**Part 3**

**What is GitHub? When was it created? Why? By who? What similar platforms exist? Why would you use such a platform?**

**GitHub**:

GitHub is the internet hosting service of your git. Git is an open source version control system which is invented by Linus Torvalds . When you are working in a team it helps in the coordination of work which can keep track of changes in files. GitHub is collection of all these files. We have repositories in GitHub which helps in maintance of projects. Each repository contains files related to a single project and we can have multiple collaborators for a repository.

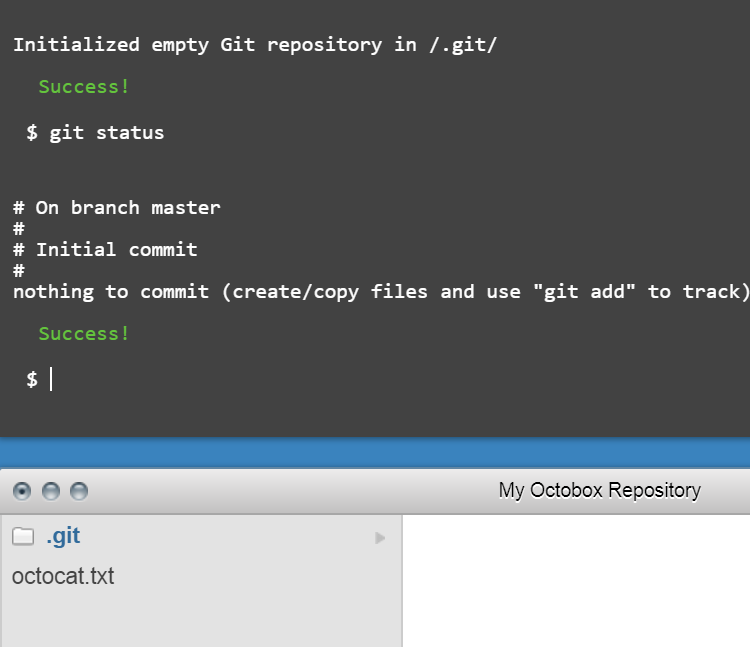
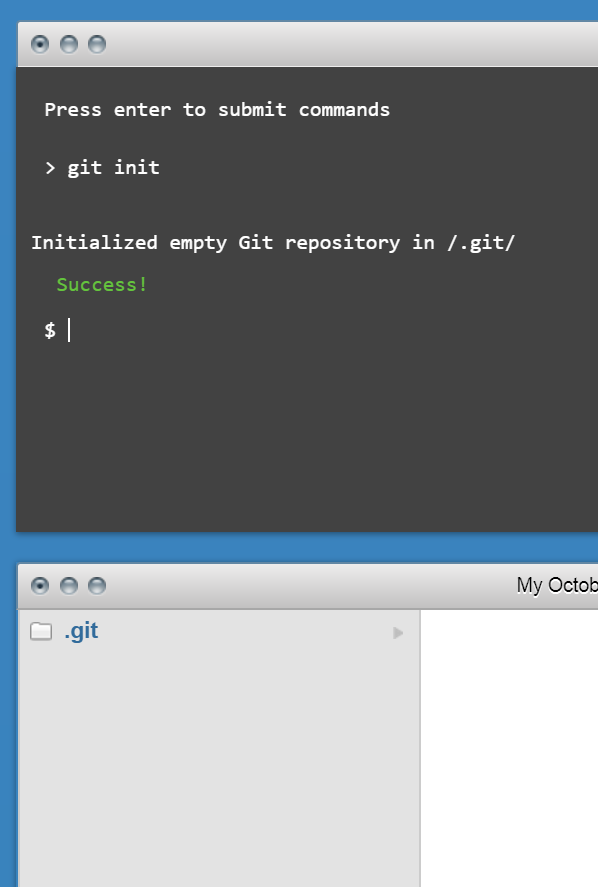
GitHub is created in 2008 by Tom Preston w erner who is software developer, inventor and entrepreneur. GitHub is now the largest online storage space of collaborative works that exists in the world. Whether you’re interested in participating in this global mind meld or in researching this massive file dump of human knowledge, you need to be here. Before GitHub existed, major companies created their knowledge mainly in private. But when you access their GitHub accounts, you’re free to download, study, and build upon anything they add to the network. Thank famed software developer [Linus Torvalds](http://en.wikipedia.org/wiki/Linus_Torvalds) for Git, the software that runs at the heart of GitHub. The problem with Git is that it’s so ancient that we have to use the command line—or Terminal if you’re a Mac user—in order to access it, typing in snippets of code like ‘90s hackers. This can be a difficult proposition for modern computer users. That’s where GitHub comes in.

