



ST1RT

Vasilis Tzelepopoulos

Git & Github



ΔΗΜΟΣ
ΑΘΗΝΑΙΩΝ

ΣΕΡΑΦΕΙΟ
ΑΘΛΗΤΙΣΜΟΣ, ΠΟΛΙΤΙΣΜΟΣ, ΚΑΙΝΟΤΟΜΙΑ



Microsoft



SOCIAL
INNOV

Τι είναι το Git

Το Git είναι ένα κατανεμημένο σύστημα ελέγχου έκδοσης
---Distributed Version Control System (DVCS)---

Git + Command Line = Love

“The command line is the only place you can run **all Git commands** – most of the GUIs only implement some subset of Git functionality for simplicity.”

<https://git-scm.com/book/en/v2/Getting-Started-The-Command-Line>

Git installation

<https://git-scm.com/downloads>

#####

Εναλλακτικά (για την ενσωμάτωση με το Github):

Windows: <https://windows.github.com>

Mac: <https://mac.github.com>

Git Clients

<https://git-scm.com/downloads/guis/>

Git init

- `$ mkdir example`
- `$ cd example`
- `$ git init`

Initialized empty Git repository in
/home/epicurus/Desktop/COURSES/Git/example/.git

Git status

- **\$ git status**

On branch master

Initial commit

nothing to commit (create/copy files and
use "git add" to track)

Προσθέτω αρχεία

Ας φτιάξουμε τώρα ένα αρχείο το οποίο δεν γνωρίζει το Git. Ανοίξτε τον αγαπημένο σας editor, ονομάστε το αρχείο `hi.py`, και γράψτε:

```
# hi.py  
print('hi Git! I am new here!')
```

Πληκτρολογήστε ξανά την εντολή:

```
git status
```


Προσθέτω αρχεία II : git add

git add

- `$ git add hi.py`
 - `$ git status`
- On branch master

Initial commit

Changes to be committed:

(use "`git rm --cached <file>...`" to unstage)

new file: hi.py

Πραγματοποιώντας αλλαγές

- **\$ git commit -m "creating hello.py"**
[master (root-commit) 25b09b9] creating hi.py
1 file changed, 3 insertions(+)
create mode 100755 hi.py
- **\$ git status**
On branch master
nothing to commit, working directory clean

Staging Area

Όταν δουλεύετε με ένα αρχείο στο Git (υποθέτουμε πως έχει γίνει commit ήδη μία φορά), μπορεί να υπάρχουν τρεις εκδόσεις του αρχείου που μπορείτε να εργαστείτε:

- η έκδοση στον σκληρό σας δίσκο που δουλεύετε
- μια διαφορετική έκδοση που έχει αποθηκευτεί στο Git
- την πιο πρόσφατη έκδοση που έχει συνδεθεί στο repo

Τι (δεν) βάζω στο αποθετήριο

Ο βασικός κανόνας για όλα τα version control systems είναι:

- Τοποθετήστε μόνο αρχεία πηγαίου κώδικα στον έλεγχο έκδοσης, όχι αρχεία που παράγονται από άλλη διαδικασία.
- Προσοχή: Ποτέ μην τοποθετείτε εμπιστευτικές πληροφορίες σε ένα δημόσιο χώρο αποθήκευσης στο GitHub. Οι κωδικοί πρόσβασης, τα κλειδιά API και παρόμοια στοιχεία δεν θα πρέπει να δεσμεύονται σε ένα repo. Κάποιος (μπορεί και) θα τα βρει τελικά.

SHA

Κάποιες φορές το Git χρησιμοποιεί την ολοκληρωμένη τιμή των 20 χαρακτήρων για να σας δείξει ένα SHA:

- `commit 25b09b9cc9191aa2d09444f1b50fa2b4c979c`

Μερικές φορές σας δείχνει μια συντομότερη έκδοση:

