Introduction to Git:

What is Git?

Git is a version control system that helps developers manage and track changes to their code. It allows you to:

1. **Track Changes**: Git records every change made to your project, including who made the change and when it was made.
2. **Collaborate**: Multiple developers can work on the same project simultaneously, making it easy to collaborate on code.
3. **Branching**: Git enables you to create branches to work on new features or bug fixes independently.
4. **Merging**: You can merge changes from different branches back into the main codebase when they are ready.
5. **History**: Git maintains a complete history of your project, making it easy to identify when and why specific changes were made.

Basic Concepts:

- **Repository (Repo)**: A Git repository is a directory that contains your project's files and the entire history of those files.
- **Commit**: A commit is a snapshot of your project at a specific point in time. Each commit has a unique identifier.
- **Branch**: A branch is a separate line of development. You can create branches to work on specific features or bug fixes.
- **Merge**: Merging is the process of combining changes from one branch into another.
- **Clone**: Cloning is creating a copy of a Git repository on your local machine.

Installing Git on Windows:

Follow these steps to install Git on a Windows machine:

1. Visit the official Git website: https://git-scm.com/downloads.
2. Download the Windows version of Git by clicking the "Download for Windows" button.
3. Run the downloaded installer.
4. Follow the installation wizard's instructions. You can generally accept the default settings.
5. During installation, you can choose the default text editor and other preferences. The default options are usually suitable for most users.
6. Once the installation is complete, open the Git Bash terminal (installed with Git) to start using Git.

Installing Git on a Mac:

Mac computers typically come with Git pre-installed. To check if Git is installed or to install it, follow these steps:

1. Open the Terminal application on your Mac. You can find it in the "Utilities" folder within the "Applications" folder.
2. To check if Git is installed, type the following command and press Enter:

   ```
   git --version
   ```

   If Git is installed, you'll see the installed Git version. If not, macOS will prompt you to install it. Follow the on-screen instructions to complete the installation.
3. Once Git is installed, you can start using it in the Terminal.

That's it! You now have Git installed on both your Windows and Mac machines, and you're ready to start using it for version control and collaboration in your IT company's projects.

Generating SSH Keys:

SSH keys are a secure way to authenticate yourself when accessing remote servers. They are also used to authenticate yourself when pushing changes to a Git repository. To generate SSH keys on Mac, follow these steps:

1. Open the Terminal application on your Mac.
2. Type the following command and press Enter:

   ```
   ssh-keygen -t ed25519 -C "your_email@example.com"
   ```

   Note:

   - If you are using an older version of macOS, use the following command instead:

     ```
     ssh-keygen -t rsa -b 4096 -C "your@email.com"
     ```
3. Press Enter to accept the default location for the SSH keys.
4. Enter a passphrase for the SSH key. You can also leave it blank.
5. Once the SSH key is generated, you'll see a message confirming the same.
6. To view the SSH key, type the following command and press Enter:

   ```
   ```

```
cat ~/.ssh/id_ed25519.pub
```

7. Copy the SSH key and paste it under the "SSH and GPG keys" section of your GitHub account.

Note : For newest Informations please visit this : https://docs.github.com/en/authentication/connecting-to-github-with-ssh/generating-a-new-ssh-key-and-adding-it-to-the-ssh-agent