unduly complex to program. ODS, recommended for publications, allows one to use proportional fonts and color. But, for a standardized report, PROC REPORT, with a mastery of all applicable options, can rival a serious DATA _NULL_ step. The main thing that PROC REPORT cannot do is to wrap observations neatly within a page as the WRAP and NAMED options create a messy output. Compacting columns reduces the need for this desirable feature.

OBJECTIVE
The objective is to display wide data sets, given a specified line size, in a neat and heuristic manner. It is not to save paper. PROC REPORT options¹ not needed to accomplish this limited objective are not discussed. A recent clinical reason for making the report compact, is the FDA guideline of using 12 point fonts with 1" margins for tables in electronic submissions. This restricts you to 90 columns and 43 lines. For a narrow data set, you can use the BOX option to draw lines between columns or around cells. You can also use the number of PANELS and PSPACE (space between panels) options to minimize paper use by putting the data in newspaper like columns (see SPECIAL AND UNPRINTABLE CHARACTERS). Do not use these options for a wide data set. It is assumed, you know how to write a basic PROC REPORT with its options and the COLUMN (COL), DEFINE and BREAK statements.

PROC PRINT
With a little trial and error, you can calculate the page size (PS) system option needed to create a columnar output by having it wrap at the number of rows in the data set and then edit this output by adding page breaks. This is not viable for a data set having variables with more observations than will fit on a single page, but it will create neat even columns. But, (1) there are no beautification options other than the ID and VAR statements; (2) wide columns are truncated at 128 for a Wyeth production standard line size of 132; and (3) one cannot control the space between columns.

DEFAULT PROC REPORT
This will create a columnar report and a limited number of variables will fit neatly on a page width. However, (1) rows are not identified for multiple pages; (2) spacing between columns is uneven; (3) column order is not optimum; and (4) columns with a width greater than 113 with a line size of 132 will result in an error message and no output will be generated. PROC REPORT was designed to run interactively and thus has some odd defaults.
• Labels are used as column headers and label words are split.
• Variables are output in position order.
• Space between columns is 2 including before the first column.

• There is no option to print observation numbers³.
• MISSING option should always be used to print all data rows.
• If all variables are numeric and none are specified as DISPLAY, they are summed instead of listed.
• If a variable name, not in the data set, is listed in both the COL and DEFINE statements, there is no error as it is assumed that it was created in a COMPUTE statement.

Thus one cannot create a useful compact columnar output without using a fair number of PROC REPORT options as well as having a good understanding of these options. Note in PC SAS, you can make a report compact using the Windows Page Setup options².

JUSTIFICATION RULES WITHIN COLUMNS
• Default is right for numeric and left for character.
• Numerical values are right justified within their specified formats and these are justified within their specified widths.
• Character values are justified in width with leading blanks retained and trailing blanks eliminated.

SEPARATELY JUSTIFYING LABELS AND VALUES
Justification rules apply simultaneously to values and their labels. Thus, it requires some effort to separately justify the two. You usually wish to left or right justify the value and center the label. You can do this by putting a SPLIT character followed by the requisite number of blanks at the beginning and/or a macro variable containing an appropriate unprintable character(s) at the end of the label (e.g., %sysfunc(byte(160)) works). Also, you can put label characters over the space before the column by using RIGHT and SPACING=0 in the DEFINE statement for the variable.

SPLIT CHARACTER
This splits both labels and variables having the FLOW option. You should pick a keyboard printable character less common than the default “/” since “/” is quite common in many entered texts. Possible choices are Wyeth’s standard of |, ~, \ or `. If you are paranoid about using any keyboard character as the possibility always exists that it may occur in your data, you may use an obscure non-keyboard printable character as shown in the next section. You may wish to indent concatenated flowed text by inserting a split character plus the desired number of spaces. The indentation won’t be correct unless one also puts a split character at the end of the text. Normally, if a word’s length in a flowed variable is greater than the variable’s width, the word will split at that width. However, if there is a split character in the text, words at the end of the field will split randomly due to a SAS bug that will be fixed in a future SAS version. To fix this, you either widen the field to eliminate non-indented flow or use the macro in this paper that inserts SPLIT characters based on knowledge of the field’s width.

SPECIAL AND UNPRINTABLE CHARACTERS
Hexadecimal A’s and C’s are line and page feeds and must be removed from flowed text to prevent unwanted splitting. You should also remove any other unprintable printer control characters that may exist in flowed variables. This is an uncommon problem that can happen with data coming from many sources (investigator comments). To find what characters should be removed for your host environment, run the following code in display manager:

```sas
data a;
  length f $1;
  do byt=0 to 255;
    if-byte(byt);
    output;
  end;
run;
proc report spacing=1 pspace=1 panels=16 ls=95 ps=20 nowd;
  col byt f;
  define byt/width=3 'Byt/___' spacing=0;
  define f/width=1 'C_';
run;
```
This will give you the SAS mapping for the 256 characters in a byte as can be seen in the following screen print of the output window:

