**What is GIT?**

* GIT is a distributed software version control system.
* GIT is part of SCM tools. (SCM tools include source management tools, source deployment tools, bla bla ))
* ( Source Control Management )
* Git is a free and open source, designed to handle everything from small to very large projects with speed and efficiency.

**Single person using the source control tool for his use alone.**

* Version control. He can have multiple versions of his own source code, so he can review them whenever he wants to.

**A team of persons using the source control for their teams.**

* A team can use this source control so that the entire team can contribute/review and approve the source code.

**Legacy SCM Tools:**

First one is the VCS == version control System.

CVS === Concurrent Version system.

These other systems (CVS Subversion, Perforce, Bazaar)

**Why is GIT better than other systems?**

* These other systems (CVS, Subversion, Perforce, Bazaar, and so on) think of the information they store as a set of files and the changes made to each file over time (this is commonly described as delta-based version control). Conceptually, those older systems store information as a list of file-based changes.

**How does GIT treat its Data?**

* Git does not think of or store its data as files or folders. Instead, Git thinks of its data more like a series of snapshots of a miniature filesystem.
* With Git, every time you commit, or save the state of your project, Git basically takes a picture of what all your files look like at that moment and stores a reference to that snapshot.
* To be efficient, if files have not changed, Git does not store the file again, just a link to the previous identical file it has already stored. Git thinks about its data more like a stream of snapshots.
* This makes Git more like a mini filesystem with some incredibly powerful tools built on top of it, rather than simply a VCS.

**Features of GIT**

1. Speed
2. Integrity

**Speed:**

* Most operations in Git need only local files and resources to operate — generally no information is needed from another computer on your network.
* If you're used to a CVS where most operations have that network latency overhead, this aspect of Git will make you think that the gods of speed have blessed Git with unworldly powers. Because you have the entire history of the project right there on your local disk, most operations seem almost instantaneous.

**GIT Has Integrity**

* Git stores everything in its database not by file name but by the hash value of its contents.
* Everything in Git is check summed before it is stored and is then referred to by that checksum.
* This means it is impossible to change the contents of any file or directory without Git knowing about it.
  + This functionality is built into Git at the lowest levels and is integral to its philosophy.
  + You cannot lose information in transit or get file corruption without Git being able to detect it.
* The mechanism that Git uses for this check summing is called a SHA-1 hash.
  + This is a 40-character string composed of hexadecimal characters (0-9 and a—f) and calculated based on the contents of a file or directory structure in Git.