Git and GitHub
Begginer Guide

ADAS

Overview

1. Install git and create GitHub account.

2. What is git?

3. How does git works?

4. What is GitHub?

5. Quick example using git and GitHub.
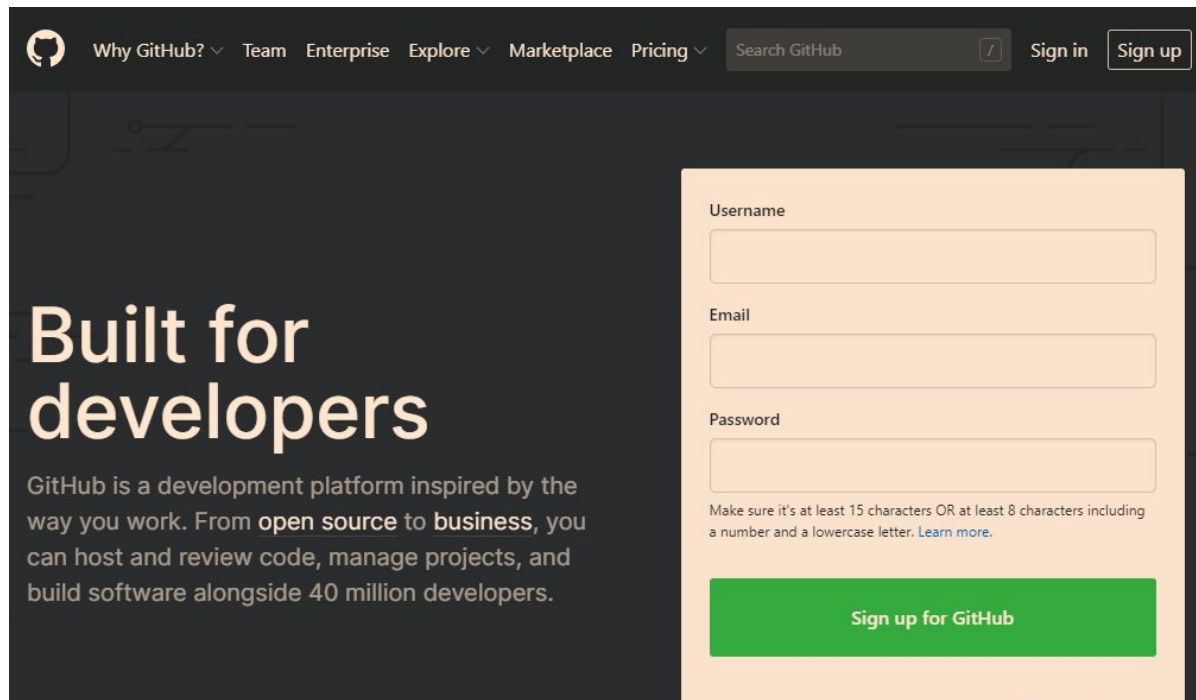
Install Git and create a GitHub Account.

Install git

› Linux (Debian)

› Command: `sudo apt-get install git`

› Linux (Fedora)

› Command: `sudo yum install git`

› Mac

› https://git-scm.com/download/mac

› Windows

› https//git-scm.com/download/win

Create GitHub account

› www.github.com

› **Free** for public repositories

What is a version control?

› A system that keeps records of your changes.

› Allows for collaborative development.

› Allows you to know who made what changes and when.

› **<u>Allows you to revert any changes and go back to previous state.</u>**

What is git?

What is version control?

› Distributed version control

› User keep entire code and history on their location machines

› User can make any changes ithout internet access

› (Except puhing and pulling changes from a remote server)

What is git?

› Stated in 2005

› Created by Linus Torvald to aid in Linux kernel development

What is git?

> Git isn't the only version control system.

> But it's the most convenient to current projects.

How does git work?

› Could be complicated at first but there are main concepts to help us to undestand git process.

