

Mastering Shell Scripts: Listing Files in the Current Directory

Welcome to our presentation on creating a powerful shell script to list all files in the current directory. Let's dive in and learn some handy scripting skills!

 by Deepak Srinivas

```
ubuntu: ~
vim ListDir.sh
chmod +x ListDir.sh
./ListDir.sh
loads          ListDir.sh  Picture
ples.desktop   Music
the list of directories
|
```

Objective

Our objective is to create a shell script that displays a comprehensive list of all files in the current directory, making it easier to manage and navigate file systems.

Instructions for Creating the Shell Script

1. Create a shell script that displays a list of all the files in the current directory.
 - The script should use appropriate commands to list the files in the current directory.
 - Save the script to a file, for example, list_files.sh.
 - Make the script executable using chmod +x list_files.sh.
 - When executed, the script should display a list of files in the current directory.

```
#!/bin/bash
echo "list of files in current directory"
for file in *: do
echo $ file
done
```

Example Output of the Script

Upon executing the shell script, you will witness the magic unfold as a list of files in the current directory is displayed, making file management effortless and convenient.

List of files in the current directory:

file1.txt

file2.jpg

directory1

Exploring Advanced Options

Customize the Output

Learn how to modify the output format to display additional information such as file sizes, permissions, and timestamps.

Manage Files with Filtering

Discover techniques to filter and search for specific files using powerful command-line tools like grep and awk.

Automate File Operations

Understand how to perform batch operations like renaming or deleting files based on specific file patterns or criteria, saving precious time and effort.



Made with Gamma

Instructions for Creating the Shell Script

1. Create a shell script that receives any number of file names as arguments and checks if each argument is a file or a directory.
 - The script should use conditional statements to determine if each argument is a file or a directory.
 - For each argument, it should display a message indicating whether it's a file, a directory, or neither.
 - Save the script to a file, for example, check_files.sh.
 - Make the script executable using chmod +x check_files.sh.

Task 1: Display a List of Files in the Current Directory

```
bash

#!/bin/bash

echo "List of files in the current directory:"
ls
```

Task 2: Check if Arguments are Files or Directories

```
#!/bin/bash

for arg in "$@"; do
    if [ -f "$arg" ]; then
        echo "$arg is a file."
    elif [ -d "$arg" ]; then
        echo "$arg is a directory."
    else
        echo "$arg is neither a file nor a directory."
    fi
done
```

Example Output of the Script

Example Usage:

```
./check_files.sh file1.txt directory2 file3.jpg
```

Example Output:

file1.txt is a file.

directory2 is a directory.

file3.jpg is a file.

Conclusion

Congratulations! You are now equipped with the knowledge to create a powerful shell script that lists all files in the current directory effortlessly. Enjoy exploring the endless possibilities of shell scripting!