

# Πρακτική εξάσκηση για το Git

GreekLUG

May 26, 2023

## Abstract

Ασκήσεις για το σεμινάριο Git των GreekLUG. Οι πρακτικές αυτές ασκήσεις θα σας βοηθήσουν:

- να ανοίγετε λογαριασμούς σε δικτυακές υπηρεσίες για τη φιλοξενία αποθετηρίων,
- να ζητάτε πρόσβαση σε αποθετήρια,
- να δουλεύετε με πολλαπλούς κλάδους,
- και να συγχρονίζεται αποθετήρια.

## Contents

1	Εισαγωγή	1
2	Δημιουργία λογαριασμού σε δικτυακή υπηρεσία	1
3	Απόκτηση πρόσβασης σε δικτυακό αποθετήριο	3
4	Εργασία με πολλαπλούς κλάδους	5
5	Συγχρονισμό με απομακρυσμένο αποθετήριο	5
	<b>Bibliography</b>	<b>6</b>

## List of Figures

1	Τελική κατάσταση αποθετηρίου άσκησης . . . . .	5
---	--	---

## List of Tables

# 1 Εισαγωγή

Οι ασκήσεις σχεδιάστηκαν για να ελέγξετε το πόσο καλά μπορείτε να χειριστείτε το Git. Σημειώστε όποια δυσκολία αντιμετωπίσετε για να τη συζητήσουμε στο τελευταίο μάθημα. Σε αυτή τη σειρά πρακτικών ασκήσεων, θα

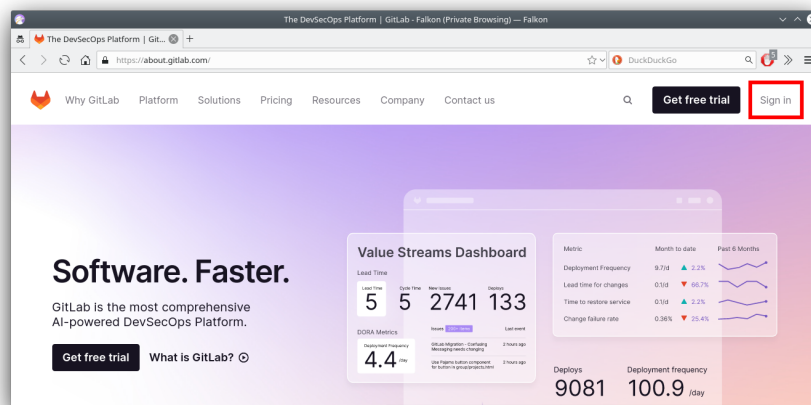
- ανοίξουμε ένα λογαριασμό σε δικτυακές υπηρεσίες για τη φιλοξενία αποθετηρίων,
- θα ζητήσουμε πρόσβαση σε ένα δοκιμαστικό αποθετήριο,
- δουλέψουμε με πολλαπλούς κλάδους,
- και θα συγχρονίσουμε ένα τοπικό αποθετήριο με ένα απομακρυσμένο.

Ο εκτιμώμενος χρόνος ολοκλήρωσης είναι 1 με 2 ώρες, ανάλογα με την εξοικείωσή σας με δικτυακές υπηρεσίες φιλοξενίας αποθετηρίων.

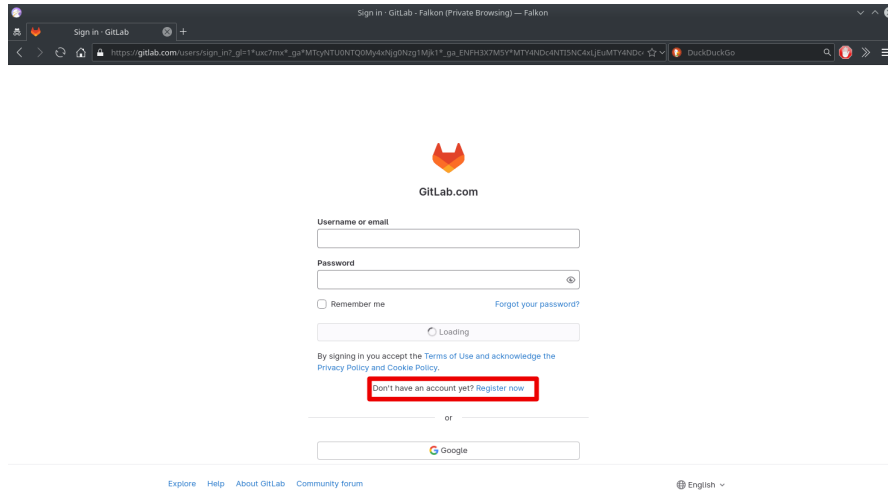
## 2 Δημιουργία λογαριασμού σε δικτυακή υπηρεσία

