



Git & Github Basics

PHP 2013

@Rado_g

<http://radorado.me>

github.com/RadoRado

Disclaimer:

Не съм най-добрият Git-аджия, който ще срещнете.

Защо ни е Version Control System ?

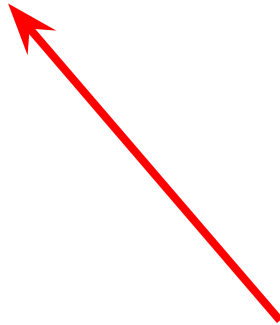
И защо не се преподава още в 1ви курс ?

Version Control System ?

- Инструменти, които ни позволяват да поддържахме различни **версии** на кода, който пишем
- Позволява ни да пишем големи проекти в екип (Без да се праща код по Skype)
- Много мощен ctrl + z (Няма фатални грешки)
- Ако писането на код е магията, то VCS е Амулет с +5 to all skills

Стандарният workflow с код

- Създаваме файлове
- Запазваме файлове
- Редактираме файлове
- **Запазваме файловете отново след промяна**



VCS - Запазва историята на промените по файловете

**Всеки път, когато
натиснем ctrl + s,
правим нова версия на
кода**

Без VCS пазим само последната.
Със VCS пазим тези версии, които
искаме.

Git != Github

Git е инструментът.

Github е облакът.

Може да използваме Git без Github.

Може да използваме Github без Git.

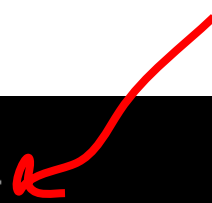
Най-добре е обаче, да ги ползваме
заедно.

Малко терминология.

- **git** = shell command за работа с Git
- **repo** = Repository / Директорията, където се намира кода, както **локално**, така и в Github
- **commit** - Глагол, означава да запазим конкретната версия на кода **локално**
- **push** - Глагол, означава да запазим **последната commit-ната версия** на централизиран сървър
- **diff** - Разликата между две версии на кода

Git е инструмент, който запазва версии на кода локално !

```
drwxr-xr-x  9 данч Administ 4096 Feb 15 19:27 ./
drwxr-xr-x  1 данч Administ 4096 Feb 15 19:03 ../
drwxr-xr-x 13 данч Administ 4096 Feb 15 19:38 .git/
-rw-r--r--  1 данч Administ  160 Feb 15 19:37 .gitignore
drwxr-xr-x  7 данч Administ 4096 Feb 15 19:27 Smarty/
-rw-r--r--  1 данч Administ 1834 Feb 25 23:17 index.php
drwxr-xr-x  5 данч Administ 4096 Feb 25 16:37 public/
```



Инсталация на Git ?



**На <https://help.github.com/> има обяснение и
като за баба ви.**

Има UI, има и Command Line Tools.
Изборът е ваш.

Работа с Git, локално

За да може да използваме Git, трябва да кажем, че директорията ще ползва Git.

git init

```
@RADORADO /c/code
$ mkdir testproject

@RADORADO /c/code
$ cd testproject/

@RADORADO /c/code/testproject
$ git init
Initialized empty Git repository in c:/code/testproject/.git/
```

**Това е! Директорията е
белязана с .git папка и
версиите ще се пазят
там.**

Не забравяйте, все още всичко се случва
локално.

Ако ви изгори лаптопа, изгаря и кода.

Git workflow : status, add, commit (and push)

git status

git add <filename>

git add <filename>

...

git commit -m 'A commit message!'

```
$ git status
# On branch master
#
# Initial commit
#
nothing to commit (create/copy files and use "git add" to track)
```

Добавяме нов файл. Статусът се променя.

```
$ touch index.php
```

```
$ git status
# On branch master
#
# Initial commit
#
# Untracked files:
#   (use "git add <file>..." to include in what will be committed)
#
#       index.php
nothing added to commit but untracked files present (use "git add" to track)
```