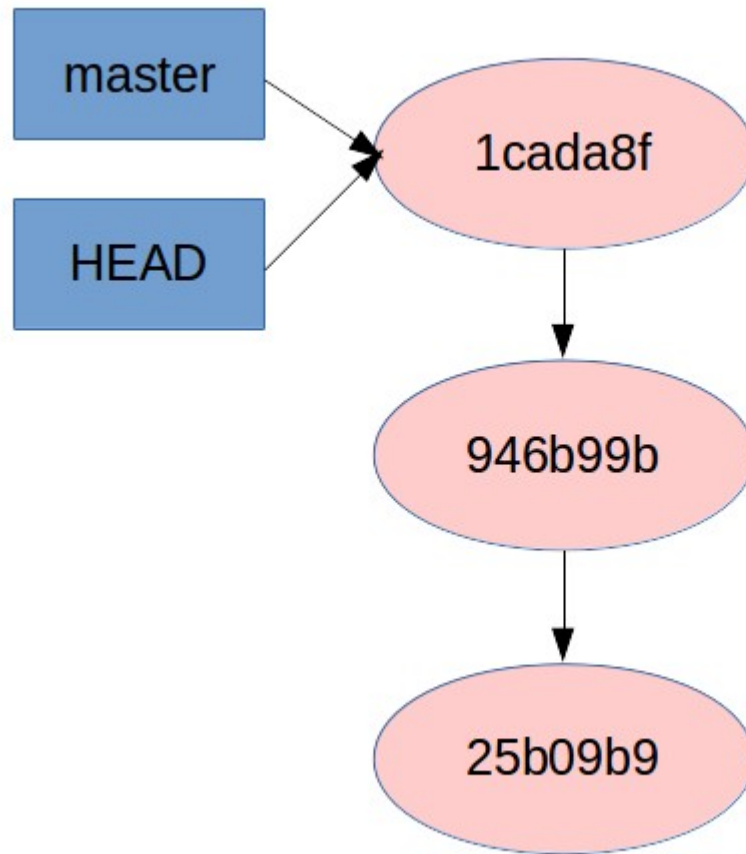
- `[master (root-commit) 25b09b9]` όταν έφτιαξε το hi.py

Git log

- `$ git log`
- `$ commit 2edea9ac5c4c66c5435ae0e3b88511841797c1ea
(HEAD -> master)
Author: Vatz1232 <vatz1232@gmail.com>
Date: Wed Oct 26 20:09:28 2019 +0300

creating hi.py`

Back to the future



Back to the future II

- **\$ git checkout 946b99bfe1641102d39f95616ceaab5c3dc960f9**

Note: checking out '946b99bfe1641102d39f95616ceaab5c3dc960f9'.

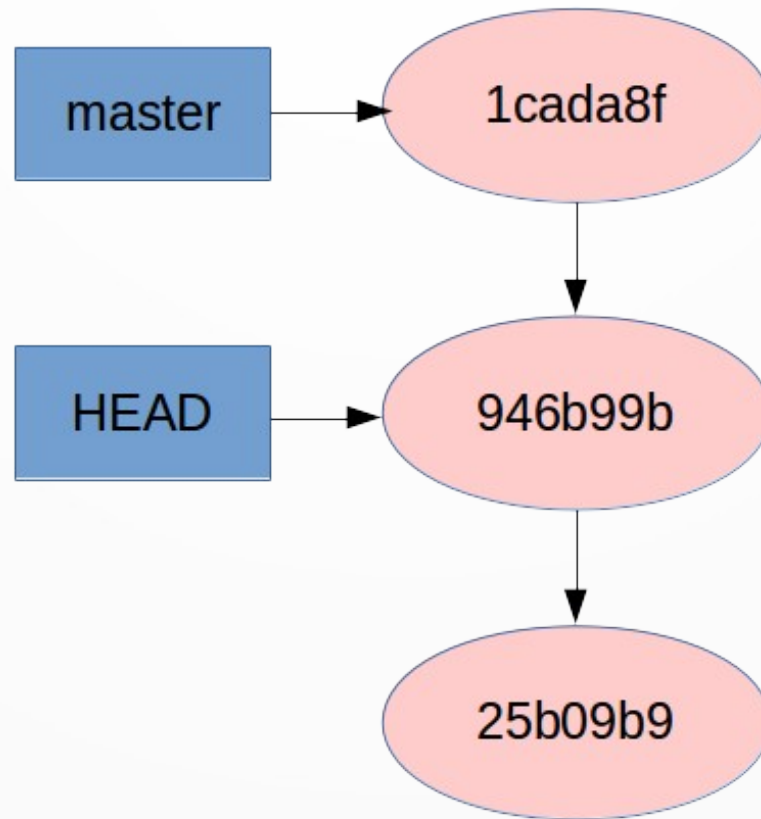
You are in 'detached HEAD' state. You can look around, make experimental changes and commit them, and you can discard any commits you make in this state without impacting any branches by performing another checkout.