› Important git terminology in following slides are in blue.

Key concepts: Snapshots

› The way git keeps track of your code history.

› Essentially records what all your files look like at the given point of time.

› You decide when you take a snapshot, and of what files.

› Have the ability to back to visit any snapshot

    › Your snapshots from later on Will stay around, too

Key concepts: Commit

› The act of creating and snapshot.

› Can be a noun or verb

› "I commited code"

› "I jus made a new commit"

› Esscentially, a full project is made up of a bunch of commits.

Key concepts: Commit

› Commits contain three pieces of information:

1. Information about how the files changed from previously.

2. A reference to the commit that came before it:

   › "Called parent commit"

3. A hash code name:

   › Will look something like:

TASK: Trailer Line Response KPI

Trailer Line Response KPI implementation. Measure the distance
of vehicle travels for returning the Trailer_Angle from Actual_Max
to 1 deg when knob is changed from 100% to 0% as step fucntion.

Resolves: ADCCR-4818

Signed-off-by: uic26460 <julio.martinez.ramirez@continental-corporation.com>

feature/ADCCR-4818/Trailer_Line_Response_KPI (#170)

uic26460 committed 4 days ago   commit 21ea84ca33b6916

Key concepts: Repositories

› Often shortened to "repo"

› A collection of all the files and the history of those files

   › Consist of all your commits

   › Place where all your hard work is stored

› Can live on local machine or on a remote server (GitHub)

› The act of copying a repository from a remote server is called cloning

› Cloning from a remote server allows team to work together

Key concepts: Repositories

› The process to downloading commits that don't exist on your local machine from a remote server repository is called pulling changes.

› The process of adding your local changes to the remote repository is called pushig changes.

Key concepts: Branches

› All commits in git live n some branch

› But there can be many, many branches

› The main branch in a project is called the master branch

So what does a typical project look like?

› A bunch of commits linked togueter that live in some branch, contained in a repository.

› Following images taken and modified from:

