Advanced Data Programming

Workshop:
Introduction to GitHub and GitHub Pages

Karl Ho
School of Economic, Political and Policy Sciences
University of Texas at Dallas

Workshop: Github and GitHub Pages

1. What is Version Control System (VCS)?
2. GitHub repository system
3. GitHub Pages as website hosting
4. GitHub and Rstudio cloud

What is Git ?

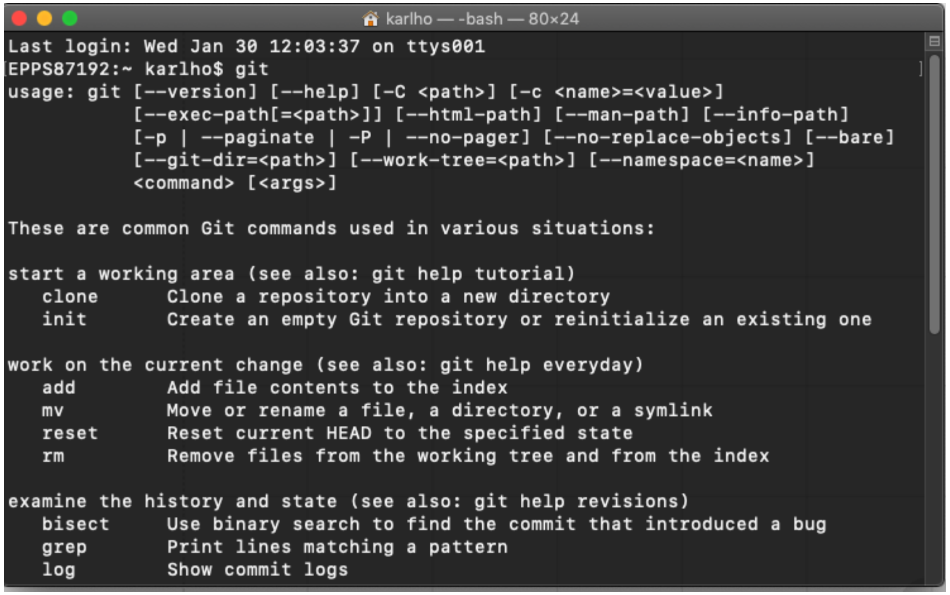
- Git is a version control system for tracking changes in files and coordinating work among multiple parties.
- It is primarily used for source code management in software development, but it can be used to keep track of changes in any set of files.

What is Git ?

- Git is a distributed version control system for tracking changes in files and coordinating work among multiple parties.
- It is primarily used for source code management in software development, but it can be used to keep track of changes in any set of files.

What is Git ?

- Git is primarily command-line based.



```
Last login: Wed Jan 30 12:03:37 on ttys001
EPPS87192:~ karlho$ git
usage: git [--version] [--help] [-C <path>] [-c <name>=<value>]
           [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
           [-p | --paginate | -P | --no-pager] [--no-replace-objects] [--bare]
           [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
           <command> [<args>]

These are common Git commands used in various situations:

start a working area (see also: git help tutorial)
   clone      Clone a repository into a new directory
   init       Create an empty Git repository or reinitialize an existing one

work on the current change (see also: git help everyday)
   add        Add file contents to the index
   mv         Move or rename a file, a directory, or a symlink
   reset      Reset current HEAD to the specified state
   rm         Remove files from the working tree and from the index

examine the history and state (see also: git help revisions)
   bisect     Use binary search to find the commit that introduced a bug
   grep       Print lines matching a pattern
   log        Show commit logs
```

What is GitHub ?

- GitHub is a web-based hosting service for version control using Git but with additional features including access control and several collaboration features such as bug tracking, feature requests, task management, and wikis for every project.

What is GitHub ?

- GitHub is now owned by Microsoft and is open and free for individual use.
- It will become one of the world's largest repository systems for hosting and social coding

1. Terminology

1. **Repository**
2. The *repository* is where files' current and historical data are stored, often on a server.

1. **Version control system (VCS)**
2. A management system of changes to documents, usually scripts or computer programs
3. e.g. Subversion (SVN), git

1. **Branch**
2. A set of files under version control may be *branched* or *forked* at a point in time so that, from that time forward, two copies of those files may develop at different speeds or in different ways independently of each other.

1. Terminology (continued)

1. **Commit**
2. To *commit* is to proceed or merge the changes made in the working copy to the repository. The terms 'commit' and 'checkin' can also be used as nouns to describe the new revision that is created as a result of committing.

