Rylan Echalar

**Git Workflow**

Git workflow is an amazing way to stay organized and effectively doing projects/work. This tool can be used as an individual or as team to manage and share ideas. Also has no standardized process to how to interact with Git. When doing a team Git managed project, it’s important to know that an agreement on changes in the flow is made as a team. Being on the same page as a team, you agreed Git workflow should be developed or selected. There are multiple ways to publicized your Git workflow to fit your team's needs. It's overwhelming trying to figure out how to start and understand Git. The best part about Git workflow is that you can mix and match. Letting you design your workflow that suits you.

There are two main branches with an infinite lifetime that the central repo holds. First is the master branch, which is the main branch. It’s the HEAD of the source code that reflects the production state. Develop is the second branch, where you make changes for the next release. Some call it the “integration branch”. After making changes and ready to be release you must merge the changes to the master branch. Then tag with a release code. To make this easier to understand, think of it as a written essay. Having a first, second, and then your final draft. In each paper you making corrections and making sure you don’t make the same mistake as before.

In the main branches there is also different types of branches, Future, Release, and Hotfix. Each branch has a specific purpose and have strict rules in where they can be used and which branch they can be merge target. These branches are categorized by how they are used. Future branch is used in the develop and must be merge back into to develop. This branch is use to develop new features of upcoming or a distant future release. It's an idea that can be use in the future release/experiment. Release branch may branch off the develop and must be merge back to develop and master. Release is the preparation of the new production release. This is where you make sure everything is perfect and ready for the release. Lastly the Hotfix branch, where it may branch off the master and merge back into the develop and master. Hotfix and release branches are very much alike to prepare for the new production release. Hotfix is used more as a fix it now situation. The branch can be branched off from a corresponding tag on the master branch that marks the production version. It's important to ack first on the hotfix to resolve any problems in the master branch.

**Java Data Types**

There are two types of Java Dats Types, which are Primitive and Objects. There are eight Primitive data's in Java, “Byte, Short, Int, Long, Float, Double, Char, and Boolean”. Byte data is an 8-bit signed. It's used to save space in large arrays, mostly to place of integers. Short data is like an int but 2 times smaller than int. An Integer (Int) is usually a default data type, whole number. Long data is used for a wider range than an int. Float data is a single- precision data type. Double is where the decimal values. Boolean data type is where its True or False. It's used to look for simple flags. Char can be used as a character, for example “#, A, m”. The data types use only a small amount of memory to represent a single item of data. In Objective Data is the oppisite of a primitive data. It uses a large chunk of memory thaty holds data along with methods.

**5 Sources**

* [**https://www.atlassian.com/git/tutorials/comparing-workflows**](https://www.atlassian.com/git/tutorials/comparing-workflows)
* [**https://nvie.com/posts/a-successful-git-branching-model/**](https://nvie.com/posts/a-successful-git-branching-model/)
* [**https://nvie.com/posts/a-successful-git-branching-model/**](https://nvie.com/posts/a-successful-git-branching-model/)
* [**https://chortle.ccsu.edu/java5/Notes/chap09C/ch09C\_2.html**](https://chortle.ccsu.edu/java5/Notes/chap09C/ch09C_2.html)
* [**https://www.geeksforgeeks.org/data-types-in-java/**](https://www.geeksforgeeks.org/data-types-in-java/)
* [**https://cs.fit.edu/~ryan/java/language/java-data.html**](https://cs.fit.edu/~ryan/java/language/java-data.html)