GitHub and LaTeX installation instructions and tutorial

January 25, 2020

Instructions for installing LaTeX and setting up a document:

1. Download LaTeX: `https://www.latex-project.org/get/`. I recommend you go with `TeX Live` on Linux, `MacTeX` on macOS, and `MiKTeX` on Windows.

2. Go to overleaf.com, make an account, and sign in.

3. Click on new project and select one of the templates (Academic Journal, Book, etc.). These give you a reference that you can start from in your own `tex` file.

4. Create a directory on your computer where you want to have your LaTeX source.

5. Create a file with the extension `.tex`, and you can write your source code for your LaTeX document there.

6. To compile, you can type `pdflatex [...].tex` in terminal, or you install a LaTeX extension in VS Code, and follow its instructions for compiling.

Getting set up with GitHub and creating a GitHub project:

1. Install `git` if you don't already have it—`https://windows.github.com` for Windows, `https://mac.github.com` for macOS, `http://git-scm.com` for Linux.

2. Make a GitHub account and log in.

3. Create a GitHub organization for your team.

4. Go to the organization page—`https://www.github.com/[your_organization_name]`—and create a project for your LaTeX source code or your aircraft modeling/design code. Note that with free GitHub accounts, you can only make public projects.

5. On your computer, create an empty folder if you don't already have a folder for your LaTeX or Python source code that you want to turn into a project.

6. In terminal, `cd` into that empty or pre-existing folder (at the top level).

7. From there, type `git init` to initialize an empty git repository.

8. Type `git add [filename]` for each file you want to add. You can also use `git add *` to automatically add all files, but git will exclude certain files based on what is in `.gitignore`. This is useful, because you don't want to include automatically generated/temporary files, such as those created when you compile a LaTeX document. Conversely, sometimes you might want to override what is in `.gitignore` and add the file anyway, which you can do using `git add -f [...]` to force git to add a file, ignoring `.gitignore`.

9. Type `git commit -am "a descriptive commit message"`. This 'commits' your local changes (i.e., changes on your computer); in other words, it saves them.

10. Type something like `git remote add origin https://github.com/hwangjt/test.git` (replacing `hwangjt` and `test` with the appropriate names). This adds a `remote`, which is like a shortcut, called `origin`.

11. Type `git push -u origin master` to do two things. First, this is a 'push', so it pushes the commits on your computer to, in this case, 'origin'. Second, it sets 'origin' as the default place to push to, so that in the future, you can push by simply typing `git push`.

12. On a different computer, you can get up with the same repository by doing the following. First, go to the home page for the repository, and click the green `Clone or download` button. Copy the link in the text field. In terminal, type `git clone https://github.com/hwangjt/test.git` from the folder in which you want the project folder to be created.

13. You can now go into that created folder, and make changes, commit them, pull, and push to share your work with your teammates and vise versa.