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>16</td>
<td>32</td>
<td>48</td>
<td>64</td>
<td>80</td>
<td>96</td>
<td>112</td>
<td>128</td>
<td>144</td>
<td>160</td>
<td>176</td>
<td>192</td>
<td>208</td>
<td>224</td>
<td>240</td>
</tr>
<tr>
<td>1</td>
<td>17</td>
<td>33</td>
<td>49</td>
<td>65</td>
<td>81</td>
<td>97</td>
<td>113</td>
<td>129</td>
<td>145</td>
<td>161</td>
<td>177</td>
<td>193</td>
<td>209</td>
<td>225</td>
<td>241</td>
</tr>
<tr>
<td>2</td>
<td>18</td>
<td>34</td>
<td>50</td>
<td>66</td>
<td>82</td>
<td>98</td>
<td>114</td>
<td>130</td>
<td>146</td>
<td>162</td>
<td>178</td>
<td>194</td>
<td>210</td>
<td>226</td>
<td>242</td>
</tr>
<tr>
<td>3</td>
<td>19</td>
<td>35</td>
<td>51</td>
<td>67</td>
<td>83</td>
<td>99</td>
<td>115</td>
<td>131</td>
<td>147</td>
<td>163</td>
<td>179</td>
<td>195</td>
<td>211</td>
<td>227</td>
<td>243</td>
</tr>
<tr>
<td>4</td>
<td>20</td>
<td>36</td>
<td>52</td>
<td>68</td>
<td>84</td>
<td>100</td>
<td>116</td>
<td>132</td>
<td>148</td>
<td>164</td>
<td>180</td>
<td>196</td>
<td>212</td>
<td>228</td>
<td>244</td>
</tr>
<tr>
<td>5</td>
<td>21</td>
<td>37</td>
<td>53</td>
<td>69</td>
<td>85</td>
<td>101</td>
<td>117</td>
<td>133</td>
<td>149</td>
<td>165</td>
<td>181</td>
<td>197</td>
<td>213</td>
<td>230</td>
<td>246</td>
</tr>
<tr>
<td>6</td>
<td>22</td>
<td>38</td>
<td>54</td>
<td>70</td>
<td>86</td>
<td>102</td>
<td>118</td>
<td>134</td>
<td>150</td>
<td>166</td>
<td>182</td>
<td>198</td>
<td>214</td>
<td>231</td>
<td>247</td>
</tr>
<tr>
<td>7</td>
<td>23</td>
<td>39</td>
<td>55</td>
<td>71</td>
<td>87</td>
<td>103</td>
<td>119</td>
<td>135</td>
<td>151</td>
<td>167</td>
<td>183</td>
<td>200</td>
<td>215</td>
<td>232</td>
<td>248</td>
</tr>
<tr>
<td>8</td>
<td>24</td>
<td>40</td>
<td>56</td>
<td>72</td>
<td>88</td>
<td>104</td>
<td>120</td>
<td>136</td>
<td>152</td>
<td>168</td>
<td>184</td>
<td>201</td>
<td>216</td>
<td>233</td>
<td>250</td>
</tr>
<tr>
<td>9</td>
<td>25</td>
<td>41</td>
<td>57</td>
<td>73</td>
<td>89</td>
<td>105</td>
<td>121</td>
<td>137</td>
<td>153</td>
<td>169</td>
<td>185</td>
<td>202</td>
<td>217</td>
<td>234</td>
<td>251</td>
</tr>
<tr>
<td>10</td>
<td>26</td>
<td>42</td>
<td>58</td>
<td>74</td>
<td>90</td>
<td>106</td>
<td>122</td>
<td>138</td>
<td>154</td>
<td>170</td>
<td>186</td>
<td>203</td>
<td>218</td>
<td>235</td>
<td>252</td>
</tr>
<tr>
<td>11</td>
<td>27</td>
<td>43</td>
<td>59</td>
<td>75</td>
<td>91</td>
<td>107</td>
<td>123</td>
<td>139</td>
<td>155</td>
<td>171</td>
<td>187</td>
<td>204</td>
<td>220</td>
<td>236</td>
<td>253</td>
</tr>
<tr>
<td>12</td>
<td>28</td>
<td>44</td>
<td>60</td>
<td>76</td>
<td>92</td>
<td>108</td>
<td>124</td>
<td>140</td>
<td>156</td>
<td>172</td>
<td>188</td>
<td>205</td>
<td>221</td>
<td>237</td>
<td>254</td>
</tr>
<tr>
<td>13</td>
<td>29</td>
<td>45</td>
<td>61</td>
<td>77</td>
<td>93</td>
<td>109</td>
<td>125</td>
<td>141</td>
<td>157</td>
<td>173</td>
<td>189</td>
<td>206</td>
<td>222</td>
<td>238</td>
<td>255</td>
</tr>
<tr>
<td>14</td>
<td>30</td>
<td>46</td>
<td>62</td>
<td>78</td>
<td>94</td>
<td>110</td>
<td>126</td>
<td>142</td>
<td>158</td>
<td>174</td>
<td>190</td>
<td>207</td>
<td>223</td>
<td>239</td>
<td>256</td>
</tr>
<tr>
<td>15</td>
<td>31</td>
<td>47</td>
<td>63</td>
<td>79</td>
<td>95</td>
<td>111</td>
<td>127</td>
<td>143</td>
<td>159</td>
<td>175</td>
<td>191</td>
<td>208</td>
<td>224</td>
<td>240</td>
<td>257</td>
</tr>
</tbody>
</table>

Note than the LS, PS (which override the system options) and PANELS options were used to create a balanced square output. The 16 by 16 format is useful as FORMCHAR is expressed in hexadecimal. All of the non-blank characters are treated as printable by SAS, including the non-keyboard characters with byte>126. Save and print this output window. The keyboard characters will be the same for the screen and paper printouts.

The characters to be excluded will cause page and line breaks, tabs and font changes. You may see many odd characters, replacing the above blanks. Note that while these blank replacing characters will print as the 12 FORMCHAR characters that PROC REPORT will cause that variable as well as the preceding variables in the COLUMN required to fully identify all observations. This gives you to room to put lines or multiple spaces between columns.

Use PROC FREQ (or my %legend) to determine whether long variables can be coded and explain the code in a legend.

Do not use the FLOW option unless necessary as it increases the number of lines per observation.

Judiciously use the SPLIT character in a flowed variable to eliminate the random word splitting SAS error.

Eliminate all but one of a group of variables that have a one to one relationship with each other.

Since formats can alter variable widths, apply them prior to calculating column widths.

Sensibly condense character variables.

Edit variables without altering their meaning.

Transfer meaning from a variable to its label.

Use width to truncate a SAS format (e.g., width=13 and f=datetime18. will remove seconds from a datetime variable)

Use the STYLE attribute, some of the six font parameters and ODS to eliminate the unattractiveness of monospace fonts. Decimal points may not line up and this is not the best approach for reports that must be standardized over all drugs.

ALPHANUMERIC VARIABLES

• Determine their maximum width in the data set.

• If a format increases this width, use that width.

• Consider removing any invariant prefixes or suffixes (e.g., leading zeroes in a patient number).

• If FLOW is required, consider the line size constraint, calculate the width plus spacing of all other variables and:
   (1) For one FLOW variable, use its maximum calculated width.
   (2) For multiple FLOW variables, determine how to best allocate their widths to minimize lines per observation.

• See if other data can be put on the added line(s) per observation (e.g., concatenate two variables with the SPLIT character between them and use the FLOW option to make them print under one another. If necessary, indent them).

• Do not make the field’s width less than the width of the largest word in the text.

NUMERIC VARIABLES

• Determine their range, minimum and maximum values and whether they are integers, and then specify an appropriate format and decimal point. Avoid the use of the BEST format.

• Use the DEFINE statement NOZERO option to suppress printing of a column whose values are zero or missing.

• For date time variables, specify an appropriate compact format (e.g., MMDDYY6.). Separate date and time with the DATEPART and TIMEPART functions. If the time is missing for all observations, remove it from the report.

• Use ORDER=INTERNAL in the DEFINE statement to prevent dates from being sorted alphabetically as default is formatted.

• If integer with trailing zeroes (common in lab tests) change leading zeroes in a patient number).

• Do not alter any variable if proofreading.

• Remove space before first column (SPACING=0 in DEFINE).

• Remove space before first column (SPACING=0 in DEFINE).