1. **Initialize**
2. to create a new, empty repository.

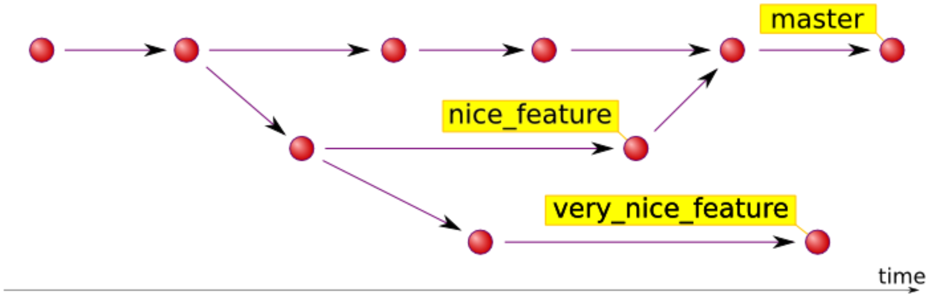
1. **Pull, push**
2. Copy revisions from one repository into another. *Pull (or Fetch)* is initiated by the receiving repository, while *push* is initiated by the source.

1. Terminology (continued)

1. **Clone**
2. *Cloning* means creating a repository containing the revisions from another repository. This is equivalent to *push*ing or *pull*ing into an empty (newly initialized) repository. As a noun, two repositories can be said to be *clone*s if they are kept synchronized, and contain the same revisions.

What is GitHub ?

1. Git works on branches, which represent independent lines of development, as each snapshot is linked to a 'parent' one it built upon. By default, everyone's repositories are on a "master" branch.



1. source: http://hades.github.io/2010/01/git-your-friend-not-foe-vol-2-branches/

Why GitHub?

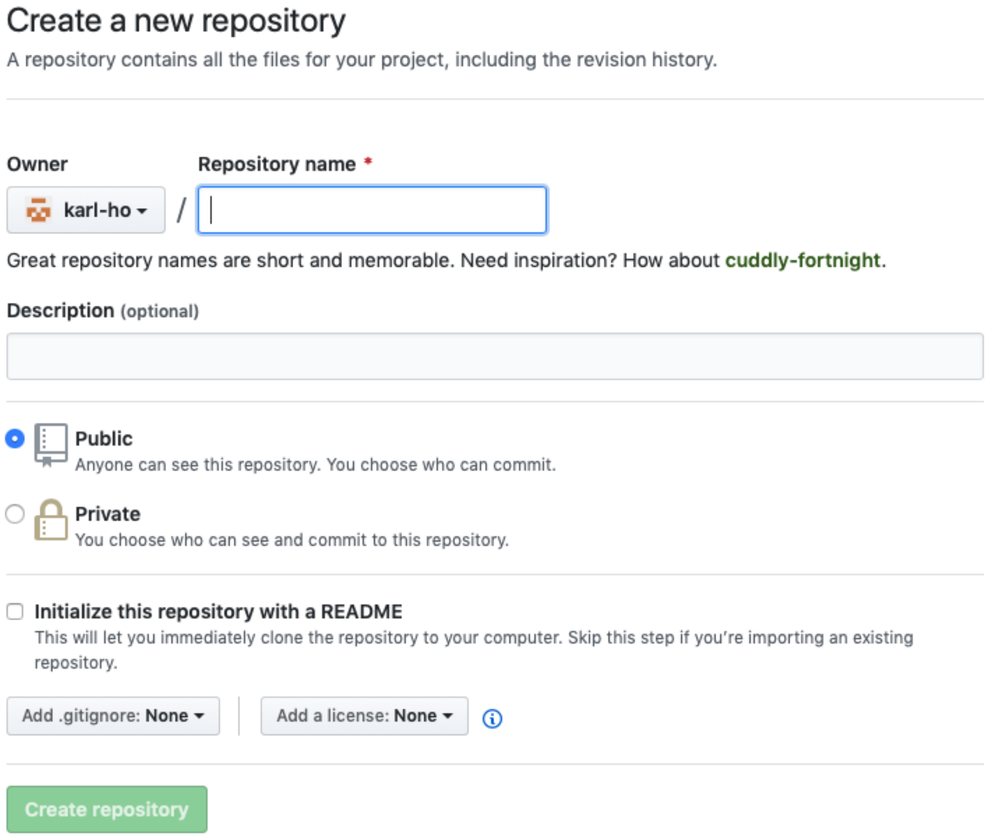
1. GitHub is an easy way to collaborate with others on shared Git repositories.
2. GitHub hosts programs, data files and data products and even websites.
3. Users can sync local files with changes made and pushed to the shared one.
4. It is designed for "social coding" and collaboration even from strangers.

