
git

MINGW32:/e/Programs/xampp/htdocs/ganev.bg

```
Welcome to Git (version 1.8.0-preview20121022)


Run 'git help git' to display the help index.
Run 'git help <command>' to display help for specific commands.

V@V-PC /e/Programs/xampp/htdocs/ganev.bg (master)
$ _
```

5 Questions

- What
- Who
- Why
- When
- How

What

- Distributed revision control and source code management system with an emphasis on speed
- GNU GPL v2

Who

- Linus Torvalds

Why

- “Take CVS as an example of what *not* to do; if in doubt, make the exact opposite decision”
- Support a distributed workflow
- Very strong safeguards against corruption, either accidental or malicious
- Very high performance

When

- 03. 04. 2005: Development start
- 06. 04. 2005: Announced
- 07. 04. 2005: Initial release
- 21. 12. 2005: Release 1.0

How

- Strong support for non-linear development
  - Rapid branching and merging
  - Specific tools for visualizing and navigating a non-linear development history
- Distributed development
  - Local copy of the entire development history
  - Local merges

How

- Compatibility with existing systems/protocols
  - HTTP, FTP, rsync, Git protocol, SSH
  - CVS server emulation – use of existing CVS clients and IDE plugins to access git repositories
  - Subversion repositories can be used directly with git-svn

How

- Efficient handling of large projects
  - Git is very fast and scalable
  - 100 times faster
  - Git does not get slower as the project history grows larger

How

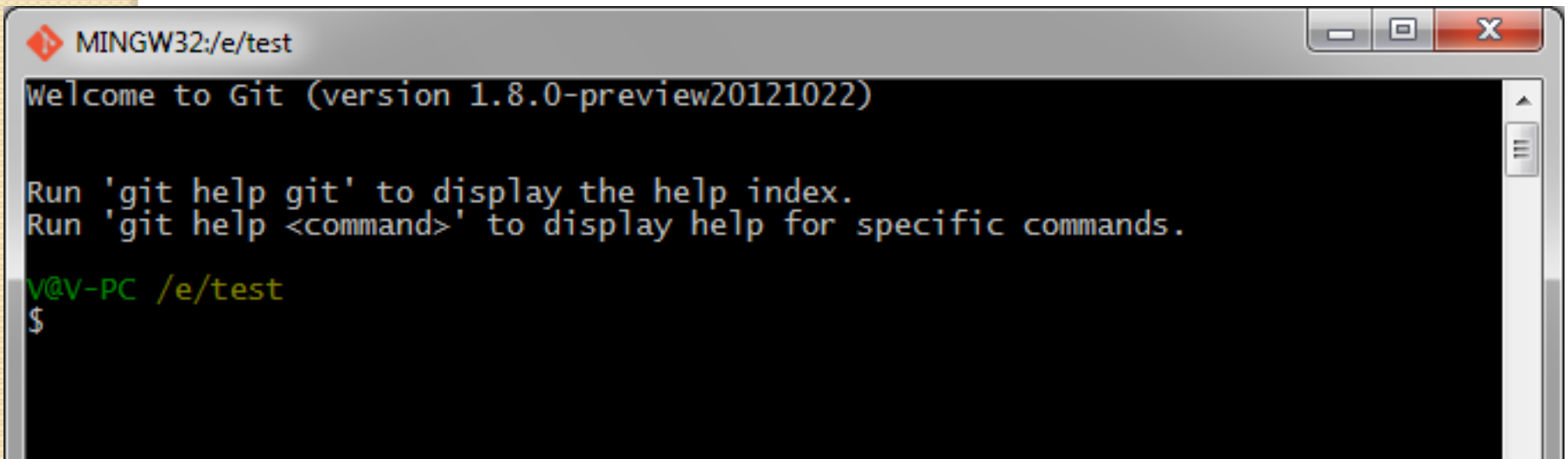
- Cryptographic authentication of history
- Toolkit-based design
  - A set of programs written in C
  - A number of shell scripts
- Pluggable merge strategies
  - Well-defined model of incomplete merge
  - Multiple algorithms for completing it

How

- Garbage accumulates unless collected
  - Garbage collection
- Periodic explicit object packing
- Snapshots
  - Renames are handled implicitly rather than explicitly

Let's start

- Install
  - Download from http://git-scm.com/
- Run
  - Make a new folder
  - Right click on it and then "Git Bash Here"



Good, console

- To create a new repository run
  - git init
- To clone a repository run
  - git clone git://[host]/[repo].git
- To connect your local repository with the one on the server run
  - git remote add origin git://[host]/[repo].git

Basic commands

- git status
  - shows the current status of local files:
    - modified
    - added (git add [filename])
    - removed (git rm [filename])
- git stash
  - saves the modified files to stack of unfinished changes that can be reapplied
    - git stash list
    - git stash apply {0}
    - git stash drop stash@{0}

Basic commands

- git add [filename]
  - adds all changed files
- git commit –m “commit message”
  - commits the changes to the local repository
- git push
  - pushes all changes to the server
- git pull
  - pulls changes from the server

Basic commands

- git branch hotfix
  - Creates a new branch “hotfix”
- git checkout hotfix
  - Switches to branch “hotfix”
  - Now do some stuff here and switch back to branch “master”
- git merge hotfix
  - Merges the “hotfix” branch to the current
- git branch -d hotfix
  - Deletes branch “hotfix”

SVN vs Git

	SVN	Git
Distributed	No	Yes
Satefy	Good	Very good
Performance	Very low	Ultra fast
Efficiency	Low	Extra
Rename files	It's a trap	No problem

And the winner is.. Git!

Reference

- http://git-scm.com/
- http://git-scm.com/book (Pro Git)
  - Dead tree version available @ Amazon
- http://gitimmersion.com/
- http://en.wikipedia.org/wiki/Git_(software)
- https://github.com/

Start now!

https://github.com/vcrazy

vihren.ganev@komfo.com