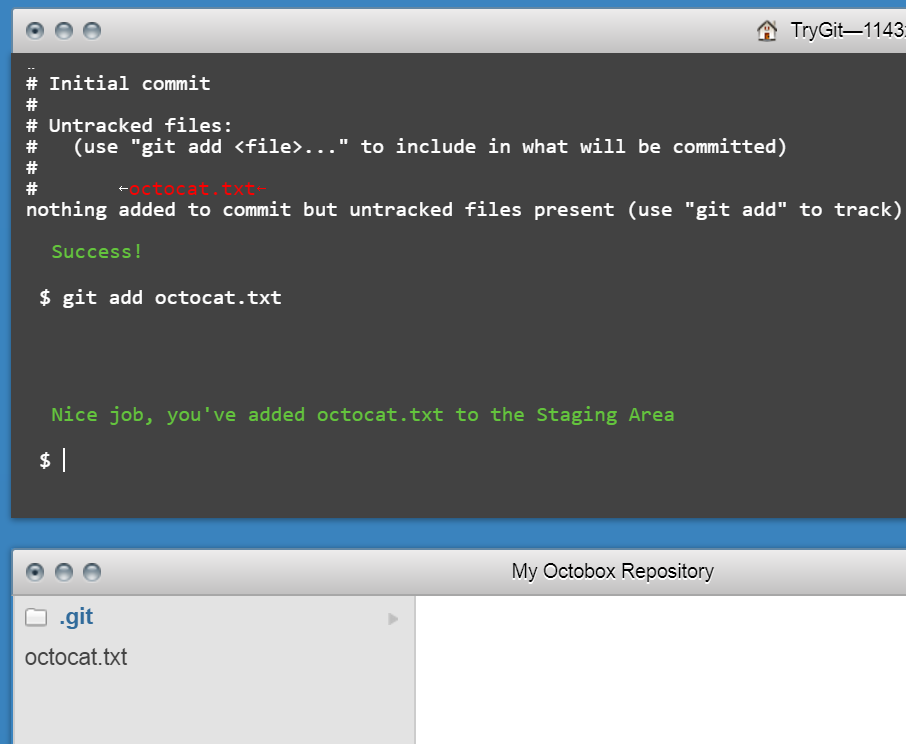
Bitbucket, SourceForge, GitLab, Kiln, Codeplane, CodePlex, Beanstalk are some other source code hosting websites (other similar platform) exist.

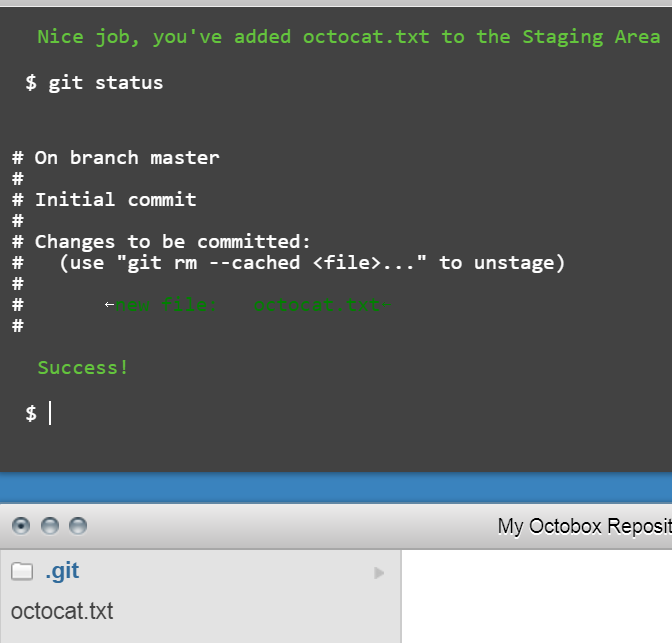
GitHub makes Git easier to use in two ways. First, if you [download the GitHub software](http://github.com/) to your computer, it provides a visual interface to help you manage your version-controlled projects locally. Second, creating an account on GitHub.com brings your version-controlled projects to the Web, and ties in social network features for good measure.

**Part 4:**

**Go through the Git tutorial here:** [**https://try.github.io**](https://try.github.io)**. While doing the tutorial, save your work the *LastnameFirstnameGitTutorial-mm-dd-yyyy.docx* file.**







**Part 5:**

**Define the following terms in the context of Git (2 lines maximum):**

* **Repository:** Basically repositories are nothing but folders which contains project related documents. We can maintain one single project in one repository. Repositories have multiple collaborators which can be public or private.
* **Commit:** Commit is something that an individual change to a file. It will generate a hash value for each and every commit which tells us about what changes were made to the file and who made the changes and when they were made.
* **Push:** Push is used when you want to send the changes committed locally into a remote repository which may be hosted by GitHub.
* **Branch:** It’s kind of other copy of your repository. In the branch version you can make changes whatever you want freely which will not make any damage to the main version. After committed changes whenever you want to copy it to the main you can just merge the branch.
* **Fork:** When you want to contribute or help others you can fork that repository where you can work freely and when you want to propose the changes to make to the repository owner you can pull request and let them know the changes you made to the project.
* **Merge:** Merging will be done when you want to apply the changes you done in a branch. Otherwise it can be done by pull request where other user will pull request their work done by them and if you want to take the work, you can merge it through pull request.
* **Clone:** Clone is parallel version of the repository which resides on your desktop where we can edit the code using your favorite editor. However the changes is synced between remote version and local version.
* **Pull:** Pull comes into action when you want to fetch the file which is changed by some other person in your team into your local version, we will use pull.
* **Pull request:** Pull requests are done by the other user where they will propose the changes they made to your repository. You may accept or decline the pull request. For this there is a discussion board.