Create a GitHub account

1. Login to github.com
2. Pick a username that you will use for login and future public profile. If your name is available, add a number or a name that identifies you. Avoid funny or whimsical terms since this is your professional profile.

Create a repository

1. Once login, you may create a repository like folder or directory.
2.



Create a new repository

A repository contains all the files for your project, including the revision history.

Owner: karl-ho / Repository name *

Great repository names are short and memorable. Need inspiration? How about cuddly-fortnight.

Description (optional)

● Public
Anyone can see this repository. You choose who can commit.

○ Private
You choose who can see and commit to this repository.

□ Initialize this repository with a README
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: None | Add a license: None

Create a repository

1. You may initialize the repository with a Readme.md file
2. md stands for Markdown, which is a new web language that makes documentation online much easier.
3. You may edit your Readme.md on Github and leave descriptions using the markdown language.
4. Try the following (order and unordered list, with italics):
5. 1. Introduction

   2. Data Theory

   3. Messages in Data

      1. Data, *Messenger* and Message.

1. Headers:
2.
   ```
   # H1
   ## H2
   ### H3
   #### H4
   ##### H5
   ###### H6
   ```

   Emphasis (Italics)

Github Desktop

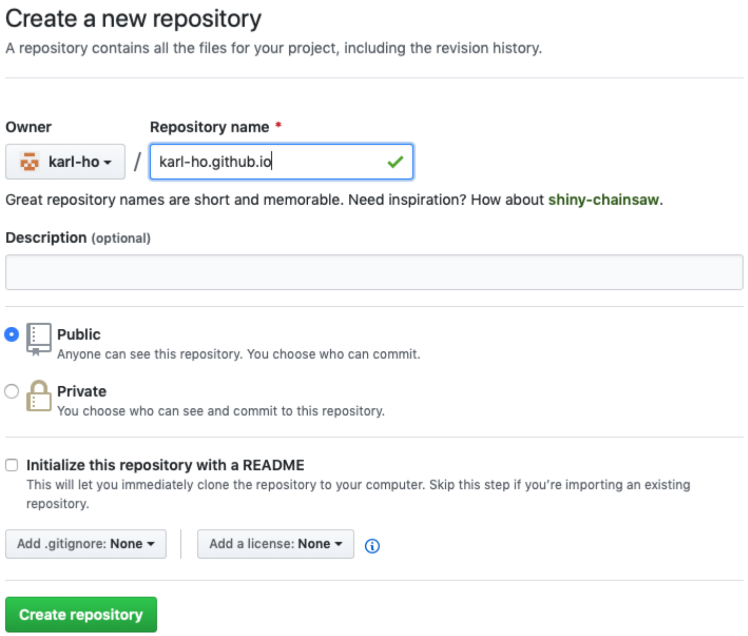
1. Github Desktop is a local app for synchronizing files to Github website.
2. For now, using the web version of Github is recommended.

GitHub website

1. GitHub hosts website by project or account names with the domain github.io (e.g. karl-ho.github.io)
2. That's why your account name is important!

Create GitHub website

1. Create GitHub repository to host you web content
2. create karl-ho.github.io
3.



Create a new repository

A repository contains all the files for your project, including the revision history.

Owner: karl-ho / Repository name *: karl-ho.github.io ✓

Great repository names are short and memorable. Need inspiration? How about shiny-chainsaw.

Description (optional)

● Public
Anyone can see this repository. You choose who can commit.

○ Private
You choose who can see and commit to this repository.

