git 101 What is git version control

If you have been using StarTeam, you have been using a **centralized** version control system, where all revision information is managed in a database that exists on a single server. You checkout the source files from the server database onto your local system to work with them. You then make modifications to specific files and check the changes in. The server keeps track of the history of each file independently.

git is a **distributed** version control system which works very differently from a centralized version control system. In our GitHub distributed version control system the GitHub server maintains a public {aka remote} copy of lots of independent version control databases. You **copy** {aka. clone} the complete version control databases you are interested in onto your local developer system. The local version control database resides in the same folder as the source files that exist in it. When you make changes to files in the local workspace you have the ability to [stage](http://programmers.stackexchange.com/questions/119782/what-does-stage-mean-in-git) the file changes, and then commit those changes as a **change set** into the local version control database. After you have the change set(s) in your local copy of the database, you sync it with the public database. So, git keeps track of change sets, it does not track the history of each individual file.

GitHub provides a 50 minute Basics of git and GitHub in a [WebCast](http://www.youtube.com/watch?v=pIGYELKPeog).

After watching the video, [setup git on your local system](https://devtopia.esri.com/devtopia/git-management/wiki/devtopia-101-git-and-User-and-Setup), and then you are ready to head over to the [git class](https://devtopia.esri.com/devtopia/git-class/wiki/git-class-for-working-in-devtopia)

**An important note on the WebCast video:** The video's point of view is maintaining an open source project on GitHub.com where the normal mode of making changes to public repos is to Fork the repo and then Clone it. On devtopia all team members will have **push** access to the repos in the Organizations where they participate. Since you have push access to your teams repos, the code updates within a development team will be done via [pull requests from branches within the teams repo](https://devtopia.esri.com/devtopia/git-management/wiki/devtopia-101-Only-fork-if-you-don%27t-have-push-access-to-a-repo).