# Lab 6 – Automating a starting File Structure

## Part 1

Automating tasks is a major reason why we may decide to create a PowerShell Script. In this case, a system administrator wants to create a script that builds a starting folder structure for a new project managers at our organization.

The desired folder structure looks like the following:

Write a basic PowerShell script that will create the above folder structure to the C:/ directory. Verify that you have created the correct file structure and save this script as, **lab6-1.ps1**.

## Part 2

Save a new copy of your previous script and save it as, **lab6-2.ps1**. Delete your site folders that have been created when testing your script above. Using your previous script, expand on it by placing a file called allowed-asset-types.txt inside of the Assets folder after the directory structure has been made. Hint – create the file in the C:\ Directory and copy the file.

Next, revise your program to accept input through a parameter when a user runs the script specifying the location where the site structure will be created. If the program is not provided a directory to copy to, make the default directory C:\

For example, running:

.\lab6-2.ps1 C:\Users\someuser\

will add the directory structure in the someuser directory.

Verify that your script is running correctly and save it as, **lab6-2.ps1**. Take a screenshot of the resulting file structure named **lab6-2.**

## Part 3

Your new file structure script is working great. You now want to customize this script to create different file structures for different users. Your previous file structure was designed for consultants. We want to build a slightly different file structure for accountants. The accountant file structure will look like the following:

Allow the user running the script to provide a new argument specifying the employee type (manager, or accountant) when running the script.

For example, running:

.\lab6-2.ps1 C:\Users\someuser\ accountant

will add the accountant directory structure in the someuser directory.

Use an if statement to check the input and build the correct directory structure. Test your program for both employee types, verify that your script is running correctly and save it as, **lab6-3.ps1**. Take a screenshot of the resulting file structure named **lab6-3**.

## Part 4

Accountants create weekly reports. (52 reports per year). Revise your file structure that is created for accountants to now create 52 different .csv files in the Budgets directory. The file names should be Week1.csv, Week2.csv, Week3.csv, …Week52.csv

Use a loop to create these 52 new files. Be sure to only create these files for Accountants.

When you are finished, verify that your script is running correctly and save it as, **lab6-4.ps1**. Take a screenshot of the resulting file structure of the Budgets directory (with your 52 files) named **lab6-4**.

When finished with all 4 components, upload the following scripts and screenshots to Canvas for grading:

* Scripts – lab6-1.ps1, lab6-2.ps1, lab6-3.ps1, lab6-4.ps1
* Screenshots - lab6-2, lab6-3, lab6-4