**Part 7:**

**Retrieve the README.md file at:**

[**https://github.com/paceuniversity/courses**](https://github.com/paceuniversity/courses)

**Add your name (lastname, firstname) in the file, add a comment (date and time) (REQUIRED), and update the README.md file at:** [**https://github.com/paceuniversity/courses**](https://github.com/paceuniversity/courses)**. Your name should appear at the provided** [**https://github.com/paceuniversity/courses**](https://github.com/paceuniversity/courses)**. Please check the work of previous students.**

**List the commands and strategy you use to do this part of the exercise in the *LastnameFirstnameGitTutorial-mm-dd-yyyy.docx* file and push it to:** [**https://github.com/yourpseudo/CSXXX20XX**](https://github.com/yourpseudo/CSXXX2016)**.**

**Steps:**

1. Open the link <https://github.com/paceuniversity/courses>

Login to your GitHub account and navigate through this link.

1. Fork the repository

On the top right corner of the repository, you can find the fork option. When you fork the repository you can have your personal copy of the project.

1. Edit the page

Once you are done with fork, you can open the readme page and on the top right you have an edit option, then you can edit the page.

1. Commit changes

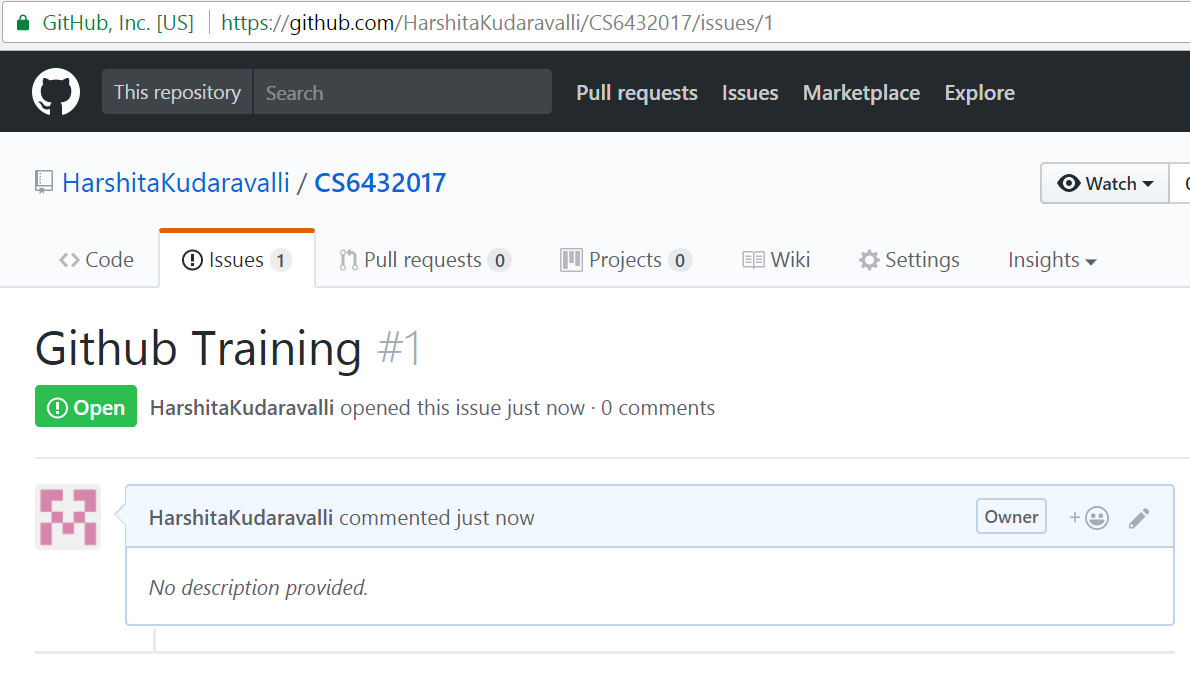
Once you are done with editing, at the bottom you can find commit changes option. By clicking this you can save the changes made.

1. Making a Pull request

At last, you can propose the changes made by you by making a pull resquest. On the top of the repository you can have compare and pull request option.

**Part 8:**

Add an issue with title “GitHub training” in your repository called CSXXX20XX. Issues will be used for tasks and bug reports.



**Part 9:**

\Edit the main page of the wiki in

your repository called CSXXX20XX. Add the title “CS XXX 20XX” to the page. The wiki will be used for documenting your work.