If you want to create a new branch to retain commits you create, you may do so (now or later) by using `-b` with the checkout command again. Example:

- **git checkout -b <new-branch-name>**

HEAD is now at 946b99b... added myname module


Back to the future III






git checkout 1cada8f

git checkout master

Δημιουργώντας ένα repo στο Github

 Search or jump to... Pull requests Issues Marketplace Explore



  

Create a new repository

A repository contains all the files for your project, including the revision history.


Owner


Repository name

 epicurusgarden / techtalent 


Great repository names are short, simple, and unique. Your new repository will be created as **techtalent-studio** by **epicurusgarden**.

Description (optional)

☒  **Public**
Anyone can see this repository. You choose who can commit.

☐  **Private**
You choose who can see and commit to this repository.

☐ **Initialize this repository with a README**
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: **None** | Add a license: **None** 


Create repository

© 2018 GitHub, Inc. [Terms](#) [Privacy](#) [Security](#) [Status](#) [Help](#)



[Contact GitHub](#) [Pricing](#) [API](#) [Training](#) [Blog](#) [About](#)

Δημιουργώντας ένα repo στο Github II


 Search or jump to... / Pull requests Issues Marketplace Explore

epicurusgarden / techtalent

Unwatch 1 Star 0 Fork 0

[Code](#) [Issues 0](#) [Pull requests 0](#) [Projects 0](#) [Wiki](#) [Insights](#) [Settings](#)


Quick setup — if you've done this kind of thing before

or **HTTPS** **SSH** 

We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).


...or create a new repository on the command line

```
echo "# techtalent" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/epicurusgarden/techtalent.git
git push -u origin master
```



...or push an existing repository from the command line

```
git remote add origin https://github.com/epicurusgarden/techtalent.git
git push -u origin master
```



...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

[Import code](#)

Δημιουργώντας ένα repo στο Github IV

```
git remote add origin [URL]  
git remote -v
```

CHECK IT

Ασκησούλα

- α. Δημιουργείστε ένα αρχείο README στο αποθετήριο σας με τουλάχιστον 2 γραμμές κειμένου.
- β. Προσθέστε στο αποθετήριό σας ένα ακόμη αρχείο

\$KON@K1

```
git add [FILE]
git commit -m "MESSAGE"
git status
git push
```


Branches

- **\$ git checkout -b my_new_feature**
Switched to a new branch 'my_new_feature'
- **\$ git status**
On branch my_new_feature
nothing to commit, working directory clean