□ Initialize this repository with a README
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: None

Add a license: None

Create GitHub website

1. Once repository is created, create a new file called index.html
2. Click commit new file

3. Type "Your name Github website"
4. Try some HTML script?
    1. HTML:

    2.
       ```
       <h1>This is a Heading</h1>
       <p>This is a paragraph.</p>
       ```

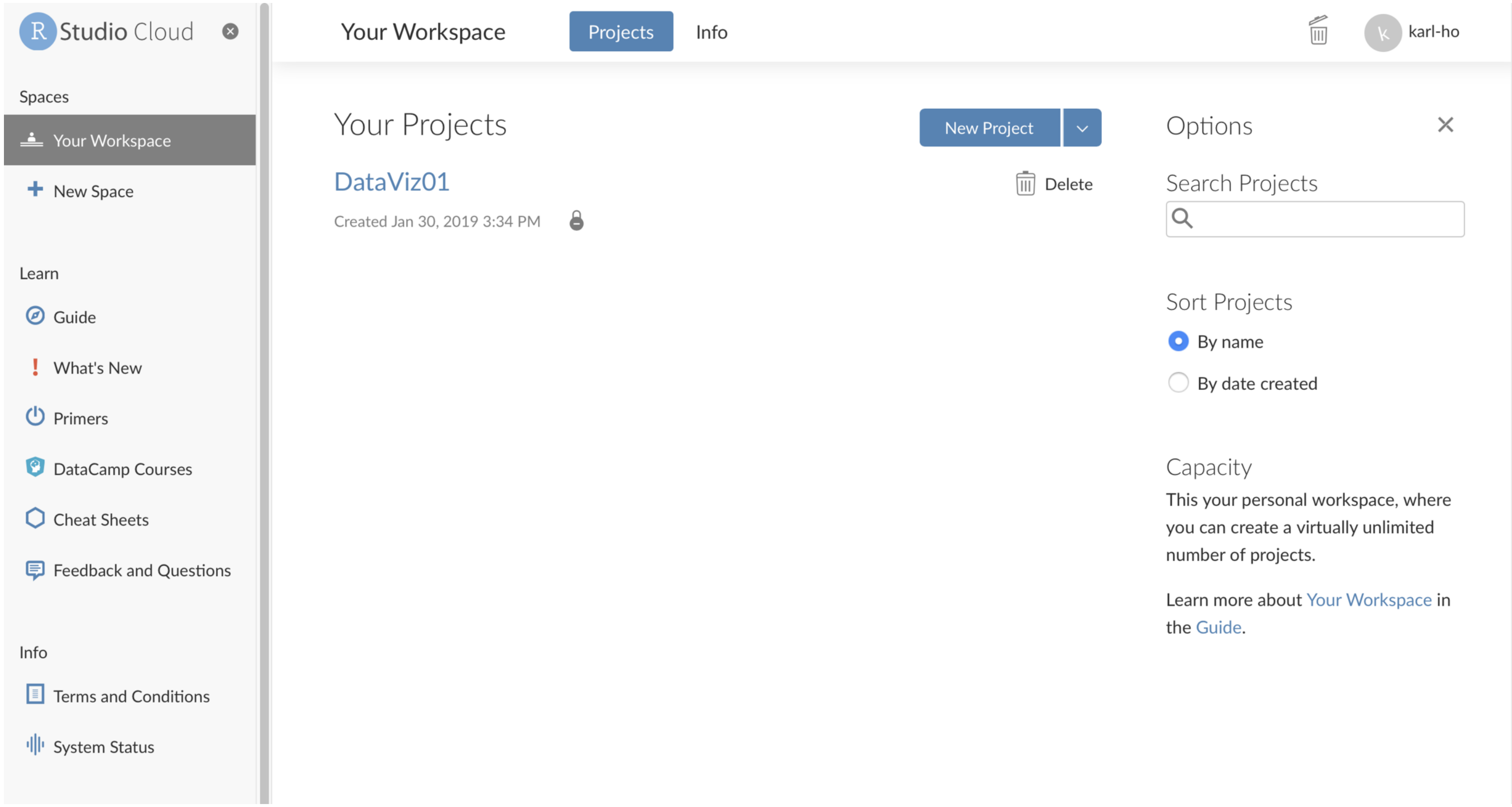
    3.
       ```
       <center>Center this line</center>
       ```

GitHub page: blog

1. Jekyll/Jekyll now
2. Hugo
3. Pelican (Python based)

GitHub and RStudio cloud

Github works closely with Rstudio, allowing to login to Rstudio cloud using GitHub account and run R program on cloud.



Created Jan 30, 2019 3:34 PM

Sort Projects

● By name

○ By date created

Capacity

This your personal workspace, where you can create a virtually unlimited number of projects.

Learn more about Your Workspace in the Guide.

Reference

1. Bitbucket Tutorial: Git Branch
2. https://www.atlassian.com/git/tutorials/using-branches
3. The House of Hades: Git Is Your Friend not a Foe v2: Branches
4. http://hades.github.io/2010/01/git-your-friend-not-foe-vol-2-branches/