Project Report: File Organizer

Your Name

June 14, 2023

1 Introduction

This report provides an overview of the File Organizer project, including its structure, code specifics, and highlights from the development process.

2 Project Overview

The File Organizer is a bash script designed to organize files within a directory based on specified criteria. It provides options for organizing files by extension or creation date, as well as the ability to exclude specific file types or directories. Additionally, it offers the option to compress files with a particular extension into a tarball.

3 Code Structure

The code for the File Organizer begins with defining several functions to achieve the desired functionality:

- `print_usage()` - Prints usage instructions and available options.
- `create_folder()` - Creates a folder if it doesn't exist.
- `move_file()` - Moves or copies files to the destination folder.
- `compress_files()` - Compresses files in a folder using tar.

The script also includes a command-line argument parser to handle various options, such as organization style, file exclusion, deletion of original files, and generating a log file.

4 Code Specifics

The code utilizes the `getopts` command[1] to extract options given to the bash file and parse the arguments to suitable variables. It uses the `find` command to

traverse the source directory and locate files. It then processes each file based on the selected organization style or compresses files with specific extensions into tarballs.

The `move_file()` function moves or copies files to the appropriate destination folder based on their extension or creation date. It also handles cases where a file with the same name already exists in the destination folder by appending a counter to the filename.

The `compress_files()` function compresses files in a folder using the `tar` command, creating a compressed tarball[4] with the specified output file name.

5 Journey and Lessons Learned

I made a Github Repository File-Organiser/git on which I Kept the various drafts of code updated. The initial Draft contained a rudimentary organiser with just a -d option. After discovering the `getopts` command, specifying options became a lot easier. Handling the files with no extension proved to be challenging every time the code structure changed. Eventually, with the help of some online references[3], I was able to handle it.

Throughout the development process of the File Organizer, several key insights were gained:

- Planning and designing the script structure upfront helped in achieving modularity and code reusability.
- Various handy flags were discovered like -n,-e,-f etc.
- Utilizing existing command-line tools like `find` and `tar` simplified the implementation and improved efficiency.

6 Conclusion

The File Organizer script offers a convenient solution for organizing files based on various criteria, providing flexibility and efficiency in file management. The journey of developing this project has enhanced scripting skills and deepened understanding of command-line tools.

References

[1] *Resource for getopts.* URL: `https://www.geeksforgeeks.org/getopts-command-in-linux-with-examples/`.

[2] *Stackoverflow page for counting no. of files recursively.* URL: `https://stackoverflow.com/questions/15216370/how-to-count-number-of-files-in-each-directory`.

[3] *Stackoverflow page for extracting extensions.* URL: `https://stackoverflow.com/questions/965053/extract-filename-and-extension-in-bash`.

[4] *Webpage for tar usage*. URL: https://www.cyberciti.biz/faq/how-to-tar-a-file-in-linux-using-command-line/.