

**CHEAT SHEAT  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**TABLE OF CONTENTS**

1. **Introduction**
2. **Creating a Snapshot**

**Introduction**

**What is Version Control?**

Version control, also known as source control, is the practice of tracking and managing changes to software code. Version control systems are software tools that help software teams manage changes to source code over time.a

**2 Types of version control system?**

* Centralized
* Distributed

**What is Git?**

**Git** is the most commonly used version control system.

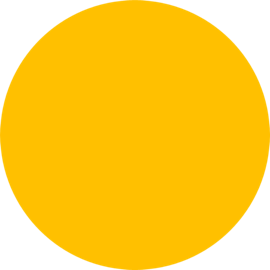
**Git** tracks the changes you make to files, so you have a record of what has been done, and you can revert to specific versions should you ever need to.

**Git** also makes collaboration easier, allowing changes by multiple people to all be merged into one source.

**Why Git?**

* **Free**
* **Open source**
* **Super-fast**
* **Scalable**
* **Cheap branching/ Merging**

**Git workflow**

****

****

**Directory Staging Area Repository**

**Creating Snapshots**

**Helpful Message**

git –help # Display help in full version

git -h # Display help in shorter version

**Initializing a repository**

git init

**Staging files**

git add sample.html # Stages a single file

git add sample.html sample2.html # Stages multiple files

git add \*.html # Stages with a pattern

git add . # Stages all the content of your current dir

**Viewing the Status**

git status # View the full status

git status -s # View the short status

**Committing the staged files**

git commit -m “Your Message” # Commits with one line of message

git commit # Open the default editor to type long message

**Skipping the staging area**

git commit -am “Message”