

Linux & Git-GitHub Cheat-Sheet

Linux Commands

File & Directory Management

- `ls` – List directory contents.
- `cd [directory]` – Change the current directory.
- `pwd` – Print working directory.
- `mkdir [directory]` – Create a new directory.
- `rmdir [directory]` – Remove an empty directory.
- `rm [file]` – Remove a file.
- `cp [source] [destination]` – Copy files or directories.
- `mv [source] [destination]` – Move or rename files or directories.
- `touch [file]` – Create an empty file or update the timestamp of an existing file.

File Viewing & Editing

- `cat [file]` – Concatenate and display file content.
- `less [file]` – View file content page by page.
- `head [file]` – Display the first 10 lines of a file.
- `tail [file]` – Display the last 10 lines of a file.
- `nano [file]` – Edit a file using the Nano text editor.
- `vim [file]` – Edit a file using the Vim text editor.

Permissions & Ownership

- `chmod [permissions] [file]` – Change file permissions.
- `chown [owner]:[group] [file]` – Change file owner and group.
- `ls -l` – List files with detailed information including permissions and ownership.

System Monitoring & Management

- `top` – Display active processes.
- `ps aux` – List all running processes.
- `df -h` – Display disk space usage.
- `du -sh [directory]` – Display disk usage of a directory.
- `free -h` – Display memory usage.
- `uptime` – Show system uptime and load averages.

Git Commands

Basic Commands

- `git init` – Initialize a new Git repository.
- `git clone [url]` – Clone a repository from a URL.
- `git add [file]` – Add a file to the staging area.

- `git commit -m "[message]"` – Commit changes with a message.
- `git status` – Show the status of changes.
- `git log` – Display commit history.

Branching & Merging

- `git branch` – List all branches.
- `git branch [branch_name]` – Create a new branch.
- `git checkout [branch_name]` – Switch to a branch.
- `git merge [branch_name]` – Merge a branch into the current branch.
- `git rebase [branch_name]` – Rebase the current branch onto another branch.

Stashing

- `git stash` – Temporarily save changes without committing.
- `git stash pop` – Apply stashed changes and remove them from the stash.
- `git stash list` – List all stashed changes.
- `git stash drop` – Remove a specific stash.
- `git stash clear` – Remove all stashes.

Cherry-Picking

- `git cherry-pick [commit_hash]` – Apply a specific commit from another branch.

Conflict Resolution

- `git status` – Show files with conflicts.
- `git diff` – Show differences between conflicting versions.
- `git add [file]` – Stage resolved files for commit.

GitHub Tips

Repository Management

- **Create a Repository:** Click "New" on GitHub and follow the prompts.
- **Fork a Repository:** Click "Fork" on the top-right of the repository page.
- **Clone a Repository:** Use `git clone [url]` to copy a repository locally.

Collaboration

- **Pull Requests:** Create a pull request to propose changes. Review and discuss before merging.
- **Issues:** Use GitHub Issues to track bugs, tasks, and feature requests.
- **Wiki:** Utilize the GitHub Wiki for project documentation.

Branch Protection

- **Enable Branch Protection:** Prevent force-pushes and require pull request reviews before merging.

GitHub Actions

- **Set Up CI/CD:** Create workflows using `.github/workflows` to automate build, test, and deployment processes.