• Transfer meaning from a variable to its label.

• Use width to truncate a SAS format (e.g., width=13 and f=datetime18. will remove seconds from a datetime variable)

• Use the STYLE attribute, some of the six font parameters and ODS to eliminate the unattractiveness of monospace fonts. Decimal points may not line up and this is not the best approach for reports that must be standardized over all drugs.

TO MAKE THE REPORT COMPACT

• Make every reasonable effort to limit the page width to one. (Many users find a width of more than one page undesirable.)

• Reduce the SPACING between columns to one.

• Remove space before first column (SPACING=0 in DEFINE).

• Do not alter any variable if proofreading.

• Drop unnecessary variables from COL statement (e.g. Meta variables such as the source case report form number).

• Drop variables having the same or missing values for all rows and consider putting them in a title, footnote or legend.

• For all columns, use the minimum possible width that will not truncate any data. Do not use the variable’s default width.

• Sort the data by sensible variables having a fair number of rows for each combination in the BY statement. Use this BY in PROC REPORT.

• For data sets wider than a single page, pick the minimum number of ID variables to adequately identify all observations. Balance the width of the non-ID variables across pages so that each page has about the same width. This gives you to room to put lines or multiple spaces between columns.

• Use PROC FREQ (or my %legend) to determine whether long variables can be coded and explain the code in a legend.

• Do not use the FLOW option unless necessary as it increases the number of lines per observation.

• Judges use the SPLIT character in a flowed variable to eliminate the random word splitting SAS error.

• Eliminate all but one of a group of variables that have a one to one relationship with each other.

• Since formats can alter variable widths, apply them prior to calculating column widths.

• Sensibly condense character variables.

• Edit variables without altering their meaning.

• Transfer meaning from a variable to its label.

• Use width to truncate a SAS format (e.g., width=13 and f=datetime18. will remove seconds from a datetime variable)

• Use the STYLE attribute, some of the six font parameters and ODS to eliminate the unattractiveness of monospace fonts. Decimal points may not line up and this is not the best approach for reports that must be standardized over all drugs.
TO MAKE A COMPACT REPORT MORE ATTRACTIVE

- Use ODS and a proportional font.
- Appropriately order the COL statement variables.
- Use neat and informative titles, footnotes and/or legends.
- Appropriately specify the ORDER option in the DEFINE statements for the initial variables in the COL statement.
- Consider using NOPRINT in a DEFINE statement to order the observations by a variable which is not printed.
- Use the BREAK BEFORE statement to put blank or solid lines between ID observations and separate the labels from the values. With the FORMCHAR option you can make a line of any character (e.g., '-') available in your host system.
- A LINE statement in a COMPUTE BLOCK will allow you to customize spacing lines.
- Use informative labels neatly spanned in the COL statement and appropriately split in the DEFINE statement. They should be designed with the column width in mind.
- Separately justify values and their labels as discussed.
- End all labels with a SPLIT character followed by two underline characters (i.e., __) to separately underline labels.
- If necessary, expand a label’s meaning in a legend page.
- Use neat and informative titles, footnotes and/or legends.

ADVERSE EVENT LISTING EXAMPLE

Wyeth has highly sophisticated validated modules, with built in error checks, that create data sets for listing and summarization. The project programmer for the specified drug chooses the options error checks, that create data sets for listing and summarization.

The project programmer for the specified drug chooses the options error checks, that create data sets for listing and summarization.

• If necessary, expand a label’s meaning in a legend page.

ADVERSE EVENT LISTING EXAMPLE

Wyeth has highly sophisticated validated modules, with built in error checks, that create data sets for listing and summarization. The project programmer for the specified drug chooses the options error checks, that create data sets for listing and summarization.

• If necessary, expand a label’s meaning in a legend page.

(1) Labels are not neat. The space allocated for most columns was determined by their label length rather than by their data.

(2) The SAS error of random word splitting caused by the SPLIT character in the text occurs in MUSCULOSKELETAL SYST.

(3) The lines per observation are excessive and the indented variable flows to the beginning of the line. The times are blank

(4) Using DATE7 and no width would remove the date’s century.

(5) SEV with only four different values, while sparse, has more than four characters. It should either be flowed or coded.

(6) ACTN was truncated in a data step to 3 characters, which limits you to one code as it had a space after the comma.

(7) OUTC is truncated to 3 characters but is not in the footnotes.

(8) An incorrect template caused DAI to be inadvertently left out of the call in the module. No error resulted as it was in the COL and DEFINE statements and thus was blank.

(9) The template also caused “VERBATIM” to be wrongly indented since its label had no split character at its end.

(10) There are only 15 different COSTART body system codes, 12 of which are used in this report. It thus makes sense to code them by their unique first two characters and put the code in a footnote rather than listing them for each observation.

INDENTATION MACRO

With reasonable values of the parameters, the following macro will (1) eliminate the SAS error; (2) give the correct indentation when an indented line flows; and (3) handle the largest word’s width being larger than the field width as well as blank indented lines:

```sas
%mend ind;
run;
```

At first glance, the first page of this output (with the titles, which are the same as Figure 2, stripped) in Figure 1 seems acceptable. The options SPACING=1 and SPACING=0 for the first column should yield a compact report and the above adds up exactly to a usage of 132 columns. There is a BY statement for investigator and treatment name and a DEFINE statement for every variable in the column statement. However, you should always carefully scrutinize an entire PROC REPORT output for problems.

(1) Labels are not neat. The space allocated for most columns was determined by their label length rather than by their data.

(2) The SAS error of random word splitting caused by the SPLIT character in the text occurs in MUSCULOSKELETAL SYST.

(3) The lines per observation are excessive and the indented variable flows to the beginning of the line. The times are blank

(4) Using DATE7 and no width would remove the date’s century.

(5) SEV with only four different values, while sparse, has more than four characters. It should either be flowed or coded.

(6) ACTN was truncated in a data step to 3 characters, which limits you to one code as it had a space after the comma.

(7) OUTC is truncated to 3 characters but is not in the footnotes.

(8) An incorrect template caused DAI to be inadvertently left out of the call in the module. No error resulted as it was in the COL and DEFINE statements and thus was blank.

(9) The template also caused “VERBATIM” to be wrongly indented since its label had no split character at its end.

(10) There are only 15 different COSTART body system codes, 12 of which are used in this report. It thus makes sense to code them by their unique first two characters and put the code in a footnote rather than listing them for each observation.
and the **split** character used. The macro creates another data step but the code could be changed so that it is inserted in an existing data step. This macro corrects the indentation and splitting of the 4th and 5th Figure 1 observations.