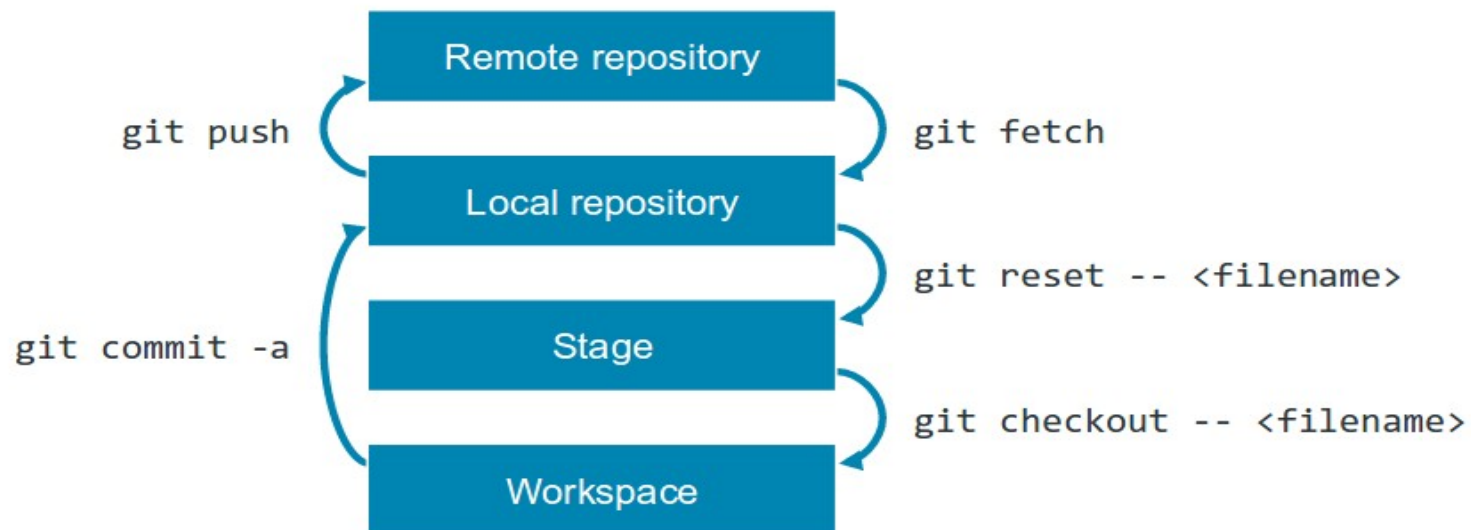
Δουλεύοντας με απομακρυσμένα αποθετήρια

clone
fetch
pull
push

git clone
<https://github.com/gitextensions/gitextensions.git>

Δουλεύοντας με απομακρυσμένα αποθετήρια II

Git 102



Μια απλή ροή εργασίας στο Git

git status - Βεβαιωθείτε ότι η τρέχουσα περιοχή εργασίας σας είναι εντάξει.

git pull - Αποκτήστε την πιο πρόσφατη έκδοση από το απομακρυσμένο αποθετήριο. Αυτό μπορεί να προλάβει προβλήματα συγχώνευσης αργότερα.

Επεξεργαστείτε τα αρχεία σας και πραγματοποιήστε τις αλλαγές.

git status - Βρείτε όλα τα αρχεία που έχουν αλλάξει.

git add [files] - Προσθέστε τα αρχεία που έχουν αλλαγές στο stage.

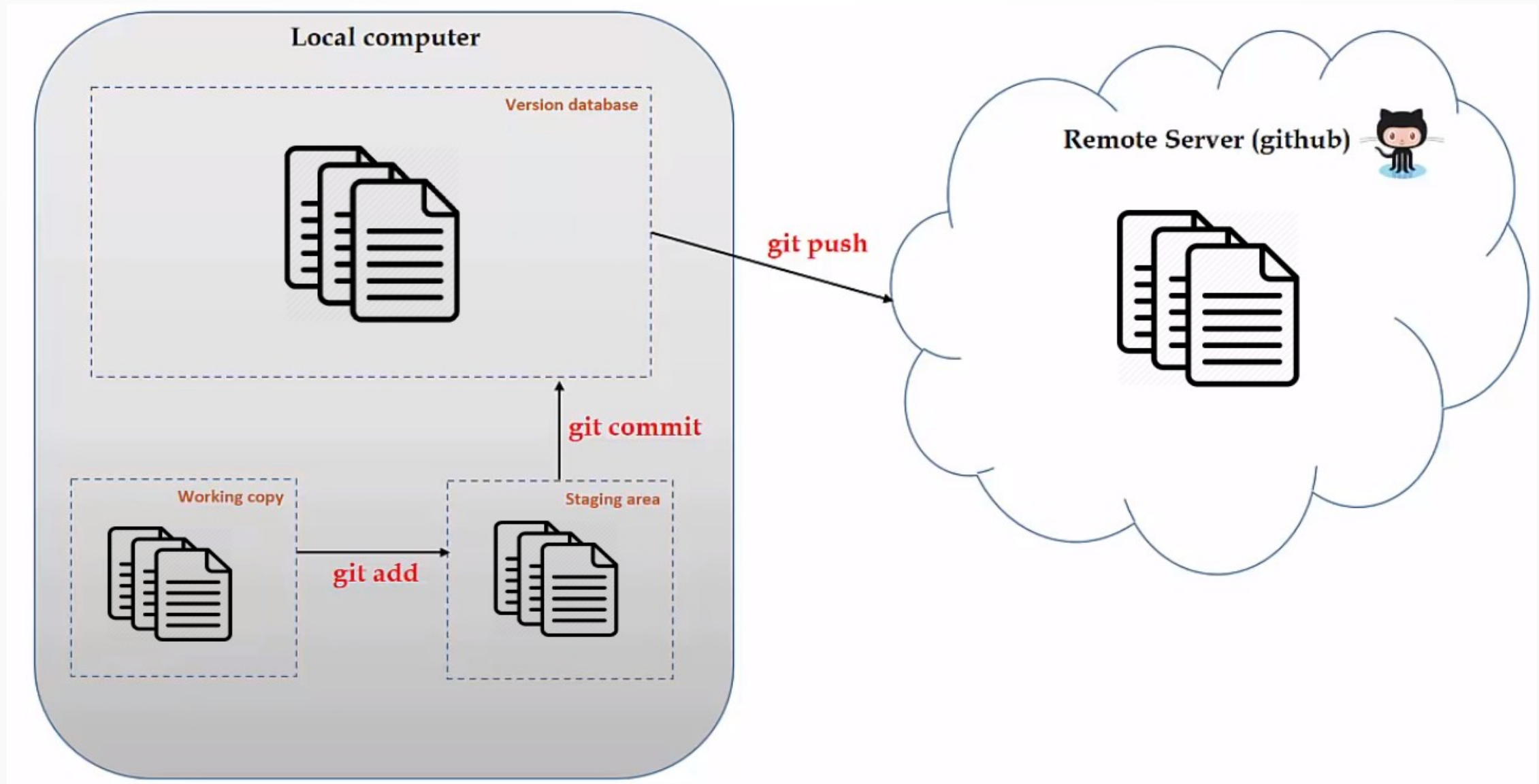
git commit -m "message" - Κάντε ένα νέο commit.