Δικτυακές υπηρεσίες όπως το GitLab συνήθως επιτρέπουν την αντιγραφή ανοιχτών αποθετηρίων μέσω HTTPS χωρίς ταυτοποίηση αλλά απαγορεύουν τη συνεισφορά κώδικα από ανώνυμους χρήστες. Για να δοκιμάσετε λοιπόν εντολές συγχρονισμού του αποθετηρίου σας με ένα δικτυακό αποθετήριο στην section 5, αν το αποθετήριο φιλοξενείτε σε κάποια δικτυακή υπηρεσία, θα χρειαστεί να δημιουργήσετε λογαριασμό στην υπηρεσία όπου φιλοξενείται. Για παράδειγμα, σε αυτή την άσκηση θα χρησιμοποιήσουμε το GitLab. Αν επιθυμείτε να δουλέψετε μόνο τοπικά ή αν έχετε ρυθμίσει κάποιο δικό σας δικτυακό αποθετήριο (π.χ. μέσω ssh) τότε δε χρειάζεται να εργαστείτε με το GitLab. Για τη δημιουργία λογαριασμού στο GitLab:

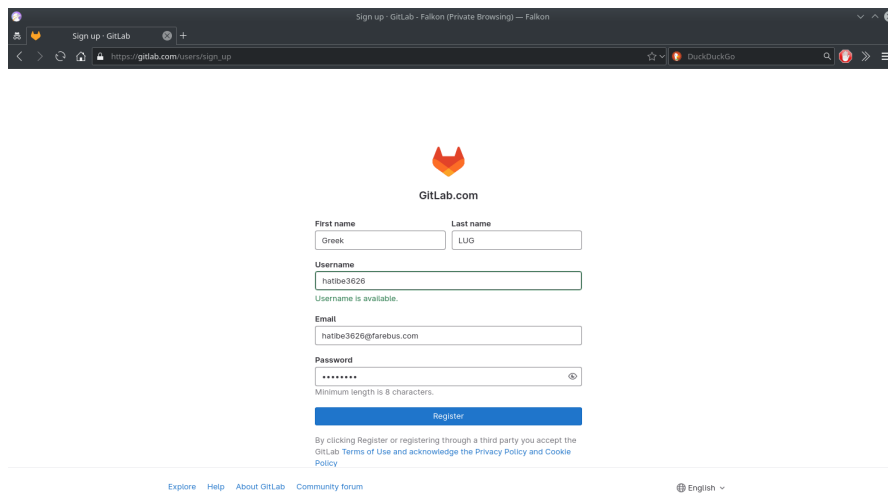
1. κατευθυνθείτε στην αρχική σελίδα του GitLab κι ανοίξετε το σύνδεσμο εισόδου,



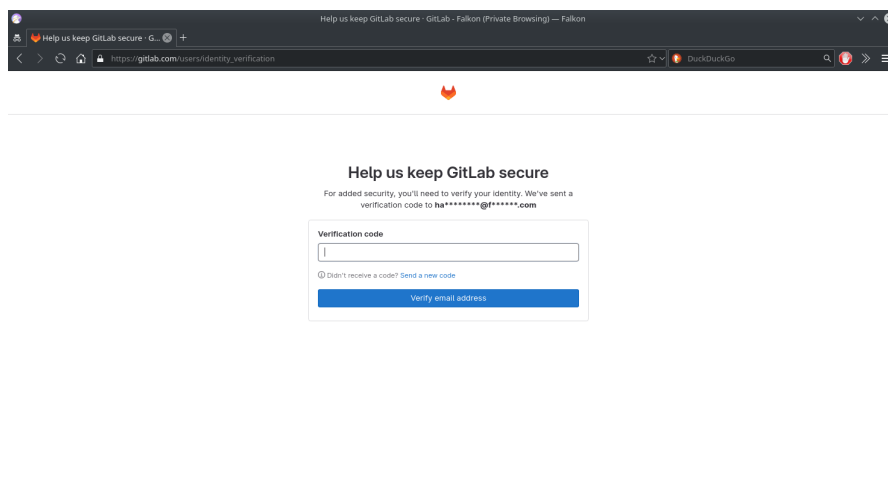
2. επιλέξτε τη δημιουργία νέου λογαριασμού,