RESPIRATORY SYSTEM

COUGH INCREASED

Cough

MUSCULOSKELETAL SYSTEM

ARTHRALGIA

REVISED ADVERSE EVENT LISTING

proc report nowd data=_report split="'" missing spacing=1;

by invtext tpname;

col pat_age visit(''_Date'_ visitid strtdate stopdate)

prin incrym tese aescx toxgr actn outc fcoacen abs_rel

define pat_age/width=4 order "SUBJECT||AGE|(Y)|/SEX|__" spacing=0 flow;

define visit/width=9 "VISIT__(order);

define strtdate/f=date7."Start|__" center order internal;

define stopdate/f=date7."Stop|__" center;

define priyrm/width=1 "PRIOR?__;"

define tseo/width=1 "T.E.?__;"

define docx/width=1 "SEVERITY__;"

define actn/flow width=5 "INDEX__;"

define toxgr/flow width=4 "TOX__;"

define relat/flow width=5 "RELATIONSHIP__;"

define sev/flow width=4 "SEVERITY__;"

define toxgr/flow width=1 "NCITOX_;"

define visitdt/f=date6."DIE|__" center;

define visit/width=2 'VISIT__' order;

break after pat_age/skip;

run;

The new variables created and/or used for PROC REPORT are:

(1) v: adverse event concatenated with an indented verbatim.

(2) pat_age, which is the first two zeroes, stripped from patient age. This did not result in any extra output rows to v was already flowed.

(3) bs: body system - the first two characters of aescx.

Note from Figure 2 and the above code you can see:

(1) Labels are individually underlined by using a SPLIT character followed by 2 underline (1 underline for a width of 1).

(2) COL statement labels are spanned and made more explicit.

(3) The maximum possible width was used for the verbatim: This corrected all indenting problems and the SAS word split error.

(4) STUDY DAY value was left oriented. Putting a space after the SPLIT after STUDY centered the label.

(5) Observations were logically ordered by visit number and date.

(6) OUTC was added to the legend page.

(7) The order option was not used for stopdate and subsequent variables as it would be difficult to distinguish missing values from those that were identical to the previous observation.

(8) Superfluous label SPLIT character and options were removed

(9) Footnotes were replaced by a dynamic legend page

This reduced the output from 45 to 19 pages. Figure 2 shows data for a different patient to illustrate that the action code for an observation can have more than one value. Note that the first and third observations in Figure 1 are essentially replicates. While clinical data should not be edited, these replicates could be collapsed by (1) removing punctuation marks and case; (2) sorting the words in verbatim; (3) sorting the observations with the NODUPKEY option; and (4) outputting the first original verbatim.

**DYNAMIC LEGEND PAGE**

This does not directly reduce the width of a report but it reduces its page count over the use of footnotes repeated on every page. It must be used to explain (1) cryptic column headings created as a result of compacting the report and (2) the coded contents of a column. Creating the legend page dynamically (i.e., only showing the codes used in the report) allows you to check if any codes were inadvertently excluded from the legend page as well as eliminating codes which are extraneous to the study. The following is the legend page for Figure 2:

<table>
<thead>
<tr>
<th>Action</th>
<th>N</th>
<th>Body System</th>
<th>Severity</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>96</td>
<td>Concomitant Medication</td>
<td>Life Threatening</td>
</tr>
<tr>
<td>D</td>
<td>1</td>
<td>Discontinued Test Article</td>
<td>Life Threatening</td>
</tr>
<tr>
<td>H</td>
<td>2</td>
<td>Hospitalized</td>
<td>None</td>
</tr>
<tr>
<td>N</td>
<td>189</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>O</td>
<td>1</td>
<td>Other Action(s) Taken</td>
<td>None</td>
</tr>
<tr>
<td><strong>BO</strong></td>
<td>70</td>
<td>Body as a Whole</td>
<td>Life Threatening</td>
</tr>
<tr>
<td><strong>CA</strong></td>
<td>4</td>
<td>Cardiovascular System</td>
<td>Life Threatening</td>
</tr>
<tr>
<td><strong>DI</strong></td>
<td>75</td>
<td>Digestive System</td>
<td>Life Threatening</td>
</tr>
<tr>
<td><strong>HE</strong></td>
<td>16</td>
<td>Hemic &amp; Lymphatic System</td>
<td>Life Threatening</td>
</tr>
<tr>
<td><strong>ME</strong></td>
<td>30</td>
<td>Metabolic &amp; Nutritional</td>
<td>Life Threatening</td>
</tr>
<tr>
<td><strong>MU</strong></td>
<td>26</td>
<td>Musculoskeletal System</td>
<td>Life Threatening</td>
</tr>
<tr>
<td><strong>NE</strong></td>
<td>25</td>
<td>Nervous System</td>
<td>Life Threatening</td>
</tr>
<tr>
<td><strong>RE</strong></td>
<td>21</td>
<td>Respiratory System</td>
<td>Life Threatening</td>
</tr>
<tr>
<td><strong>SK</strong></td>
<td>7</td>
<td>Skin &amp; Appendages</td>
<td>Life Threatening</td>
</tr>
<tr>
<td><strong>SP</strong></td>
<td>3</td>
<td>Special Senses</td>
<td>Life Threatening</td>
</tr>
<tr>
<td><strong>UR</strong></td>
<td>6</td>
<td>Urogenital System</td>
<td>Life Threatening</td>
</tr>
</tbody>
</table>

Outcome | Dea | 2 | Death |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Per</td>
<td>176</td>
<td>Persisted</td>
<td>Life Threatening</td>
</tr>
<tr>
<td>Res</td>
<td>118</td>
<td>Resolved</td>
<td>Life Threatening</td>
</tr>
</tbody>
</table>

Relationship: DEF 1 Definitely
DNOT 16 Definitely Not
PNOT 46 Probably Not
POS 53 Possibly Not
PRB 11 Probably

Severity: Life 1 Life Threatening
Mild 2 Mild Threatening
Mod 2 Moderate
Sev 1 Severe

Normally, you would not put counts into a legend. However, since they automatically come out of PROC FREQ, you can see that they give some useful information such as two patients died and the severity data was quite sparse. The above dynamic legend page was created from the following master legend data set:

<table>
<thead>
<tr>
<th>Action</th>
<th>C</th>
<th>Concomitant Medication</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action</td>
<td>D</td>
<td>Discontinued Test Article</td>
</tr>
<tr>
<td>Action</td>
<td>H</td>
<td>Hospitalized</td>
</tr>
<tr>
<td>Action</td>
<td>N</td>
<td>None</td>
</tr>
<tr>
<td>Action</td>
<td>O</td>
<td>Other Action(s) Taken</td>
</tr>
<tr>
<td>Action</td>
<td>P</td>
<td>Primary Reason for Study Withdrawal</td>
</tr>
<tr>
<td>Action</td>
<td>R</td>
<td>Reduced Test Article Dose</td>
</tr>
<tr>
<td>Action</td>
<td>T</td>
<td>Temporarily Stopped Test Article</td>
</tr>
</tbody>
</table>

