

## AGENDA

*Today we will learn:*

- What is SSH protocol
- When we need to use SSH
- SSH Keys
- How to generate SSH keys
- How to configure git repository
- Known hosts
- Changing remote repository

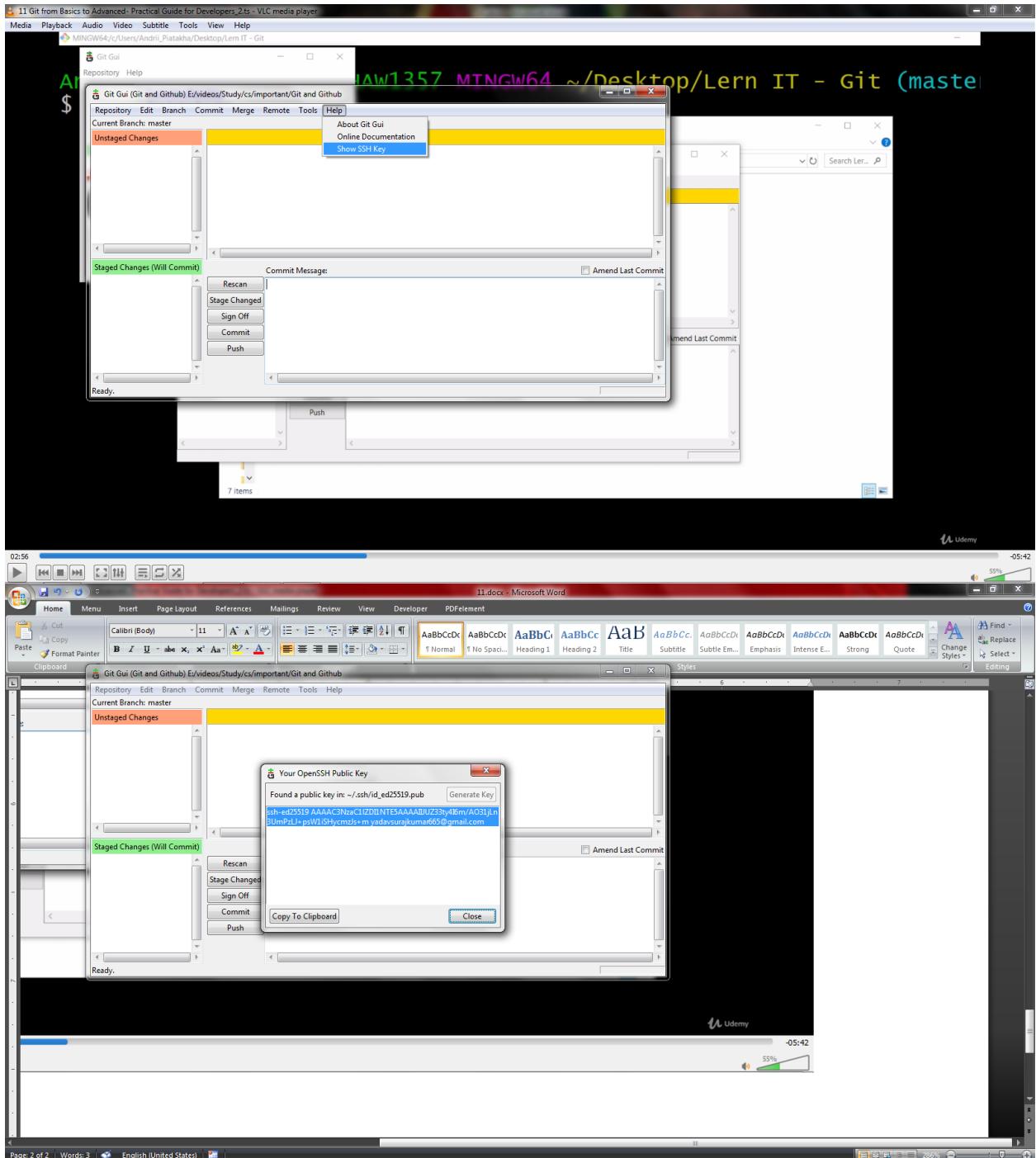


CONFIDENTIAL | © IT-Bulls.com



## Open Git GUI

A screenshot of a Microsoft Word document titled "11.docx - Microsoft Word". Overlaid on the Word window is a Git GUI interface. The Git GUI window has a toolbar with "Repository", "Edit", "Branch", "Commit", "Merge", "Remote", and "Tools" buttons. The "Current Branch: master" is selected. Below the toolbar, there are two sections: "Unstaged Changes" (empty) and "Staged Changes (Will Commit)". Under "Staged Changes", there is a "Commit Message:" field and several buttons: "Rescan", "Stage Changed", "Sign Off", "Commit", and "Push". To the right of the Git GUI, the Microsoft Word ribbon is visible, showing tabs like Home, Insert, Page Layout, etc. The status bar at the bottom of the Word window shows "Page 1 of 1 | Words: 3 | English (United States) | 285%". The footer of the slide also includes the IT-BULLS and Udemy logos.



it shows the already generated SSH keys, and if not press generate SSH keys

It also asks for a passphrase (similar like enter a word if you forget your password). If you don't want any passphrase just press enter without entering anything in the dialog box.

Now, open github → settings → SSH & GPG keys

The screenshot shows the GitHub settings page under 'SSH & GPG keys'. On the left, there's a sidebar with options like Profile, Account, Security log, Security & analysis, Emails, Notifications, Scheduled reminders, Billing, SSH and GPG keys (which is selected), Repositories, Organizations, Saved replies, Applications, and Developer settings. The main area is titled 'SSH keys' and contains two entries: 'hp-elitebook' and 'ASUS.Home'. Each entry has a key icon, a title, a SHA256 fingerprint, an 'Added on' date, a 'Last used' date, and a 'Delete' button. Below the list is a link to a guide on generating SSH keys.