git push origin [branch-name] - Προωθήστε τις αλλαγές σας στο αποθετήριο.

Ροή εργασίας στο Git & Github



Ροή εργασίας στο Git & Github II



Let's practice!

<https://learngitbranching.js.org/>

Εργαλεία

<https://www.gitkraken.com/>

Πηγές:

Official git site and tutorial

<https://git-scm.com/>

GitHub guides

<https://guides.github.com/>

Commands cheatsheet

<https://education.github.com/git-cheat-sheet-education.pdf>

Interactive git tutorial

<https://try.github.io/>

Visual/interactive cheatsheet

<http://ndpsoftware.com/git-cheatsheet.html>

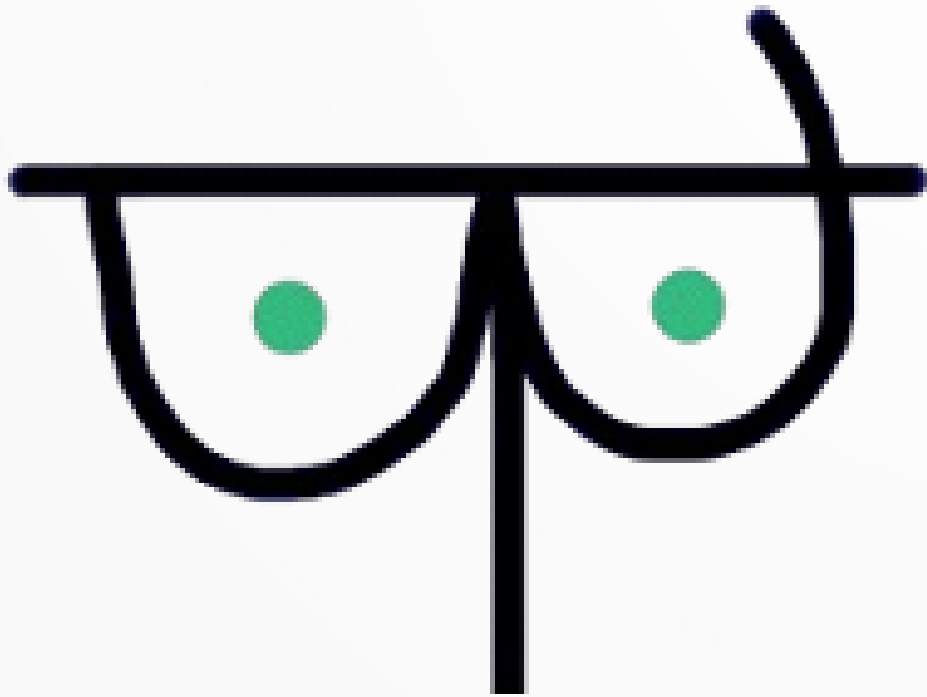
Ένα tweet για το δρόμο...



"The most valuable thing you can make is a mistake - you can't learn anything from being perfect."

Adam Osborne

Ευχαριστώ για την προσοχή σας!



Twitter: @vatzelos



ST1RT

JOIN US

WWW.STARTPROJECT.GR



SPONSORS