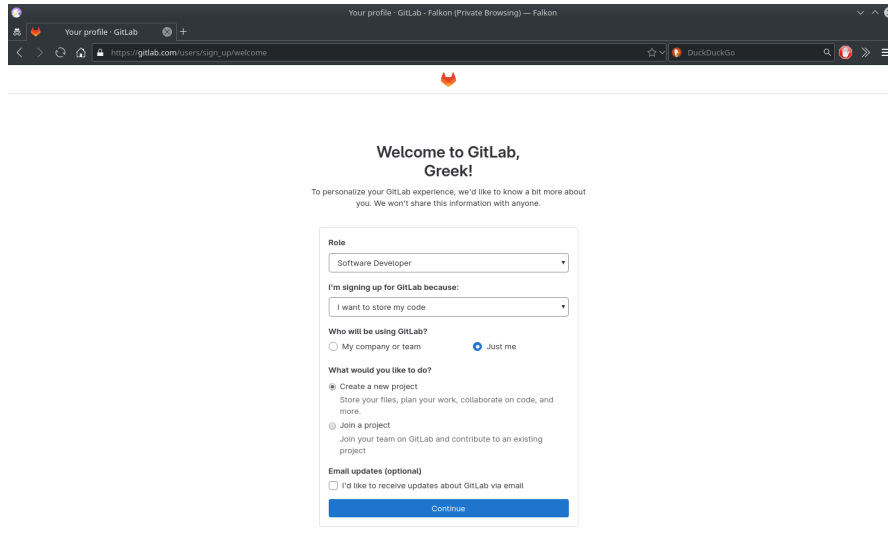
3. εισάγετε τα στοιχεία σας,



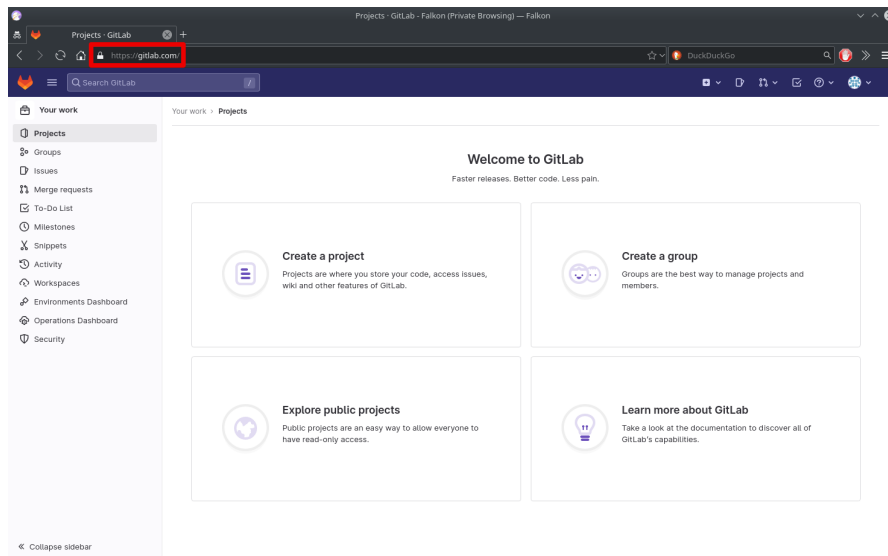
4. επιβεβαιώστε τη διεύθυνση ηλεκτρονικού ταχυδρομείου σας,



5. αγνοήστε τις περαιτέρω ρυθμίσεις καθώς είναι καλύτερο να συμπληρωθούν καθώς προκύπτουν οι αντίστοιχες ανάγκες, και τέλος



6. κατευθυνθείτε στην αρχική σελίδα του GitLab.

