Basic Computing Tools

Group 22

April 11, 2016

Table of Contents

Table of Contents

Table of Contents

Introduction

LaTeXis a typesetting system that is very suitable for producing scientific and mathematical documents of high typographical quality. It is also suitable for producing all sorts of other documents, from simple letters to complete books. LaTeXenables authors to typeset and print their work at the highest typographical quality, using a predefined, professional layout. LaTeXwas originally written by Leslie Lamport.

commands

LaTeXcommands are case sensitive, and take one of the following two formats:

1. They start with a backslash \and then have a name consisting of letters only. Command names are terminated by a space, a number or any other 'non-letter.'
2. They consist of a backslash and exactly one non-letter.
3. Many commands exist in a 'starred variant' where a star is appended to the command name.

Table of Contents

Introduction

Gnuplot is a portable command-line driven graphing utility for Linux, OS/2, MS Windows, OSX, VMS, and many other platforms. The source code is copyrighted but freely distributed (i.e., you don't have to pay for it). It was originally created to allow scientists and students to visualize mathematical functions and data interactively, but has grown to support many non-interactive uses such as web scripting. It is also used as a plotting engine by third-party applications like Octave. Gnuplot has been supported and under active development since 1986.

Fuctions available

abs(x)	absolute value of x, $\|x\|$
acos(x)	arc-cosine of x
asin(x)	arc-sine of x
atan(x)	arc-tangent of x
cos(x)	cosine of x, x is in radians.
cosh(x)	hyperbolic cosine of x, x is in radians
erf(x)	error function of x
exp(x)	exponential function of x, base e
inverf(x)	inverse error function of x
invnorm(x)	inverse normal distribution of x
log(x)	log of x, base e
log10(x)	log of x, base 10
norm(x)	normal Gaussian distribution function
rand(x)	pseudo-random number generator
sgn(x)	1 if $x > 0$, -1 if $x < 0$, 0 if $x = 0$
sin(x)	sine of x, x is in radians

Table of Contents

Table of Contents

Table of Contents

Introduction

A **version control system (VCS)** allows you to track the history of a collection of files. It supports creating different versions of this collection. Each version captures a snapshot of the files at a certain point in time and the VCS allows you to switch between these versions. These versions are stored in a specific place, typically called a repository. **Git** is a widely used source code management system for software development. It is a **distributed revision control system** with an emphasis on speed, data integrity, and support for distributed, non-linear workflows. In a distributed version control system each user has a complete local copy of a repository on his individual computer. The user can copy an existing repository. This copying process is typically called cloning and the resulting repository can be referred to as a clone.

Git terminology

- **Cloning** : The process of copying an existing Git repository is called cloning. After cloning a repository the user has the complete repository with its history on his local machine.
- **Working Tree** : A local repository provides at least one collection of files which originate from a certain version of the repository. This collection of files is called the working tree.
- **Branching** : Git supports branching which means that you can work on different versions of your collection of files. A branch separates these different versions and allows the user to switch between these versions to work on them.
- **Repository** : A repository contains the history, the different versions over time and all different branches and tags. In Git each copy of the repository is a complete repository.

Setting up a Repository

The git init command creates a new Git repository. It can be used to convert an existing, unversioned project to a Git repository or initialize a new empty repository. Executing git init creates a .git subdirectory in the project root, which contains all of the necessary metadata for the repo.

git init Transform the current directory into a Git repository. This adds a .git folder to the current directory and makes it possible to start recording revisions of the project.

The git clone command copies an existing Git repository.

git clone "repo" Clone the repository located at ¡repo¿ onto the local machine. The original repository can be located on the local filesystem or on a remote machine accessible via HTTP or SSH.

Saving changes

The git add command adds a change in the working directory to the staging area. It tells Git that you want to include updates to a particular file in the next commit.

git add "file" Stage all changes in "file" for the next commit.

The git commit command commits the staged snapshot to the project history. Committed snapshots can be thought of as "safe" versions of a project—Git will never change them unless you explicity ask it to.

git commit Commit the staged snapshot. This will launch a text editor prompting you for a commit message. After you've entered a message, save the file and close the editor to create the actual commit.

git commit -m "message" Commit the staged snapshot, but instead of launching a text editor, use "message" as the commit message.

Inspecting a repository

The git status command displays the state of the working directory and the staging area. It lets you see which changes have been staged, which haven't, and which files aren't being tracked by Git. Status output does not show you any information regarding the committed project history.
git status List which files are staged, unstaged, and untracked.

Undoing Changes

The git revert command undoes a committed snapshot. But, instead of removing the commit from the project history, it figures out how to undo the changes introduced by the commit and appends a new commit with the resulting content. This prevents Git from losing history.

git revert "commit" Generate a new commit that undoes all of the changes introduced in "commit", then apply it to the current branch.

Git reset can be used to remove committed snapshots, although it's more often used to undo changes in the staging area and the working directory. In either case, it should only be used to undo local changes—you should never reset snapshots that have been shared with other developers.

git reset "file" Remove the specified file from the staging area, but leave the working directory unchanged. This unstages a file without overwriting any changes.

Undoing Changes

git reset Reset the staging area to match the most recent commit, but leave the working directory unchanged.
The git clean command removes untracked files from your working directory. This is really more of a convenience command, since it's trivial to see which files are untracked with git status and remove them manually. Like an ordinary rm command, git clean is not undoable, so make sure you really want to delete the untracked files before you run it.
git clean -n Perform a "dry run" of git clean. This will show you which files are going to be removed without actually doing it.
git clean -f Remove untracked files from the current directory. The -f (force) flag is required unless the clean.requireForce configuration option is set to false (it's true by default).

Table of Contents