Guide to Git and GitHub

by
Alan Labouseur
based on
Designate a *working directory* on your local disk for Git to monitor. We’ll use D:\alan\proj1 in this example. This is sometimes called *creating a repository.* Give the repository a name (*GitHubTest* in this example.)
Create your project files (A, B, and C in this example) in this *working directory*.
Add, stage, and commit the project files. This is sometimes called a check-in. Be sure to write interesting, helpful, and complete commit comments.
This is “Version 1” of our project, consisting of three (3) files: A, B, and C.
As we continue working on our project, we change some files. Let’s say we change files A and C.
Check-in again. *Stage* and *commit* the changed project files.
Be sure to continue to write interesting, helpful, and complete commit comments.
This is “Version 2” of our project, still consisting of three (3) files — A, B, and C — but with new versions of only files A and C (because we did not change B).
“Version 3” results from our changing file C and committing.
“Version 4” results from our changing files A and B and committing.
“Version 5” results from our changing files B and C and committing.
You can see the pattern; you get the idea.
All of this is happening thanks to Git watching your local working directory.
All of this is happening thanks to Git watching your **local working directory**. We need to get this online and accessible from the web.
That’s where GitHub comes in.
We publish from our **local** Git repository to our **global** (online, in the cloud, Internet-based) GitHub repository so that our code is accessible from the web.
We publish from our local Git repository to our global (online, in the cloud, Internet-based) GitHub repository, which is accessible from the web.

This is read me. So read. Reading is fundamental.

Wait! Here is more to read.
Overview and Workflow

GitHubTest

D:\alan\proj1

Diagram showing a process with labeled steps: A *, B, C *