  › https://marklodato.github.io/visual-git-guide/index-en.html

  › Also a good tutorial!
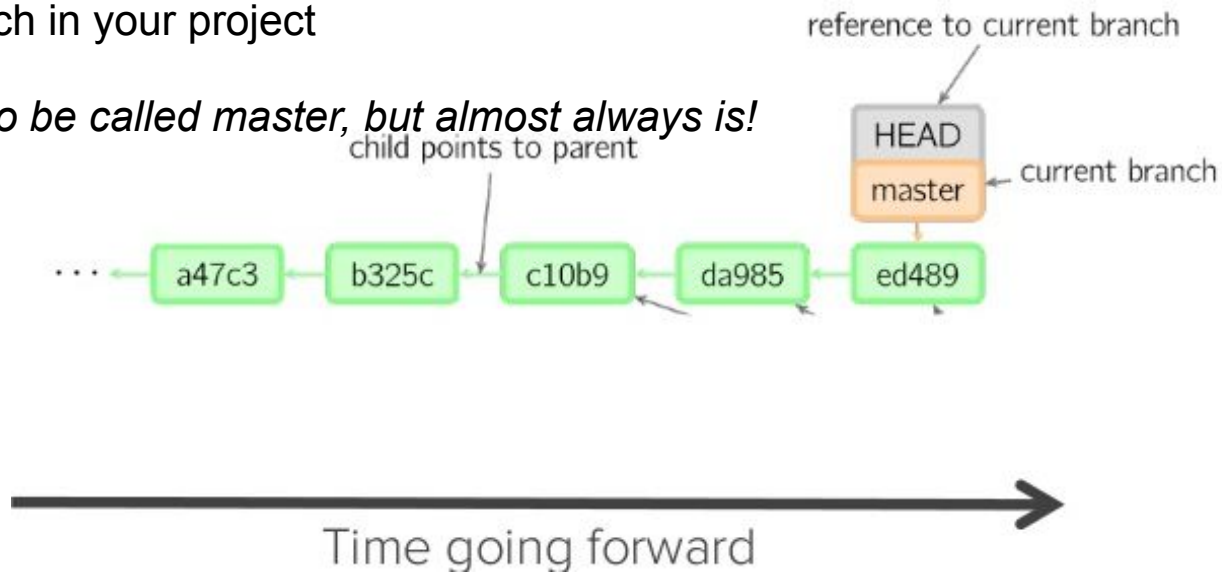
So, what does a tipical project look like?
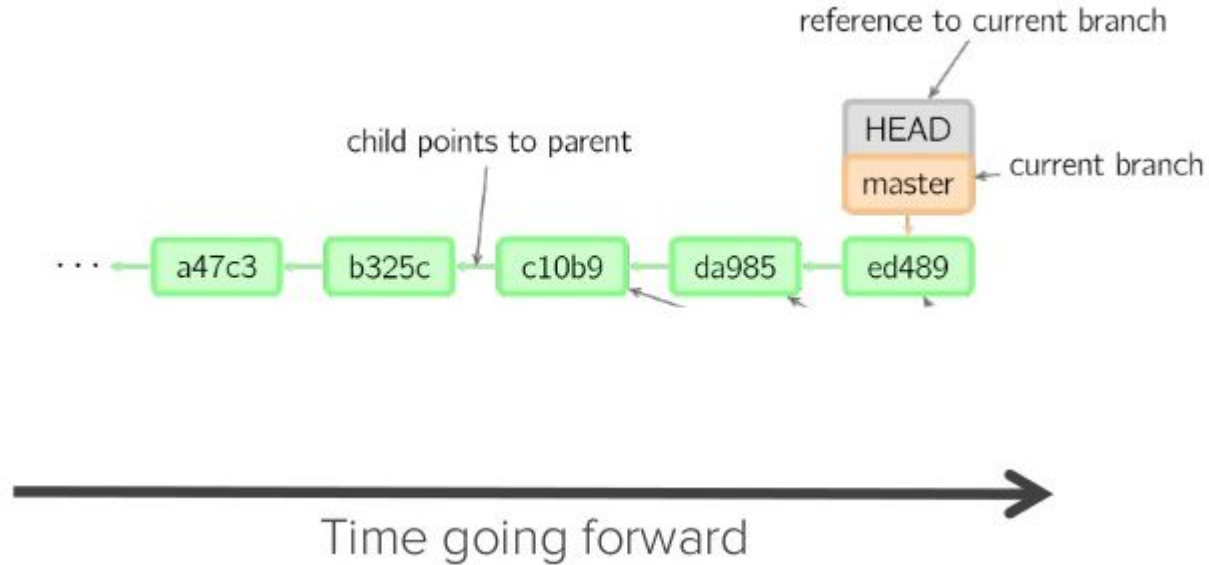
So, what is HEAD?

