singularsystems
simplifying complexity

28 Fort Street, Birnam, JHB 2196 / PO Box 785261, Sandton 2146 / **Tel:** +27 (0) 10 003 0700 / **Fax:** +27 (0) 11 885 3835 / **Email:** info@singular.co.za / **www.singular.co.za**

SINGULAR SYSTEMS

PRACTICAL INTERVIEW - DEVOPS

INSTRUCTIONS

The practical assessment aims to assess your ability to solve technical problems.

What to submit

Cloud storage link to the zipped solution, e.g. Dropbox

Technologies to use

- PowerShell 7
- Suggested Editor: Visual Studio Code (PowerShell extension recommended)
- HTML (Bonus objective)

Guidelines

- Ensure that you are the creator of the project and that you did not use existing projects found online.
- This question does not aim to be over prescriptive - create a solution that you feel comfortable with to accurately portray your skillset.
- You may research and use technologies that you are not too familiar with.
- Use of advanced PowerShell features such as functions, classes, pipelines, etc will be rewarded.
- Have fun!

SCENARIO

You have been tasked with downloading and analysing an application’s log files in order to provide a report on the number of info, warning and error messages being logged per month. The log files sit online in an Azure storage account. An index file containing a list of the log files is located here:
https://files.singular-devops.com/challenges/01-applogs/index.txt

The log files sit under the same folder as the index file. They are stored in a fixed width CSV format. Some basic schema detail is available here:
https://files.singular-devops.com/challenges/01-applogs/schema.md

TASKS

Using PowerShell, write a script which performs the following actions:

- Download and read the contents of the index file
- Use the index file to generate links for and download each of the application log files and save them to a local folder in the current working directory called logs.
- Run through the contents of each log file and extract the following information:
    - The month and year
    - The number of info, warning and error messages
- Generate a report file in JSON format which contains an array of the monthly statistics:
    - The year and month
    - Number of info, warning and error messages
    - The percentage increase or decrease in warnings and errors from the previous month
- Save the report file as a file called report.json in a report folder under the current working directory.

BONUS OBJECTIVE: In addition to the JSON file, generate a human readable HTML report as a file called report.html in a report folder under the current working directory. The styling can be kept very basic.