Body System: AD 

Body System: BO 

Body System: CA 

Body System: DI 

Body System: EN 

Body System: NE 

Body System: MU 

Body System: NE 

Body System: RE 

Body System: SK 

Body System: SP 

Body System: TE 

Body System: UR 

Outcome: Dea 

Outcome: Per 

Outcome: Res 

Outcome: Res 

Outcome: Per 

Outcome: Res 

Outcome: Per 

Outcome: Res 

Outcome: Per 

Outcome: Per 

Severity: LifeLife Threatening
Severity Mild Mild Threatening
Severity Mod Moderate
Severity Sev Severe

run;

Then use %legend, specifying the data set and coded variables:

%macro legend(data,v=);
%let i=0;
%do %while(;%length(%scan(&v,%eval(&i+1),' ')));
%let i=%eval(&i+1);
end;
data a;
  set legend(obs=0 keep=abbrev count);
  length name $8;
run;
%do %k=1 %to &i;
  run;
end;
%macro legend(data,v=);
%let c&i=%scan(&v,&i,' ');
%let i=%eval(&i+1);
data b;
  set b;
  length name $8;
run;
proc append base=a data=b force;
run;
%end;
data b;
run;
%do %while(%length(%scan(&v,%eval(&i+1),' ')));
%legend(data,a);
%do %k=1 %to &i;
  run;
end;
%do %k=1 %to &i;
  run;
end;
data a(keep=abbrev count where=(abbrev^=''))
  j=abbrev;
i=1;
do until(%scan(j,i,', ')='');
  abbrev=%scan(j,i,', ');
  output;
i=i+1;
end;
data a(keep=abbrev count where=(abbrev^=''))
  j=abbrev;
i=1;
do until(%scan(j,i,', ')='');
  abbrev=%scan(j,i,', ');
  if category=' ' then category=trim(name)||'?';
  output;
i=i+1;
end;
data a;
set legend(obs=0 keep=abbrev count);
length name $8;
set legend(obs=0 keep=abbrev count);
run;
proc append base=a data=b force;
run;
%end;

Note that %legend will identify codes that are not in the master legend data set with the name of the variable in which they were found. Thus by using a null master legend data set with an appropriate abbrev width, you can make a decision as to whether or not it makes sense to code a variable as well as determine what its codes should be. Also, it handles the observation in Figure 2 that has two action codes as long as the width of abbrev in the master legend data set is as big as the largest concatenated action. The width of the codes can be different as FORCE is used in PROC APPEND. Note a width need only be specified for count as master legend data set is as big as the largest concatenated actn.

Abbreviations Text
Severity Mild Mild Threatening
Severity Mod Moderate
Severity Sev Severe

run;

GENERALIZED ADVERSE EVENTS LISTING
Considering Figure 2, there is some justification for writing a validated listing program that would handle all drugs and protocols. Other projects may require additional variables (e.g. visit name, dose of drug at onset, duration of event, study analysis interval, etc.) and may require different sort orders and BY variables. To write a general program you must determine every possible variable that can be used for any project and give the user the ability to specify variables and their order. For each variable:
(1) Specify its label attractively with SPLIT characters.
(2) Calculate its width if not constant over drugs or else specify it.
(3) Check if it is blank (e.g., time) or missing and, if so, exclude it.
(4) Maximize the width of the verbatim variable.
(5) Decide whether it should be flowed (e.g. ACTN and DAI).
(6) Change to alphanumeric, concatenate, SPLIT and flow the 3 dates and times, first changing the time to blank if it is zero. These would have to be ordered by the NOPRINT datetime variable. Then the times would print under the ordered dates.
(7) From Figure 2, you can see that this report should be able to fit on a single page width with perhaps a narrowed verbatim. Let the user specify a predetermined list of sort orders within PROC REPORT and/or BY variables after sorting the data.

EGCTEST EXAMPLE
Wyeth has standardized SAS views for many different types of data for all drugs and protocols. EGCTEST contains 57 variables. A drug and protocol was chosen that had 29 ECG observations with the objective of printing this view without losing any information in the width of a single page. First, the data was carefully examined. 37 of the variables (1) had either the same value for each observation; (2) were either blank or missing; or (3) were meta variables which contained no useful information. A compressed PROC CONTENTS of the remaining is given below:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Format</th>
<th>W</th>
<th>Label</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABS_REL</td>
<td>5</td>
<td></td>
<td>Prim Relative Day</td>
<td>Integer</td>
</tr>
<tr>
<td>COUNTRY</td>
<td>3</td>
<td></td>
<td>Country</td>
<td>Abbreviated</td>
</tr>
<tr>
<td>CPENM</td>
<td>20.2</td>
<td>16</td>
<td>Planned Event Name</td>
<td>1 to 10, visit</td>
</tr>
<tr>
<td>DAI</td>
<td>3.9</td>
<td></td>
<td>Data Analysis Interval</td>
<td>1 to 10, d, ord</td>
</tr>
<tr>
<td>DAI ORD</td>
<td>1</td>
<td></td>
<td>DAI Order</td>
<td>1 to 10, d, ord</td>
</tr>
<tr>
<td>INVEST</td>
<td>10.5</td>
<td></td>
<td>Investigator</td>
<td>Uninformative</td>
</tr>
<tr>
<td>LVALC</td>
<td>200.15</td>
<td></td>
<td>answer char</td>
<td>related sasname</td>
</tr>
<tr>
<td>MILESTN</td>
<td>30.1</td>
<td></td>
<td>milestone name</td>
<td>1 char different.</td>
</tr>
<tr>
<td>PATIENT</td>
<td>10.5</td>
<td></td>
<td>patient id</td>
<td>1 leading 0's</td>
</tr>
<tr>
<td>OVALUE</td>
<td>70.3</td>
<td></td>
<td>label type qualifying</td>
<td>1 to 10, visit</td>
</tr>
<tr>
<td>REGIMEN</td>
<td>30.5</td>
<td></td>
<td>Regimen</td>
<td>blank or tname</td>
</tr>
<tr>
<td>RELDAY</td>
<td>30.5</td>
<td></td>
<td>time from milestone</td>
<td>Integer</td>
</tr>
<tr>
<td>SASNAME</td>
<td>8.6</td>
<td></td>
<td>sas name</td>
<td>1 to 10, test</td>
</tr>
<tr>
<td>STDYSITE</td>
<td>10.3</td>
<td></td>
<td>Study site</td>
<td>Uninformative</td>
</tr>
<tr>
<td>STINT</td>
<td>30.25</td>
<td></td>
<td>planned study interval</td>
<td></td>
</tr>
<tr>
<td>TEST</td>
<td>20.12</td>
<td></td>
<td>Question name</td>
<td>1 to 10, sasname</td>
</tr>
<tr>
<td>TPCODE</td>
<td>10.3</td>
<td></td>
<td>Therapy code</td>
<td>1 to 10, tname</td>
</tr>
<tr>
<td>TPCODE</td>
<td>20.3</td>
<td></td>
<td>Therapy text</td>
<td>4° &amp; 5° char diff</td>
</tr>
<tr>
<td>VISIT</td>
<td>6</td>
<td></td>
<td>WAR number</td>
<td>1 to 10, openm</td>
</tr>
<tr>
<td>VISITDT</td>
<td>DATETIME20</td>
<td></td>
<td>Visit date/time</td>
<td>No times entered</td>
</tr>
</tbody>
</table>