› A reference to most recent commit
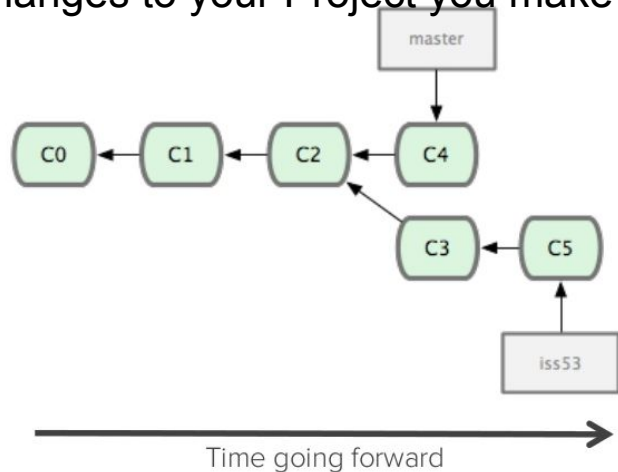
› *(In most cases- not always truth)*
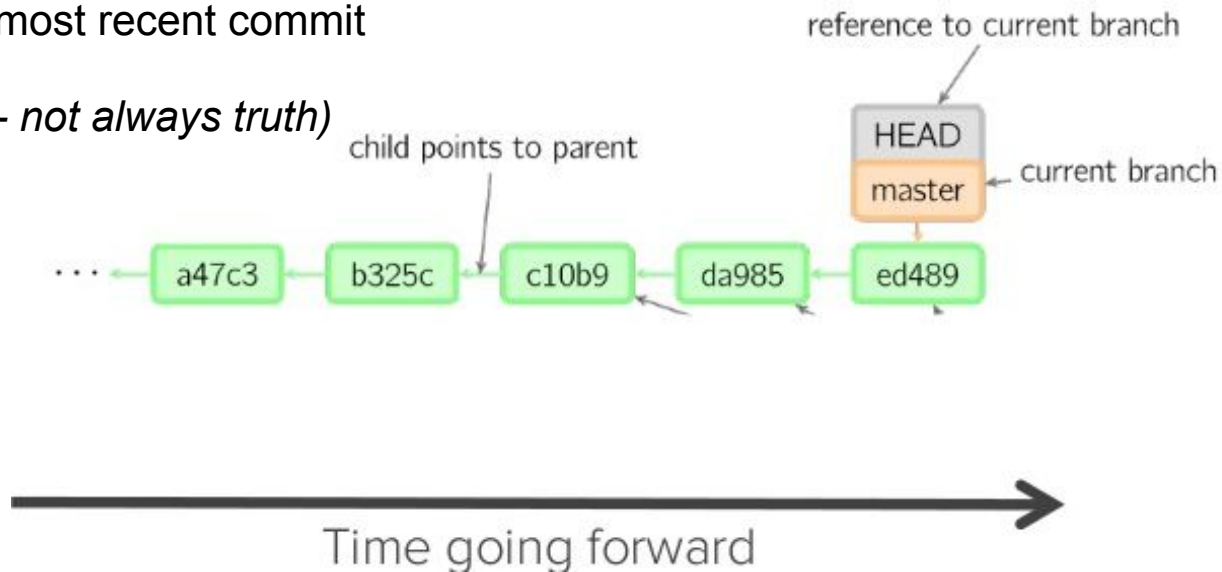
So what's the MASTER?

› The main branch in your project

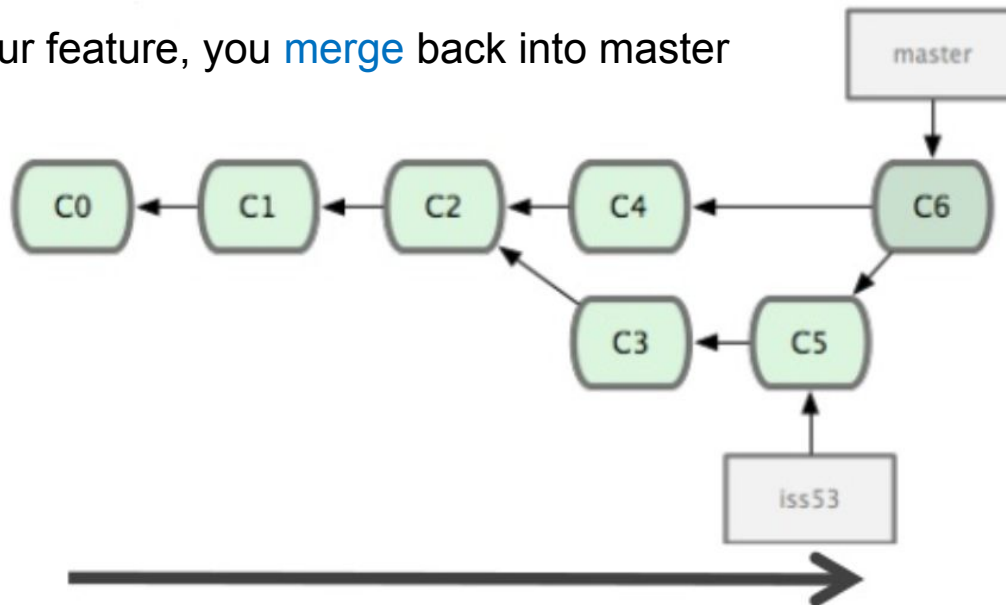› *Doesn't **have** to be called master, but almost always is!*

