Version Control with Git (./): Git Cheatsheets for Quick Reference

Key Points

Automated Version Control (./01-basics/index.html)	• Version control is like an unlimited ‘undo’. • Version control also allows many people to work in parallel.
Setting Up Git (./02-setup/index.html)	• Use `git config` with the `--global` option to configure a user name, email address, editor, and other preferences once per machine.
Creating a Repository (./03-create/index.html)	• `git init` initializes a repository. • Git stores all of its repository data in the `.git` directory.
Tracking Changes (./04-changes/index.html)	• `git status` shows the status of a repository. • Files can be stored in a project’s working directory (which users see), the staging area (where the next commit is being built up) and the local repository (where commits are permanently recorded). • `git add` puts files in the staging area. • `git commit` saves the staged content as a new commit in the local repository. • Write a commit message that accurately describes your changes.
Exploring History (./05-history/index.html)	• `git diff` displays differences between commits. • `git checkout` recovers old versions of files.
Ignoring Things (./06-ignore/index.html)	• The `.gitignore` file tells Git what files to ignore.

Remotes in GitHub (./07-github/index.html)	• A local Git repository can be connected to one or more remote repositories. • Use the HTTPS protocol to connect to remote repositories until you have learned how to set up SSH. • `git push` copies changes from a local repository to a remote repository. • `git pull` copies changes from a remote repository to a local repository.
Collaborating (./08-collab/index.html)	• `git clone` copies a remote repository to create a local repository with a remote called `origin` automatically set up.
Conflicts (./09-conflict/index.html)	• Conflicts occur when two or more people change the same lines of the same file. • The version control system does not allow people to overwrite each other’s changes blindly, but highlights conflicts so that they can be resolved.
Open Science (./10-open/index.html)	• Open scientific work is more useful and more highly cited than closed.
Licensing (./11-licensing/index.html)	• People who incorporate General Public License (GPL’d) software into their own software must make their software also open under the GPL license; most other open licenses do not require this. • The Creative Commons family of licenses allow people to mix and match requirements and restrictions on attribution, creation of derivative works, further sharing, and commercialization. • People who are not lawyers should not try to write licenses from scratch.
Citation (./12-citation/index.html)	• Add a CITATION file to a repository to explain how you want your work cited.
Hosting (./13-hosting/index.html)	• Projects can be hosted on university servers, on personal domains, or on public forges. • Rules regarding intellectual property and storage of sensitive information apply no matter where code and data are hosted.
Supplemental: Using Git from RStudio (./14-supplemental-rstudio/index.html)	• Using RStudio’s Git integration allows you to version control a project over time.

Git Cheatsheets for Quick Reference

- Printable Git cheatsheets in several languages are available here (https://github.github.com/training-kit/) (English version (https://github.github.com/training-kit/downloads/github-git-cheat-sheet.pdf)). More material is available from the GitHub training website (http://try.github.io/).
- An interactive one-page visualisation (http://ndpsoftware.com/git-cheatsheet.html) about the relationships between workspace, staging area, local repository, upstream repository, and the commands associated with each (with explanations).
- Both resources are also available in other languages (e.g. Spanish, French, and more).
- “Happy Git and GitHub for the useR (http://happygitwithr.com)” is an accessible, free online book by Jenny Bryan on how to setup and use Git and GitHub with specific references on the integration of Git with RStudio and working with Git in R.
- Open Scientific Code using Git and GitHub (https://open-source-for-researchers.github.io/open-source-workshop/) - A collection of explanations and short practical exercises to help researchers learn more about version control and open source software.

Glossary

changeset
A group of changes to one or more files that are or will be added to a single commit in a version control repository.

commit
To record the current state of a set of files (a changeset) in a version control repository. As a noun, the result of committing, i.e. a recorded changeset in a repository. If a commit contains changes to multiple files, all of the changes are recorded together.

conflict
A change made by one user of a version control system that is incompatible with changes made by other users. Helping users resolve conflicts is one of version control’s major tasks.

HTTP
The Hypertext Transfer Protocol used for sharing web pages and other data on the World Wide Web.

merge
(a repository): To reconcile two sets of changes to a repository.

protocol
A set of rules that define how one computer communicates with another. Common protocols on the Internet include HTTP and SSH.

remote
(of a repository) A version control repository connected to another, in such way that both can be kept in sync exchanging commits.

repository
A storage area where a version control system stores the full history of commits of a project and information about who changed what, when.

resolve
To eliminate the conflicts between two or more incompatible changes to a file or set of files being managed by a version control system.

revision
A synonym for commit.

SHA-1

SHA-1 hashes (https://en.wikipedia.org/wiki/SHA-1) is what Git uses to compute identifiers, including for commits. To compute these, Git uses not only the actual change of a commit, but also its metadata (such as date, author, message), including the identifiers of all commits of preceding changes. This makes Git commit IDs virtually unique. I.e., the likelihood that two commits made independently, even of the same change, receive the same ID is exceedingly small.

SSH

The Secure Shell protocol used for secure communication between computers.

timestamp

A record of when a particular event occurred.

version control

A tool for managing changes to a set of files. Each set of changes creates a new commit of the files; the version control system allows users to recover old commits reliably, and helps manage conflicting changes made by different users.

Edit on GitHub (https://github.com/swcarpentry/git-novice/edit/gh-pages/reference.md) / Contributing (https://github.com/swcarpentry/git-novice/blob/gh-pages/CONTRIBUTING.md) / Source (https://github.com/swcarpentry/git-novice/) / Cite (https://github.com/swcarpentry/git-novice/blob/gh-pages/CITATION) / Contact (mailto:team@carpentries.org)

Using The Carpentries style (https://github.com/carpentries/styles/) version 9.5.3 (https://github.com/carpentries/styles/releases/tag/v9.5.3).