



Section 3: ACL Scripts

Section Objectives

At the end of this section, you should be able to:

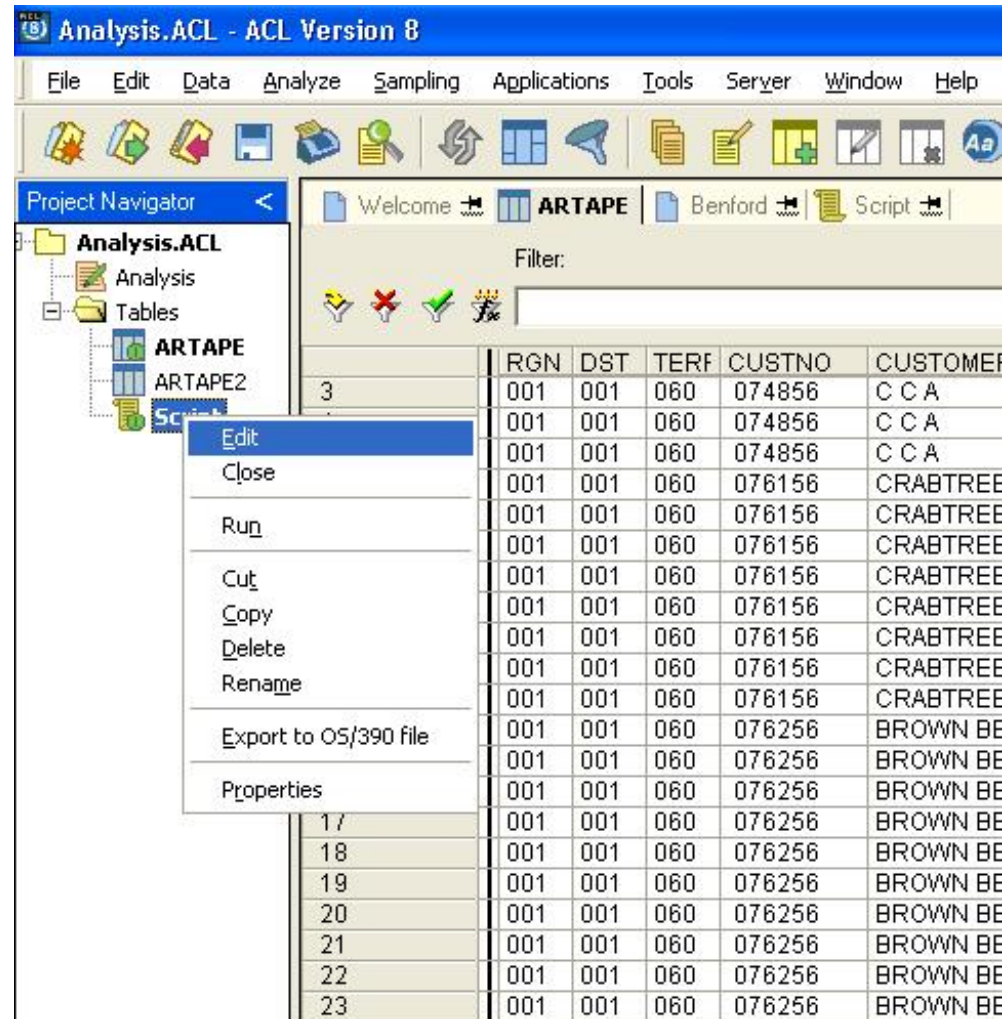
- Describe scripts
- Create a script within ACL
- Describe Command Syntax formation

Scripting

- A script is a series of ACL commands stored as a unit in an ACL project.
 - This series of commands can be executed repeatedly and automatically.
 - Any ACL command can be stored within a script.
- Scripts are stored in the Overview in the same manner as other ACL data objects.
 - To edit – Locate the script to be edited and double-click it
 - To run – Locate the script to be run, right-click and select RUN
 - To run from Menu Bar – Select Tools > Run Script and select the script to be run

Scripting

- To edit a script, double-click from the Project Navigator window or right-click on the script and select edit.



Creating Scripts

- Using the Script Recorder
 - As commands are executed using the GUI, the commands are simultaneously written to the script.
 - Select **Tools > Set Script Recorder On.**
 - All subsequent commands are captured.
 - To turn off the script recorder, select **Tools > Set Script Recorder On.**
- Manually editing a script
 - There are three ways to populate scripts with commands:
 - Capture command syntax – click this button and as menu commands are selected, their syntax is written to the script
 - Copy and paste from the log into a script – select the commands from the log desired, right-click and select copy and then paste into a script
 - Type commands manually – requires familiarity with the ACL command language

Creating Scripts

- Command Syntax
 - ACL command syntax is the way in which commands are assembled in a script.
 - Contains three key aspects:
 - The command name – shown here in red
 - Parameter names – shown here in green
 - Parameter values – shown here in blue
 - An example of this would be:

```
CLASSIFY ON loc SUBTOTAL value IF qty > 5 TO "Output.FIL"
```
- Syntax Rules
 - Most commands should be written on one line.
 - The order of parameters does not matter.
 - Some parameters are optional and do not need to be included for execution.

Guided Exercise – Task 1

Calculated Field Definition

Open the SourceSAS99 file from the Training.acl project, and define the following computed field:

Name	Definition
valAbsAmt	ABS(valAmt)

What is the total for valAbsAmt?

ACL concepts usage:

- Edit table layout
- ABS()
- Total

Guided Exercise – Task 2

Summarize on Document Type

Develop a summary of the source data by txtDocType with the accumulation (subtotaling) of any numeric fields in the source data. Output should be sent to an ACL table to allow review. Complete this task using a script.

Key fields in the table: txtDocType

ACL concepts usage:

- Summarize
- Scripting

SUMMARIZE syntax:

```
SUMMARIZE ON key-fields <SUBTOTAL summarize-fields>  
<WHILE test> <PRESORT> <OPEN>
```

Guided Exercise – Task 3

Statistics on valAmt

Perform a statistical analysis on valAmt including the standard deviation. Output should be sent to a text file to allow review. Complete this task using a script.

Key fields in the table:

valAmt

ACL concepts usage:

- Statistics
- Scripting

Guided Exercise – Task 4

User ID Analysis

Import EmployeeFile.xls and name it Employees. Join the SourceSAS99 file with the Employees file bringing all of the fields from the SourceSAS99 file and Department from the Employees file into the new table. Export the records to Excel that were not made by a user in the Accounting department. Try completing this task using a script.

1. How many records were exported to Excel?
2. What are the totals of valAmt and valAbsAmt?

ACL concepts usage:

- Join, Export, Scripting, Count, Total

Section Summary

- A script is a series of ACL commands stored as a unit in an ACL project.
- Scripts are stored in the Overview in the same manner as other ACL data objects.
- There are three ways to populate scripts with commands:
 - Capture command syntax
 - Copy and paste from the log into a script
 - Type commands manually
- ACL command syntax is the way in which commands are assembled in a script.