* **Windows 11 installation**
* Preparation:
* System Requirements Check:
  + Ensure your PC meets the minimum system requirements for Windows 11:
  + 64-bit processor with at least 1 GHz clock speed and 2 or more cores.
  + 4 GB RAM.
  + 64 GB storage.
  + UEFI firmware with Secure Boot capability.
  + DirectX 12 compatible graphics / WDDM 2.x.
  + TPM version 2.0.
  + High definition (720p) display.
  + Internet connection and Microsoft account for some features.
* Backup Important Data:
  + Create backups of your important files to prevent data loss during the installation process.
* Download Windows 11:
* Check for Compatibility:
  + Visit the official Windows 11 system requirements page to confirm compatibility: [Windows 11 System Requirements] (https://www.microsoft.com/en-us/windows/windows-11-specifications).
* Download Windows 11 Installation Media:
  + Go to the Microsoft website or use the Media Creation Tool to download the Windows 11 installation files.
  + Visit [Microsoft's Download Windows 11 page] (https://www.microsoft.com/en-us/software-download/windows11) to download the Media Creation Tool.
* Create Installation Media:
  + Use the Media Creation Tool to create a bootable USB drive or download the ISO file that you can later burn to a DVD.
* Install Windows 11:
* Boot from Installation Media:
  + Insert the bootable USB drive or DVD with the Windows 11 installation files into your PC.
* Access BIOS/UEFI Settings:
  + Restart your computer and enter BIOS/UEFI settings (usually by pressing F2, F12, Esc, or Delete key during startup).
* Set Boot Priority:
  + In BIOS/UEFI settings, set the boot priority to boot from the USB drive or DVD where you have the Windows 11 installation files.
* Start Installation:
  + Save settings and exit BIOS/UEFI. Your PC will now boot from the installation media.
  + Follow the on-screen instructions to begin the installation of Windows 11.
  + Select your language, time, and keyboard preferences.
* Install Windows 11:
  + Click "Install Now" and proceed with the installation process.
  + Follow the prompts to choose where you want to install Windows (typically, select the primary drive and format it if necessary).
* Complete Installation:
  + Windows will now install. Your PC may restart several times during this process.
  + Follow the on-screen instructions to personalize your settings, such as region, keyboard layout, and Wi-Fi settings.
* Sign In and Update:
  + Sign in with your Microsoft account or create a new one.
  + Once Windows 11 is installed, check for and install any available updates via Settings > Update & Security > Windows Update.
* Install Drivers and Applications:
  + Install necessary drivers for your hardware (graphics card, network adapters, etc.).
  + Install your essential applications from the Microsoft Store or their respective websites.
* Data Restoration:
  + Restore your backed-up files to your new Windows 11 installation.
* Enjoy Windows 11:
  + Your PC is now ready with Windows 11 installed. Enjoy the new features and improvements!
* Additional Tips:
  + Product Activation: Ensure your Windows 11 installation is activated with a valid product key.
  + Troubleshooting: If you encounter any issues during installation, refer to Microsoft's support website or community forums for assistance.
* **Visual Studio Code Installation**
* Visit the Official Website:
  + Go to the Visual Studio Code website at [https://code.visualstudio.com/] (https://code.visualstudio.com/).
* Download for Your Operating System:
  + Click on the "Download for [Your Operating System]" button. VS Code supports Windows, macOS, and Linux.
* Install Visual Studio Code:
* For Windows:
* Run the Installer:
  + Once the download completes, run the installer (`.exe` file) that you downloaded.
* Accept User Account Control Prompts:
  + If prompted by User Account Control, click "Yes" to allow the installer to make changes to your device.
* Choose Setup Options:
  + Follow the installer prompts. You can choose the installation location and whether to add VS Code to the PATH variable (recommended).
* Complete Installation:
  + Click "Next" and then "Install" to begin the installation process. Wait for the installation to complete.
* Launch Visual Studio Code:
  + Once installed, you can launch VS Code from the Start menu or desktop shortcut.
* Launch Visual Studio Code:
  + Launch VS Code from the Applications menu or by typing `code` in the terminal.
* Getting Started with Visual Studio Code:
* Extensions:
  + Explore and install extensions to add languages, debuggers, themes, and more. Open VS Code and click on the Extensions view icon (`Ctrl + Shift + X`).
* Settings:
  + Customize VS Code to your preferences. Access settings by clicking on the gear icon in the bottom left corner or by pressing `Ctrl + ,`.
* Integrated Terminal:
  + VS Code includes an integrated terminal. Open it using ``Ctrl + ` `` (backtick).
* Useful Shortcuts:
  + Familiarize yourself with common shortcuts (`Ctrl + S` to save, `Ctrl + P` for quick file search, etc.).
* Integrated Git:
  + VS Code has built-in Git support. Initialize a repository or clone an existing one to start version controlling your projects.
* Installation of Git
* Download Git:
  + Visit the official Git website: [https://git-scm.com/](https://git-scm.com/)
  + Download the appropriate version of Git for your operating system (Windows)
  + Follow the installation instructions provided by the installer.
* Configure Git:
  + After installation, open a terminal (Command Prompt on Windows, Terminal on macOS/Linux).
  + Set your username and email address:
* git config --global user.name "Your Name"
* git config --global user.email "your.email@example.com"
  + This information is used for identifying your commits.
* Create a GitHub Account
* Sign up for GitHub:
  + Go to [https://github.com/] (https://github.com/).
  + Click on "Sign up" and follow the instructions to create your GitHub account.
  + Choose a username, enter your email address, and set a password.
* Initialize a Git Repository and Make Your First Commit
* Create a New Project Directory:
  + Decide where you want to keep your project files (e.g., `~/projects/myproject`).
  + Open a terminal and navigate to this directory:
* cd ~/projects/my project
* Initialize a Git Repository:
  + Inside your project directory, initialize a new Git repository:
* git init
* Create Your Project Files:
  + Add some files to your project directory (e.g., `index.html`, `script.js`, `styles.css`).
* Add Files to the Git Staging Area:
  + Stage the files you want to commit:
* git add index.html script.js styles.css
  + You can also use `git add .` to add all files in the directory.
* Commit Your Changes:
  + Commit the staged files with a commit message:
* git commit -m "Initial commit"
  + Replace `"Initial commit"` with a meaningful message describing your changes.
* Connect Your Local Repository to GitHub
* Create a New Repository on GitHub:
  + Log into your GitHub account.
  + Click on the "+" sign in the upper right corner and select "New repository."
  + Enter a repository name (e.g., `my project`), optionally add a description, and click "Create repository."
* Link Your Local Repository to the GitHub Repository:
  + On the GitHub repository page, under "Quick setup," copy the repository URL (ends in `.git`).
  + In your terminal, add the GitHub repository as a remote:
* git remote add origin https://github.com/your-username/myproject.git
  + Replace `https://github.com/your-username/myproject.git` with the URL you copied.
* Push Your Changes to GitHub:
  + Push your local commits to the GitHub repository:
* git push -u origin master
  + Enter your GitHub username and password if prompted.