Key concepts: Branching off the master branch

› The start of a branch points to a specific commit

› Whn you wnat to make any changes to your Project you make a new branch based on commit

Time going forward

Key concepts: Merging

› Once you are done with your feature, you merge back into master

Key concepts: How do you make a **commit** anyway?

› There are a lot of 'states' and 'places' a file can be.

› Local on your computer: the "working directory"

› When a file is ready to be put in a commit you add it onto the "index" or "staging"

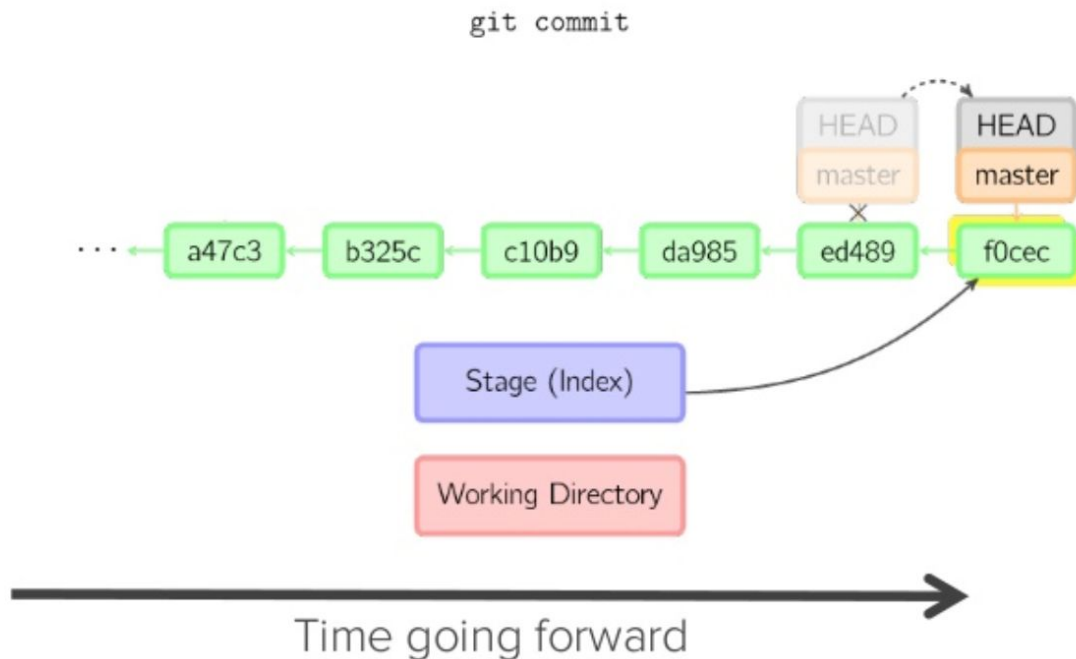 › Staging is the new preferred term but you can see both "index" and "staging" being used

Key concepts: How do you make a **commit** anyway?