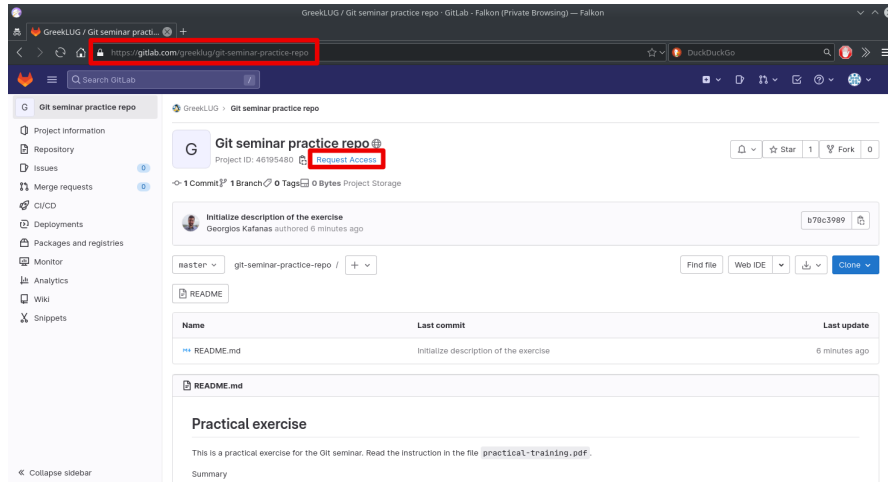


Ο λογαριασμός σας είναι τώρα έτοιμος, και μπορείτε να τον χρησιμοποιήσετε για να συνεισφέρετε κώδικα σε αποθετήρια που φιλοξενούνται στο GitLab.

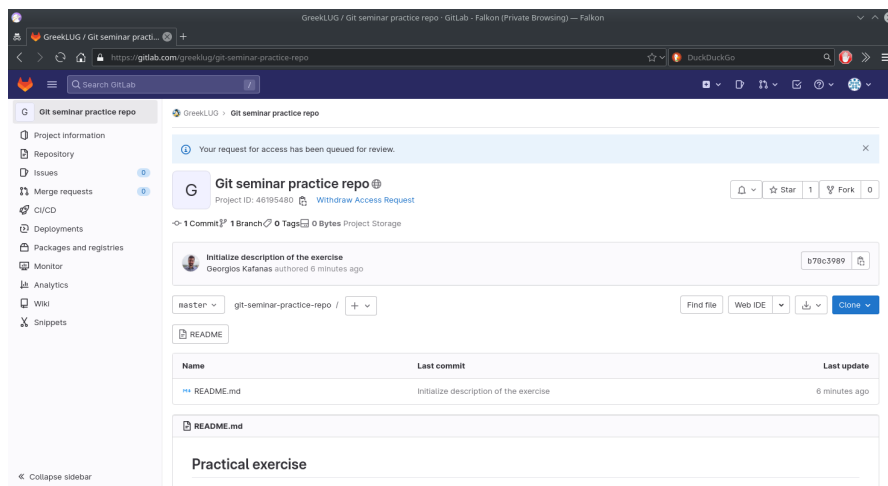
### 3 Απόκτηση πρόσβασης σε δικτυακό αποθετήριο

Το GitLab απαιτεί την ταυτοποίηση χρηστών που συνεισφέρουν σε αποθετήρια ώστε οι διαχειριστές να ελέγχουν ποιος εισάγει κώδικα στο αποθετήριο υπό τη διαχείρισή τους. Για να ζητήσετε άδεια να ανεβάσετε κώδικα στο αποθετήριο της άσκησης που φιλοξενείται στο GitLab:

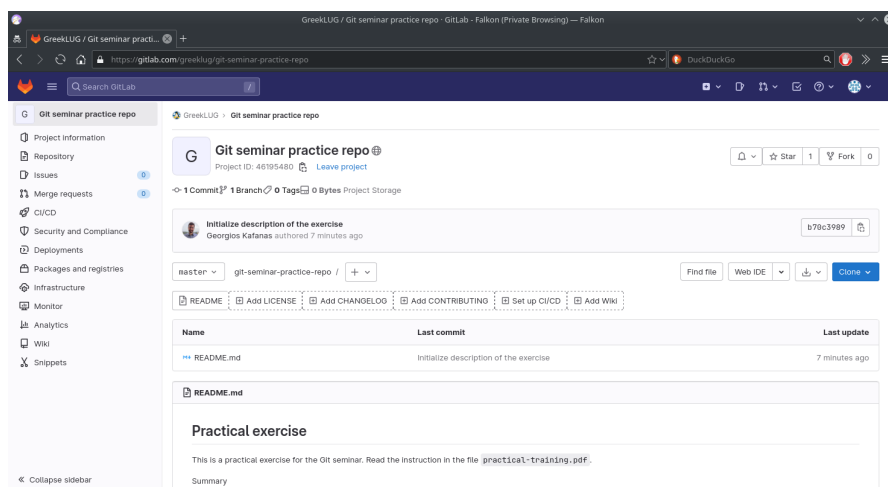
1. κατευθυνθείτε στην **σελίδα του GitLab όπου φιλοξενείται το αποθετήριο της άσκησης** και ζητήστε πρόσβαση,



2. περιμένετε να εγκρίνει το αίτημά σας κάποιος από τους διαχειριστές του αποθετηρίου,



3. και μετά την έγκριση της αίτησής σας,



είστε έτοιμοι να συνεισφέρετε τον κώδικά σας!

