Resources to read for DAY 1

- https://www.perforce.com/blog/vcs/what-is-version-control
- https://backlog.com/blog/git-vs-svn-version-control-system/

Frequently Asked Questions (FAQ's) for DAY 1

(Sorce - https://career.guru99.com/top-40-interview-questions-on-git/)

1. **What is GIT?**

   GIT is a distributed version control system and source code management (SCM) system with an emphasis to handle small and large projects with speed and efficiency.

2. **What is a repository in GIT?**

   A repository contains a directory named .git, where git keeps all of its metadata for the repository. The content of the .git directory are private to git.

3. **What are the advantages of using GIT?**

   a) Data redundancy and replication

   b) High availability

   c) Only one.git directory per repository

   d) Superior disk utilization and network performance

   e) Collaboration friendly

   f) Any sort of projects can use GIT

4. **What language is used in GIT?**

   GIT is fast, and 'C' language makes this possible by reducing the overhead of runtimes associated with higher languages.

5. **Why is GIT better than Subversion?**

   GIT is an open source version control system; it will allow you to run 'versions' of a project, which show the changes that were made to the code overtime also it allows you keep the backtrack if necessary and undo those changes. Multiple developers can checkout, and upload changes and each change can then be attributed to a specific developer.

6. **What is the difference between GIT and SVN?**

   The difference between GIT and SVN is

   a) Git is less preferred for handling extremely large files or frequently changing binary files while SVN can handle multiple projects stored in the same repository.

   b) GIT does not support ‘commits’ across multiple branches or tags. Subversion allows the creation of folders at any location in the repository layout.

   c) Gits are unchangeable, while Subversion allows committers to treat a tag as a branch and to create multiple revisions under a tag root.

Also, refer to this link (https://www.oshyn.com/blogs/2012/06/version-control-systems-distributed-vs-centralized)