Click on new SSH key and give any title and paste the newly generated SSH Keys

Now go to any directory on github and copy the SSH link as shown below

The screenshot shows a GitHub repository page for 'AndriiPiatakha/learnit\_git'. The top navigation bar includes links for Pull requests, Issues, Marketplace, and Explore. The repository name 'AndriiPiatakha / learnit\_git' is displayed. Below the repository name are buttons for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. The repository details section shows 'master' branch, 1 branch, 0 tags, and file lists for 'src', '.gitignore', 'README.md', and 'license.txt'. To the right, there's a 'Clone' section with options for HTTPS, SSH, GitHub CLI, and a copy link. Other sections include 'About' (no description), 'Releases' (no releases), 'Packages' (no packages), and 'Languages' (Java 100.0%).

and then set the above as remote

```
MINGW64/c/Users/Andrii_Piatakha/Desktop/Lern IT - Git
Andrii_Piatakha@EPUAKHAW1357 MINGW64 ~/Desktop/Lern IT - Git (master)
$ git remote -v
origin  https://github.com/AndriiPiatakha/learnit_git.git (fetch)
origin  https://github.com/AndriiPiatakha/learnit_git.git (push)

Andrii_Piatakha@EPUAKHAW1357 MINGW64 ~/Desktop/Lern IT - Git (master)
$ git remote set-url origin git@github.com:AndriiPiatakha/learnit_gi
git

Andrii_Piatakha@EPUAKHAW1357 MINGW64 ~/Desktop/Lern IT - Git (master)
$ git remote -v
origin  git@github.com:AndriiPiatakha/learnit_git.git (fetch)
origin  git@github.com:AndriiPiatakha/learnit_git.git (push)

Andrii_Piatakha@EPUAKHAW1357 MINGW64 ~/Desktop/Lern IT - Git (master)
$
```

To push files on already created git repository i.e. updating the repository type → “**git push origin master**”. Note: we don’t use ‘-u’ here, it means git repository knows the this master branch is already connected to the git remote location.