|  |  |
| --- | --- |
| Github for developers  \*Cheat sheet |  |
| |  |  | | --- | --- | | **$** | **prompt command / Terminal** | | **//** | **comment** | | **||** | **“or”** | | **&&** | **“and”** |   **Local Git Configuration**   * Checking your Git version  *$ git --version*  * Configuring your user name and email   *git config --global user.name "FIRST\_NAME LAST\_NAME"*  *git config --global user.email "MY\_NAME@example.com"*  Configuring autocrif  \* Convert LF endings into CRLF when you check out code:  *$ git config --global core.autocrlf true*  \* Convert CRLF to LF  $ git config --global core.autocrlf input  \* Windows-only project  *$ git config --global core.autocrlf false*  **Working Locally with Git**  Creating a Local copy of the repo  *$ git clone*  Our favorite Git command  *$ git status*  Using Branches locally  *$ git checkout -b new\_branch // Find variations of this command*  Switching branches  *$ git checkout new\_branch*  The Two Stage Commit  *$ git add . || git add file.js || git add -A*  *$ git commit -m “Add new feature called ‘x’” // variations of git commit*  **Collaborating on Your Code**  Pushing your changes to GitHub  *$ git push origin master || git push upstream master*  Updating Your Local Repository  *$ git pull origin master*  Cleaning Up the Unneeded Branches  *$ git branch -d feature/login*  *$ git push origin --delete feature/login*  **Viewing Local Project History**   * Review previous commits from current branch * *$ git log*   Streaming Your Workflow with Aliases. \*Rename commands at your preference [Link](https://dzone.com/articles/git-alias-make-your-git-experience-simpler-easier#:~:text=To%20add%20repo%20specific%20alias,That's%20it%20for%20now.)  *$ git config --global alias.st status // translates into*  *$ git st*  **Searching for Events in Your Code**  *$ git log*  *$ git bisect start*  *$ git bisect bad*  *$ git bisect good commit\_hash\_bc01*  **Understanding reset**  *$ git reset -- hard*  **Merge Strategies:**   * *(features\_branch) $ git rebase master*   *(master) $ git merge features\_branch* |  |
|  |  |