## 4 Εργασία με πολλαπλούς κλάδους

Ο σκοπός της άσκησης αυτής είναι να δουλέψει ο καθένας σε ένα τοπικό αποθετήριο, και στη συνέχεια να ανεβάσει τη δουλειά του σε ένα κεντρικό αποθετήριο, σε ένα κλάδο που θα είναι εύκολα αναγνωρίσιμος από τους άλλους χρήστες του αποθετηρίου.

Αρχικά, δημιουργήστε ένα τοπικό αποθετήριο με την εντολή

```
git init git-seminar-practice-repo
```

και μεταβείτε στο φάκελο του αποθετηρίου. Καταγράψτε σε ένα αρχείο της ορισμένες οδηγίες για τη ρύθμιση του Git ή οτιδήποτε άλλο σας ενδιαφέρει. Το ιστορικό από το παράδειγμα των διαλέξεων μπορείτε να το βρείτε στην fig. 1. Στο παράδειγμα υπάρχουν 2 κλάδοι ανάπτυξης, master και configuration. Τα commits C & F έγιναν στον κλάδο configuration, κι όλα τα υπόλοιπα στο κλάδο master.

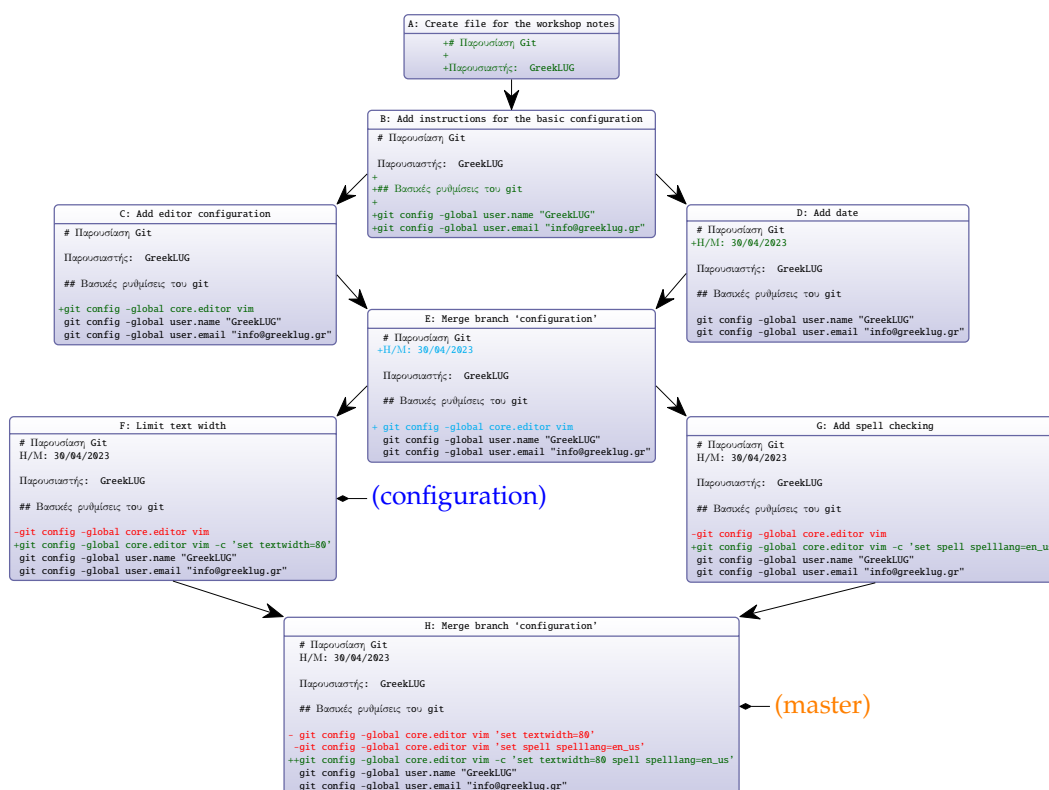


Figure 1: Τελική κατάσταση του αποθετηρίου της άσκησης

Όταν ολοκληρώσετε τη δουλειά σας στο τοπικό αποθετήριο, είστε έτοιμοι να ανεβάσετε τις αλλαγές σας στο κοινό δικτυακό αποθετήριο.