(As an aside, note that PROC PRINT with a PS of 165, after adding page breaks, will yield a neat columnar output of these variables.) The maximum width, W, was calculated for each variable. While this is appreciably less that that of the formats, the total width is still too big to fit on a single page. The code for investigator and site is not informative. These two variables must be translated via a merge with a patient information view to the actual names. COUNTRY is abbreviated to DEU for “Deutchland” and ESP for "España". Replacing these with Germany and Spain would be more informative. There were only 3 investigators and two sites, each with rather long names, so I concatenated these variables with country into INV. Since times were not entered, the data can be reduced to a width of 6 using MMDDYY6. TPNAME has values of “EKB XXmg+FU/LV/IRINOT 180 mg” where XX is the rest of the characters in the variable; and (3) let REGIMEN be the only thing that varies from observation to observation. Thus, these two variables could be reduced to 2 characters plus an explanation in a footnote. I chose instead to (1) put “EKB” in the label, (2) leave the rest of the characters in the variable; and (3) let REGIMEN be either “Same” or blank. TPCODE, which had a one to one relationship to TPNAME, was dropped. The first 4 zeroes were related sasname

Other projects may require additional variables (e.g. visit name, dose of drug at onset, duration of event, study analysis interval, etc.) and may require different sort orders and BY variables. To write a general program you must determine every possible variable that can be used for any project and give the user the ability to specify variables and their order. For each variable:
(1) Specify its label attractively with SPLIT characters.
(2) Calculate its width if not constant over drugs or else specify it.
(3) Check if it is blank (e.g., time) or missing and, if so, exclude it.
(4) Maximize the width of the verbatim variable.
(5) Decide whether it should be flowed (e.g. ACTN and DAI).
(6) Change to alphanumeric, concatenate, SPLIT and flow the 3 dates and times, first changing the time to blank if it is zero. These would have to be ordered by the NOPRINT datetime variable. Then the times would print under the ordered dates.
(7) From Figure 2, you can see that this report should be able to fit on a single page width with perhaps a narrowed verbatim. Let the user specify a predetermined list of sort orders within PROC REPORT and/or BY variables after sorting the data.
It is obvious that SASNAME and TEST have a one to one relationship as do VISIT and CPENM (visit name). A close scrutiny of TEST shows that it either tells you whether the EKG was normal or whether there was a change from the baseline VISIT 1. Thus, the 15 character LVALC can be reduced to a one character Y or N. TEST is thus labeled EKG and given the six-character value of “Normal” or “Change”. MILESTN can be reduced to, MILES, its seventh character. CPENM can be abbreviated to “Screen” and “Final” without losing meaning Figure 3, with 132 characters, was generated by the following data step and PROC REPORT code:

```sas
data test(drop=patient visitdt sasname lvalc cpenm regimen tpname milestn);
length pt $2 ecg visit $6 yn $1 reg $4 tp $24 land $7;
pt=substr(patient,5);
tp=substr(tpname,5);
miles=substr(milestn,7,1);
date=datepart(visitdt);
if sasname=:'O' then ecg=’Normal’;
else ecg=’Change’;
if lvalc=’Nor’ then yn=’Y’;
else yn=’N’;
if cpenm=’S’ then visit=’Screen’;
else visit=’Final’;
if regimen=’ ’ then reg=’Same’;
else reg=’ ’;
if country=’DEU’ then land=’Germany’;
else land=’Spain’;
run;
```

The first variable in the COL and DEFINE statements is COUNTRY with the ORDER and NOPRINT options. This causes the data to be sorted by COUNTRY rather than alphabetically by investigator. Since, the first column is not printed, the SPACING=0 option is not needed as long as the ID option is not used. As was the case in the previous example, the ORDER variables are only printed if their value changes. TP and REG are labeled in the COL statement rather than in the DEFINE. For, illustrative purposes, SPACING=0 was used to label MILES over the space between columns as was the spanned “Relative Day”. The 2 character PT was given a 4 character label.

**GENERALIZED ECGTEST COMPACT REPORT**

It should be reiterated that this example, unlike the previous one, is presented only for illustrative purposes. Still, many of the considerations for generalizing the adverse events listing are valid. To create code for listing ECGTEST for all drugs and protocols:

- Decide what variables to always exclude.
- Count and remove applicable leading zeroes.
- Determine what data condensing tricks work for all cases.
- Predetermine which variables need FLOW.
- Determine which variables have a fixed width.
- Calculate the width of all remaining variables.

It is unlikely that the data can be squeezed into the width of a single page for all ECGTEST views. You can get more space to list the variables by either using the ID option or using a BY statement for the investigator. In this example, the ID option would list all data on two page widths while the BY statement would require three page widths. Since the BY statement was already illustrated in the adverse events example, let us consider the ID option. If you (1) add the NOCENTER option to the PROC REPORT; (2) increase the width of INV to 31 (so that investigator only prints on two lines) and add the SPACING=0 option; and (3) add the ID option to VISIT, you get the output in Figure 4. (The title and labels, which are identical to those in Figure 3, were stripped and only the first ten observations of the second page included to save space). This output is not balanced in that the second page has a lesser width than the first page. However, it is assumed that the reason for adding the ID option was to widen and add variables with the output still fitting neatly on two pages. There is sufficient room to do this and still put additional spaces between the columns to balance the two page widths.

**CONCLUSION**

It does not take an undue amount of effort to create an attractive compact columnar output using PROC REPORT. It does take a reasonable knowledge of the options, but this knowledge is not particularly difficult to acquire and definitely worth the effort. You should never generate a report by copying it from a template without understanding the basic principles of PROC REPORT.