git add <filename>

```
@RADORADO /c/code/testproject (master)
$ git add index.php
warning: LF will be replaced by CRLF in index.php.
The file will have its original line endings in your working directory.

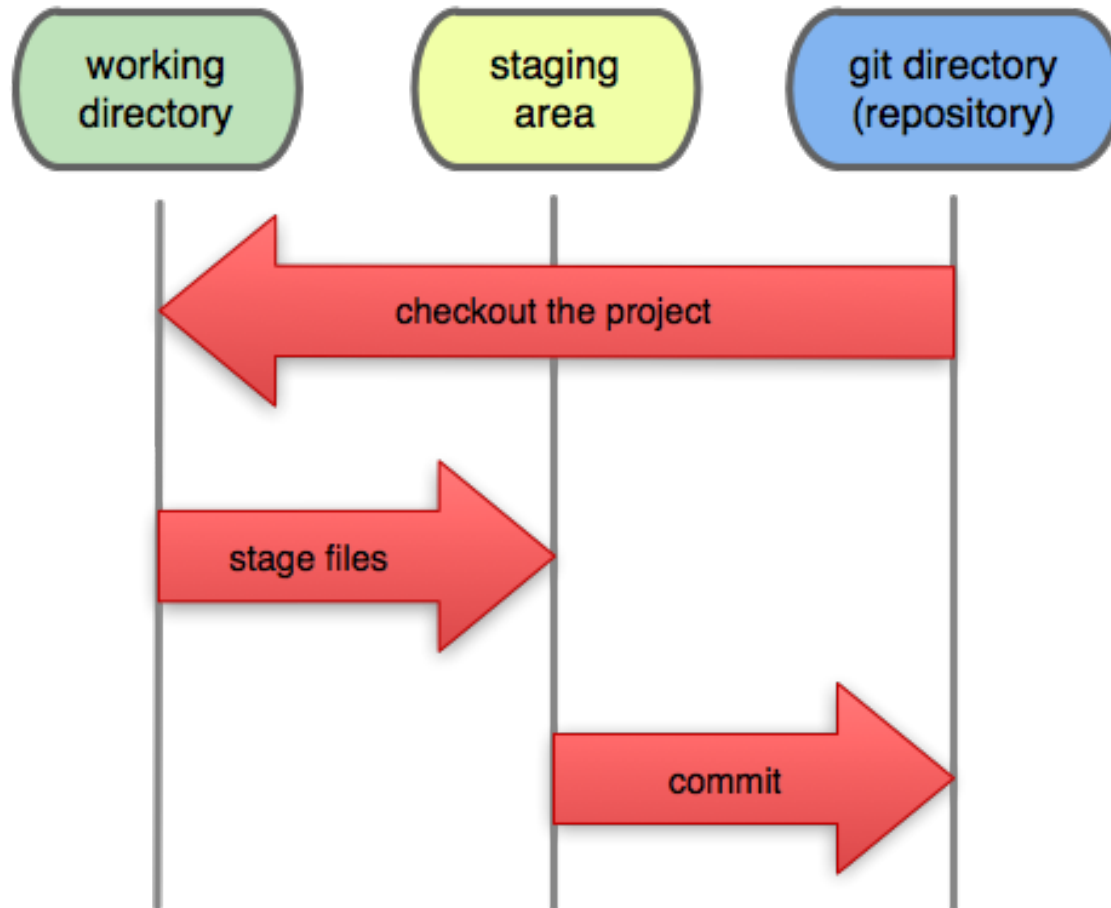
@RADORADO /c/code/testproject (master)
$ git status
# On branch master
#
# Initial commit
#
# Changes to be committed:
#   (use "git rm --cached <file>..." to unstage)
#
#       new file:   index.php
#
```

Когато сме готови, чрез `git commit` създаваме нова версия (локално)

```
@RADORADO /c/code/testproject (master)
$ git commit -m 'Adding index php to the project'
[master (root-commit) 9c6a5bf] Adding index php to the project
warning: LF will be replaced by CRLF in index.php.
The file will have its original line endings in your working directory.
1 file changed, 2 insertions(+)
create mode 100644 index.php
```

```
@RADORADO /c/code/testproject (master)
$ git status
# On branch master
nothing to commit, working directory clean
```

Local Operations



So far So good. Ами Github ?

Github е обкалът-услуга (Почти като Обкалът-Атлас), в който може да си запазите вашето repository.

Ако ви изгори компютъра, *git clone* *<repo_url>* и сте пак в играта.

Github е този централен сървър, където може да си пазите кода И да работите с други програмиси в/у проекта.

За да вържете Git с Github, ви трябва `git remote`

Global setup:

Download and install [Git](#)

```
git config --global user.name "Tekkub"
```

```
git config --global user.email tekkub@github.com
```

Next steps:

```
mkdir help-example
```

```
cd help-example
```

```
git init
```

```
touch README
```

```
git add README
```

```
git commit -m 'first commit'
```

```
git remote add origin git@github.com:tekkub/help-example.git
```

```
git push origin master
```

Existing Git Repo?

```
cd existing_git_repo
```

```
git remote add origin git@github.com:tekkub/help-example.git
```

```
git push origin master
```

Importing a Subversion Repo?

[Click here](#)



git push <remote> <branch> -
ВСИЧКИ НОВИ ВЕРСИИ НА СЪРВЪРА!

git remote -v ще ви изкара списък с всички
отдалечени сървъри, които имате

git remote show <remote_name> ще ви
изкара подробна информация за
съответният remote

git branch ще ви покаже в кой branch се
намирате

**За програмист да
използва VCS е като за
хирург да си мие ръцете
преди операция.**

<http://www.slideshare.net/spolsky/outc>

**Най-лесно - направете
си акаунт и започнете да
тествате.**

<http://learn.github.com/p/intro.html>

**Ако ще работите по
двойки, играта е друга.**

Разгледайте <https://github.com/nvie/gitflow>

Полезни ресурси

- <https://github.com/github/gitignore>
- <http://windows.github.com/>
- Pro-Git Book - <http://git-scm.com/book>