

CST8177 - Lab 12

Script Design and Implementation

Student Name	Student number	Lab section:

Objective

To practice problem-solving skills with several small scripts and their PDL, Data Dictionary (where necessary), and so on. Note: Print out your finished PDL, scripts , and test output to include in your Lab Book.

In-Lab Demo -- Demo the scripts from this lab, explaining the behaviour.

Creating script headers

The purpose of the script, named **header**, is to set up a script file with a header for you: it creates an otherwise empty script file with the comment header, writing the script header and the hash-bang line to the file, and makes the script executable. Additionally, the script can use either default header information from your configuration file or override the defaults by providing new information .

Task #1:

Create a configuration file **header.conf** in an **etc** directory in your home directory that sets the following variables, using meaningful names for these default values as they apply to you:

- script interpreter (that would be **/bin/bash**)
- your name
- your student number
- this course name
- the course id
- your lab section number
- your default script directory (**bin** in your home directory)

Task #2:

Write the PDL and then a short script to:

- If the configuration file exists, use **source** to include its contents in this script.
- Prompt for the file name of the new script file you are creating. Optional extra: accept the new file name from a command-line argument if present, otherwise prompt for and read it. If the file name already exists in the default script directory, request Replace, Quit, or New name.
- Create the file in the default script directory, writing only the default values from **header.conf**.
- If file creation was successful, make it executable. Otherwise, display an error message and exit with an error value.

Note: Suppress all output from the utilities: you control the messages displayed on the screen.

Task #3:

Modify your PDL, Data Dictionary, and script as follows:

Display one item at a time from the list and prompt the user for a replacement value. Also request the date the script is due and its brief purpose. I suggest a tidy display; end your prompt with a space for improved readability.

Example:

```
Enter configuration (press Enter for default)
Script Interpreter  [/bin/bash]:
Student name       [Patrick Ouellette]:
Student number     [040-001-002]:
Course name        [Linux Operating System II]:
Course number      [CST8177]:
Lab Section Number: [315]:
Script Directory:  [~/bin]:
Due date:
Purpose:
New Script Name:
```

When the user has entered all the necessary information, create the file in the final script directory (check that a replacement directory is valid before trying to use it) unless the requested script name is already in use (see above).

Note that the New Script Name, the Due Date, and the Purpose must always be entered.

Display the contents of the newly-created script file and request the user's approval. If it's not approved, delete the file just created.

Task #4: Simplified Test Plan

- Test run #1: Use only default data, do not override defaults;
- Test run #2: Use custom data for at least half the items;
- Test run #3: use a custom Script Directory.

Task #5: Optional Extras

- At user request, replace the default **header.conf** configuration file with the new values.
- Error check all the replacement information.
- Use function(s) to simplify the logic of the final script.
- Allow the user to choose which configuration file to use (for example, one for this class, a different one for another class, and so on).