## 5 Συγχρονισμό με απομακρυσμένο αποθετήριο

Οι αλλαγές στο τοπικό αποθετήριο τώρα θα συγχρονιστούν με το δικτυακό αποθετήριο ώστε να είναι διαθέσιμες και σε άλλους χρήστες. Προσθέστε το δικτυακό αποθετήριο της άσκησης στα απομακρυσμένα του τοπικού αποθετηρίου `git-seminar-practice-repo`, που δημιουργήσατε στη section 4, με την εντολή:

```
git remote add origin git@gitlab.com:greeklug/git-seminar-practice-repo.git
```

Στη συνέχεια, ανεβάστε τους κλάδους `master` & `configuration` του τοπικού σας αποθετηρίου στο δικτυακό αποθετήριο. Στους κλάδους που ανεβάζετε προσθέστε το όνομα χρήστη σας ακολουθούμενο

από κάθετο (/) ως πρόθεμα. Για παράδειγμα ο χρήστης `gkaf` θα ανεβάσει στο δικτυακό αποθετήριο τους κλάδους:

```
master → gkaf/master (1)
```

```
configuration → gkaf/configuration (2)
```

Για να ανεβάσετε ένα κλάδο **branch-A** του τοπικού αποθετηρίου σε ένα κλάδο **branch-B** του απομακρυσμένου αποθετηρίου **origin**, χρησιμοποιείτε την εντολή:

```
git push origin branch-A:branch-B
```

Αφού ανεβάσετε τους νέους κλάδους στο απομακρυσμένο αποθετήριο του GitLab μπορείτε να δείτε τη δουλειά σας με στο δικτυακό περιβάλλον που παρέχει το GitLab.

Γενικά είναι καλή πρακτική να χρησιμοποιείτε καθέτους, όπως στα ονόματα φακέλων, για να διαχωρίζετε τους κλάδους σε κατηγορίες. Για παράδειγμα:

- `bug-fix/numerical-zero-in-matrix-inversion`
- `new-feature/support-matrix-multiplication`

Αποφεύγετε μόνο το πρόθεμα **remotes/** γιατί είναι δεσμευμένο για να αποθηκεύει το Git τους κλάδους απομακρυσμένων αποθετηρίων που φορτώνει τοπικά με την εντολή `fetch`.

Μπορείτε να κατεβάσετε όλους τους άλλους κλάδους που υπάρχουν στο απομακρυσμένο αποθετήριο **origin** με την εντολή:

```
git fetch origin
```

Για να κατεβάσετε ένα μόνο κλάδο **branch-A** από το απομακρυσμένο αποθετήριο, χρησιμοποιείτε την εντολή

```
git fetch origin branch-A
```

και ο κλάδος θα αντιγραφεί στον κλάδο **origin/branch-A** του τοπικού αποθετηρίου.

Να σημειώσουμε εδώ ότι οι κλάδοι που ανεβάζει ο κάθε χρήστης σε συνδέονται στο γράφημα της ιστορίας με τον **master** κλάδο του δικτυακού αποθετηρίου ή τους κλάδους που ανεβάζουν οι άλλοι χρήστες. Αυτό είναι απόλυτα φυσιολογικό από το τρόπο με τον οποίο είναι δομημένο το δικτυακό αποθετήριο. Κλάδοι χωρίς κοινό πρόγονο στο ιστορικό τους δεν μπορούν να συγχωνευτούν με την εντολή `merge`! Αν το προσπαθήσετε θα λάβετε κάποιο μήνυμα λάθους, όπως:

```
fatal: refusing to merge unrelated histories
```

Κλάδοι που δε συνδέονται με τον **master** ονομάζονται κι ορφανοί. Ο κλάδος **master** δεν έχει κάποια ιδιαίτερη σημασία, απλά είναι το όνομα με το οποίο το Git ονομάζει το κλάδο στον οποίο ανοίγει κάθε νέο αποθετήριο. Το όνομα του **master** μπορεί να αλλάξει στις ρυθμίσεις του Git.