**REFERENCES**

4. Hjelle, Dean, support.sas.com (12/00) “Quick Tip: Fitting More Variables on a Page”

**CONTACT INFORMATION**

Walter Young
Wyeth
PO Box 42528
Philadelphia, PA 19101-2528
(484) 865-5640
youngw@wyeth.com

It’s NOT IMPORTANT if you CAN read it or NOT. WHAT MATTERS is THAT it ALL FITS on the PAGE.
### Figure 1: Original Adverse Event Listing Stripped of Titles

| Subject | Sex | Age(Y) | O/S | T/T | ADVERSE EVENT | NCI DRUG | ACTION CODES | REL. DRUG REL. | TIME START | TIME STOP | STUDY | COM Analysis | Data Analysis | DAY(D) | Interval |
|---------|-----|--------|-----|-----|----------------|----------|--------------|---------------|-------------|------------|--------|------|-------------|--------------|--------|----------|
| 001001  | 69/M | Y     | N   | N   | DIGESTIVE SYSTEM | 1 DNOT C | Per 01APR2002 | -180          |             |            |        |               |              |        |          |
|         | Y N | N   | DIGESTIVE SYSTEM | 1 DNOT C | Res 01APR2002 | 03JAN2003 | -180          |               |             |            |        |               |              |        |          |
|         | Y N | N   | DIGESTIVE SYSTEM | 1 DNOT C | Per 01APR2002 | -180          |               |             |            |        |               |              |        |          |
|         | Y N | N   | RESPIRATORY SYSTEM | 1 DNOT N | Res 03SEP2002 | 06OCT2002 | -25           |               |             |            |        |               |              |        |          |
|         | Y N | N   | MUSCULOSKELETAL SYST | 1 DNOT C | Per 13SEP2002 | -15           |               |             |            |        |               |              |        |          |
|         | Y N | N   | MUSCULOSKELETAL SYST | 1 DNOT C | Res 13SEP2002 | 03JAN2003 | -15           |               |             |            |        |               |              |        |          |
|         | Y N | N   | MUSCULOSKELETAL SYST | 1 DNOT C | Per 13SEP2002 | -15           |               |             |            |        |               |              |        |          |
|         | Y N | N   | HEMIC AND LYMPHATIC | 1 DNOT C | Per 24SEP2002 | -4            |               |             |            |        |               |              |        |          |

