

# GithubIntroduction

September 18, 2016

## 1 A Very Quick Introduction to Git/Github for Julia Users

Julia's package system and Github are very closely intertwined:

- Julia's package management system (METADATA) is a Github repository
- The packages are hosted as Github repositories
- Julia packages are normally referred to with the ending ".jl"
- Repositories register to become part of the central package management by sending a pull request to METADATA.jl
- The packages can be found / investigated at Github.com
- Julia's error messages are hyperlinks to the page in Github

Because of this, it's very useful for everyone using Julia to know a little bit about Git/Github.

### 1.1 Git Basics

- Git is a common Version Control System (VCS)
- A project is a **repository** (repos)
- After one makes changes to a project, **commit** the changes
- Changes are **pulled** to the main repository hosted online
- To download the code, you **clone** the repository
- Instead of editing the main repository, one edits a **branch**
- To get the changes of the main branch in yours, you **fetch**
- One asks the owner of the repository to add their changes via a **pull request**
- Stable versions are cut to **releases**
- The major online server for git repositories is Github
- Github is a free service
- Anyone can get a Github account
- The code is hosted online, free for everyone to view
- Users can open **Issues** to ask for features and give bug reports to developers
- Many projects are brought together into **organizations** (JuliaMath, JuliaDiffEq, JuliaStats, etc.)

An example Github repository for a Julia package is is DifferentialEquations.jl:  
<https://github.com/JuliaDiffEq/DifferentialEquations.jl>

## **1.2 Using the Package Manager**

To