› The process:

    › Make some changes to a file

    › Use the "git add" command to put the fileonto the staging enviroment

    › Use "git commit" command to create a new commit

Key concepts: How do you make a commit anyway?
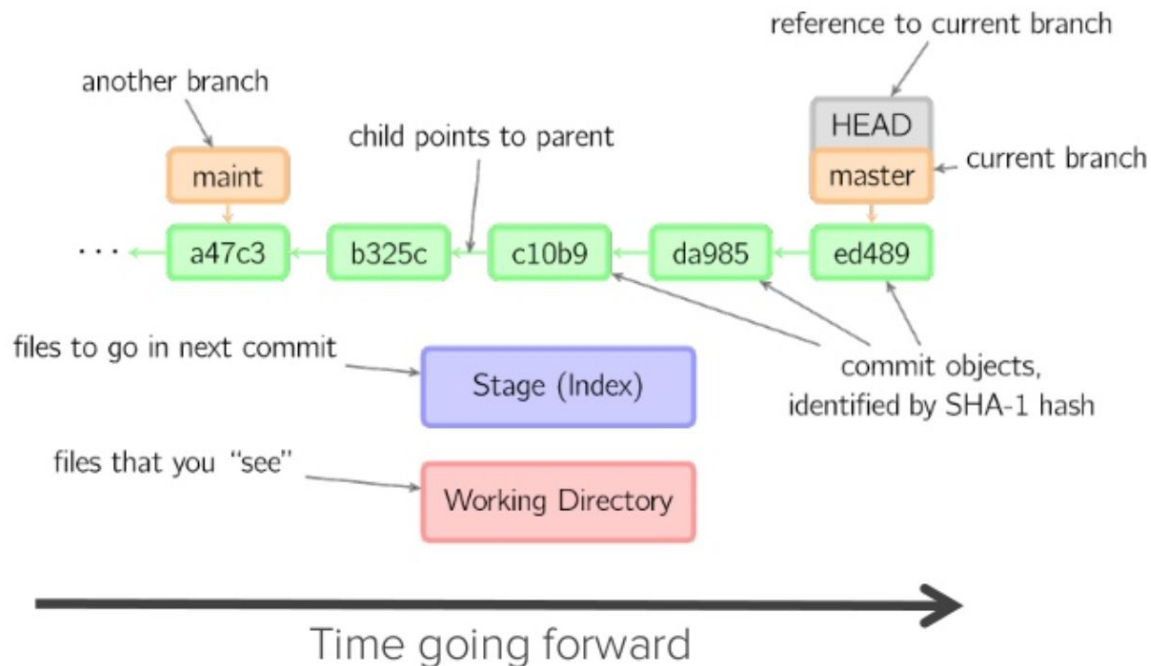
Key concepts: How do you make a commit anyway?

What is GitHub?

› www.github.com

› Largest web-based git repository hosting a service

    › Host 'remote repositories'

› Allows for code collaboration with anyone online

› Add extra functionality on top of git

    › UI, documentation, bug tracking, feature request, pull request and more!

› Founded in 2008

Tips and tricks! …and chips

$ git push -u origin feature/ADCCR-2955/VStop_review_test

When your coworker asks you which git branch you're currently working on

› Always work in feature branch/issues

› **$ git checkout -b feature/ADCCR-2955/VStop_review_test**

$ git commit -s

```
$ git log -1
commit 7b0b8d62239a021r8cbdc6581cbz3002e7e97cf6
Author: Mustermann Max (uid12345) <max.mustermann@continental-corporation.com>
Date:    Thu Jan 21 09:32:39 2019 +0100

    TASK: implement a new functionality

    Lorem ipsum dolor sit amet, consetetur sadipscing elitr,
    sed diam nonumy eirmod tempor invidunt ut labore et dol
    ore magna aliquyam erat, sed diam voluptua. At vero eos
    et accusam et justo duo dolores et ea rebum. Stet clita
    kasd gubergren, no sea takimata sanctus est Lorem ipsum
    dolor sit amet.

    Resolves: ABC-1234
```

Types of Headache

Migraine · Hypertension · Stress · GIT ERRORS

Never stop learning!