### Figure 2: Final Adverse Event Listing

<p>| Subject | Sex | Age(Y) | O/S | T/T | ADVERSE EVENT | NCI DRUG | ACTION CODES | REL. DRUG REL. | TIME START | TIME STOP | STUDY | COM Analysis | Data Analysis | DAY(D) | Interval |
|---------|-----|--------|-----|-----|----------------|----------|--------------|---------------|-------------|------------|--------|------|-------------|--------------|--------|----------|
| 1006 2  | 04DEC02 | 01MAR02 | Y N | B O | ABDOMINAL PAIN | 1 DNOT C | Per 279 baseline | -299 baseline |             |            |        |               |              |        |          |
| 9 26DEC02 | 04DEC02 | 04DEC02 | N | DI | ESOPHAGITIS | 1 DEF N Re | 1 baseline |             |            |        |        |               |              |        |          |
| 21DEC02 | 22DEC02 | N | Y | U R | URINARY TRACT DISORDER | 3 DNOT C Res | 17 cycle 1 |               |             |            |        |        |               |              |        |          |
| 26DEC02 | N | Y | BO | BACK PAIN | 2 DNT C Per | 22 cycle 1 |               |             |            |        |        |               |              |        |          |
| 16 16JAN03 | 01SEP02 | Y N | DI | ANOREXIA | 1 DNT N Per | -95 baseline |               |             |            |        |        |               |              |        |          |
| 11JAN03 | Y N | DI | FLATULENCE | 1 DNT N Res | -95 baseline |               |             |            |        |        |               |              |        |          |
| 26DEC02 | 28DEC02 | N | Y | BO | BACK PAIN | 2 DNT C Res | 22 cycle 1 |               |             |            |        |        |               |              |        |          |
| 28DEC02 | 29DEC02 | N | Y | BO | BACK PAIN | 3 DNT C Res | 24 cycle 2 |               |             |            |        |        |               |              |        |          |
| 29DEC02 | 06JAN03 | N | Y | BO | BACK PAIN | 2 DNT C Res | 25 cycle 2 |               |             |            |        |        |               |              |        |          |
| 03JAN03 | 11JAN03 | N | Y | DI | CONSTIPATION | 2 POS C Res | 30 cycle 2 |               |             |            |        |        |               |              |        |          |
| 11JAN03 | 12JAN03 | N | Y | DI | FECAL IMPACTION | 3 DNT H,O Res | 38 cycle 2 |               |             |            |        |        |               |              |        |          |
| 12JAN03 | 13JAN03 | N | DI | FECAL IMPACTION | 3 DNT C Res | 38 cycle 2 |               |             |            |        |        |               |              |        |          |
| 12JAN03 | 15JAN03 | N | DI | CONSTIPATION | 2 POS C Res | 39 cycle 2 |               |             |            |        |        |               |              |        |          |
| 19 06FEB03 | 16DEC02 | 25JAN03 | N | Y | BO | ASTHENIA | 1 PB N Res | 12 cycle 1 |               |             |            |        |        |               |              |        |          |
| 18DEC02 | N | Y | NE | ANEMIA | 1 PB N Per | 14 cycle 1 |               |             |            |        |        |               |              |        |          |
| 17JAN03 | 17JAN03 | N | Y | DI | NAUSEA | 1 POS C Res | 44 cycle 3 |               |             |            |        |        |               |              |        |          |
| 17JAN03 | 18JAN03 | N | Y | DI | VOMITING | 1 POS C Res | 44 cycle 3 |               |             |            |        |        |               |              |        |          |
| 20JAN03 | 20JAN03 | N | Y | ** Classification Unknown** | 2 PB C Res | 47 cycle 3 |               |             |            |        |        |               |              |        |          |
| 20JAN03 | N | Y | NE | ANXIETY | 1 DNT C Per | 47 cycle 3 |               |             |            |        |        |               |              |        |          |
| 24JAN03 | N | Y | DI | ANOREXIA | 1 POS N Per | 51 cycle 3 |               |             |            |        |        |               |              |        |          |</p>
<table>
<thead>
<tr>
<th>Investigator</th>
<th>Date</th>
<th>Name</th>
<th>ECG</th>
<th>Abs Miles</th>
<th>EKB Therapy</th>
<th>Regimen</th>
<th>Planned Study Interval</th>
<th>Planned Study Interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>Koehne, Klaus Medizinische Klinik und Poliklinik I Germany</td>
<td>02 041702</td>
<td>Screen Normal Y</td>
<td>-6</td>
<td>-6</td>
<td>10mg+FU/LV/IRINOT</td>
<td>180 mg</td>
<td>1 Baseline</td>
<td>Pre-study Screening</td>
</tr>
<tr>
<td>Cortes- Funes, Hernan Servicio de Oncologia Spain</td>
<td>31 052902</td>
<td>Screen Normal Y</td>
<td>-10</td>
<td>-10</td>
<td>25mg+FU/LV/IRINOT</td>
<td>180 mg</td>
<td>1 Baseline</td>
<td>Pre-study Screening</td>
</tr>
<tr>
<td>Tabernero, Josep Servicio de Oncologia Spain</td>
<td>41 041902</td>
<td>Screen Normal Y</td>
<td>-5</td>
<td>-5</td>
<td>10mg+FU/LV/IRINOT</td>
<td>180 mg</td>
<td>1 Baseline</td>
<td>Pre-study Screening</td>
</tr>
<tr>
<td>Koehne, Klaus Medizinische Klinik und Poliklinik I Germany</td>
<td>03 091802</td>
<td>Final Normal Y</td>
<td>141</td>
<td>43</td>
<td>10mg+FU/LV/IRINOT</td>
<td>180 mg</td>
<td>Cycle 3-4</td>
<td>Cycle 3-4</td>
</tr>
<tr>
<td>Cortes- Funes, Hernan Servicio de Oncologia Spain</td>
<td>33 070902</td>
<td>Screen Normal Y</td>
<td>-4</td>
<td>-4</td>
<td>50mg+FU/LV/IRINOT</td>
<td>180 mg</td>
<td>1 Baseline</td>
<td>Pre-study Screening</td>
</tr>
<tr>
<td>Tabernero, Josep Servicio de Oncologia Spain</td>
<td>42 041602</td>
<td>Screen Normal Y</td>
<td>-8</td>
<td>-8</td>
<td>10mg+FU/LV/IRINOT</td>
<td>180 mg</td>
<td>1 Baseline</td>
<td>Pre-study Screening</td>
</tr>
<tr>
<td>Koehne, Klaus Medizinische Klinik und Poliklinik I Germany</td>
<td>04 052802</td>
<td>Screen Normal N</td>
<td>-2</td>
<td>-2</td>
<td>25mg+FU/LV/IRINOT</td>
<td>180 mg</td>
<td>1 Baseline</td>
<td>Pre-study Screening</td>
</tr>
<tr>
<td>Cortes- Funes, Hernan Servicio de Oncologia Spain</td>
<td>36 10082</td>
<td>Screen Normal Y</td>
<td>-3</td>
<td>-3</td>
<td>75mg+FU/LV/IRINOT</td>
<td>180 mg</td>
<td>1 Baseline</td>
<td>Pre-study Screening</td>
</tr>
<tr>
<td>Tabernero, Josep Servicio de Oncologia Spain</td>
<td>44 053102</td>
<td>Screen Normal Y</td>
<td>-6</td>
<td>-6</td>
<td>25mg+FU/LV/IRINOT</td>
<td>180 mg</td>
<td>1 Baseline</td>
<td>Pre-study Screening</td>
</tr>
<tr>
<td>Koehne, Klaus Medizinische Klinik und Poliklinik I Germany</td>
<td>05 06052</td>
<td>Screen Normal N</td>
<td>56</td>
<td>56</td>
<td>25mg+FU/LV/IRINOT</td>
<td>180 mg</td>
<td>1 Baseline</td>
<td>Pre-study Screening</td>
</tr>
<tr>
<td>Cortes- Funes, Hernan Servicio de Oncologia Spain</td>
<td>37 070902</td>
<td>Screen Normal Y</td>
<td>-4</td>
<td>-4</td>
<td>50mg+FU/LV/IRINOT</td>
<td>180 mg</td>
<td>1 Baseline</td>
<td>Pre-study Screening</td>
</tr>
<tr>
<td>Tabernero, Josep Servicio de Oncologia Spain</td>
<td>43 052802</td>
<td>Screen Normal N</td>
<td>-2</td>
<td>-2</td>
<td>25mg+FU/LV/IRINOT</td>
<td>180 mg</td>
<td>1 Baseline</td>
<td>Pre-study Screening</td>
</tr>
<tr>
<td>Koehne, Klaus Medizinische Klinik und Poliklinik I Germany</td>
<td>06 070902</td>
<td>Screen Normal N</td>
<td>56</td>
<td>56</td>
<td>25mg+FU/LV/IRINOT</td>
<td>180 mg</td>
<td>1 Baseline</td>
<td>Pre-study Screening</td>
</tr>
<tr>
<td>Cortes- Funes, Hernan Servicio de Oncologia Spain</td>
<td>38 071502</td>
<td>Screen Normal Y</td>
<td>-3</td>
<td>-3</td>
<td>50mg+FU/LV/IRINOT</td>
<td>180 mg</td>
<td>1 Baseline</td>
<td>Pre-study Screening</td>
</tr>
<tr>
<td>Tabernero, Josep Servicio de Oncologia Spain</td>
<td>45 073102</td>
<td>Screen Normal Y</td>
<td>-6</td>
<td>-6</td>
<td>25mg+FU/LV/IRINOT</td>
<td>180 mg</td>
<td>1 Baseline</td>
<td>Pre-study Screening</td>
</tr>
<tr>
<td>Koehne, Klaus Medizinische Klinik und Poliklinik I Germany</td>
<td>07 071502</td>
<td>Screen Normal Y</td>
<td>-3</td>
<td>-3</td>
<td>50mg+FU/LV/IRINOT</td>
<td>180 mg</td>
<td>1 Baseline</td>
<td>Pre-study Screening</td>
</tr>
<tr>
<td>Cortes- Funes, Hernan Servicio de Oncologia Spain</td>
<td>48 101002</td>
<td>Screen Normal N</td>
<td>-2</td>
<td>-2</td>
<td>75mg+FU/LV/IRINOT</td>
<td>180 mg</td>
<td>1 Baseline</td>
<td>Pre-study Screening</td>
</tr>
<tr>
<td>Tabernero, Josep Servicio de Oncologia Spain</td>
<td>42 041602</td>
<td>Screen Normal Y</td>
<td>-8</td>
<td>-8</td>
<td>10mg+FU/LV/IRINOT</td>
<td>180 mg</td>
<td>1 Baseline</td>
<td>Pre-study Screening</td>
</tr>
</tbody>
</table>

**Figure 3: Compact ECGTEST Output With ID Option for Visit Name**

**Figure 4: Compact ECGTEST Output With ID Option for Visit Name**