# Git

# Part 2

By: Ali Moeinian

# 2 – كورس فرادرس

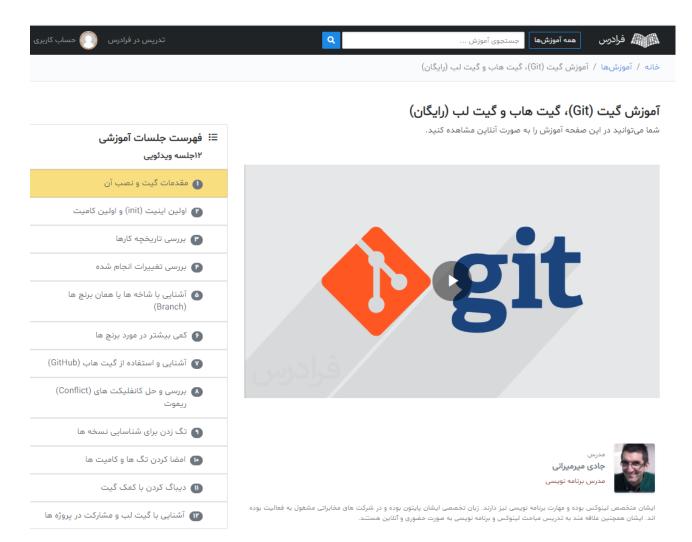


و حالا وقتشه که بریم سراغ کسی که من خیلی دوسش دارم 💙 ؼ

و اما دوره ای که در نظر گرفتم دوره ای نیست بجززززززززززززززززز دوره ی گیت جادی عزیییییز که من به شخصه خیلی دوسش دارم 💝 👺

اما نکته ی مهم اینه که اگر مطلب جدید یا توضیح جدید توی این دوره بود رو من بیان میکنم.

این دوره رو به طور رایگان میتونید از سایت فرادرس تهیه کنید.



# پاز هم تکرار میکنم که از نوشتن مطالب تکراری خودداری میکنم

### جلسه 1

ابتدای این جلسه جادی بیان میکنه که گیت رو واقعا هر برنامه نویسی الزام داره تا بلد باشه و نکته ی جالبش اینجاست که فقط برای برنامه نویس ها نیست و به در د طبف بیشتری از آدما میخوره؛ کسایی که پایان نامه مینویسن، یه فایلی رو در یک بازه ی طولانی دارن تهیه میکنن و .....

در ادامه جادی بیان میکنه که گیت خیلی به ما کمک میکنه تا وقتی دایر کتوری های زیادی رو در یک پروژه داریم، بتونیم اونا رو بهتر مدیریت کنیم.

#### جلسه 2

توی این جلسه بدون اشاره به نحوه ی نصب گیت و ... سریعا یک پوشه رو با دستور مربوطه (init) تحت گیت میکنیم.

✓ توی لینوکس و مک هر فایلی با . شروع بشه (مثلgit) و باید بدونیم که این نوع فایل ها، مخفی هستند.

✓ وقتی فایل های داخل دایر کتوری رو تغییر میدید اونها رو وارد حالت stage
 میکنید که در نهایت اماده میشن برای commit شدن.



✓ اگر چندین فایل در حالت staging داشتید نیازی نیست یکی یکی اون ها رو add کنید، بلکه میتونید از دستور Add و git add استفاده کنید تا همه ی فایل هاتون Add بشه و در نهایت اونها رو کامیت کنید.

## جلسه 3

مرور مطالب قبلی و کار کردن با git status که دیگه بلدیم 😃

### جلسه 4

✓ یه دستور جدید باحال برای فایل هایی که در مرحله ی staging هستند:
 git diff -staged

✓ برای اینکه فایلی به نام مثلا x رو از مرحله ی staging در بیارید ( به هر دلیل) میتونید از دستور زیر استفاده کنید :

git reset x

✓ اگر یه تغییری رو دادید ولی خب دیگه به نظرتون نیازی نیست، و باید اون فایل
 رو به کامیت قبلیش برگر دونید، میتونید از دستور زیر استفاده کنید:

Git checkout --x

#### جلسه 5

توضیحات کلی در ارتباط با branch و مفهوم برنچ و merge کردن و تمریناتی رو به سرعت انجام میده که خودمون بالاتر خیلی دقیق ترش رو انجام دادیم.

#### جلسه 6

✓ مهم ترین مفهومی که توی این جلسه بهش اشاره شد، اینه که به عنوان یک گیت
 کار، هیچوقت کل پروژتون رو روی master نبرید.

و همیشه اگر تغییراتی رو نیاز دارید به پروژتون بدید یا یک فیچر اضافه کنید، همیشه به عنوان برنچ به پروژه اضافه کنید.

✓ برای حذف کردن یک فایل (مثلا ۲) از کل ریپوزیتوری تحت گیت و هم از فایل سیستم ، میتونید از دستور زیر استفاده کنید :

git rm Y

#### جلسه 7

✓ اغاز کار با گیت هاب که خیلی از مواردش رو کار کردیم و تکراریه.

### جلسه 8

✓ توی این قسمت درباره ی conflict کار شده و بحث شد که چطور میتونیم این تضاد به وجود امده در پروژه رو حل کنیم کخ به طور مفصل و کامل توضیح دادم براتون.

## جلسه 9

✓ بحث تگ زدن روی کامیت ها بیان شد

#### جلسه 10

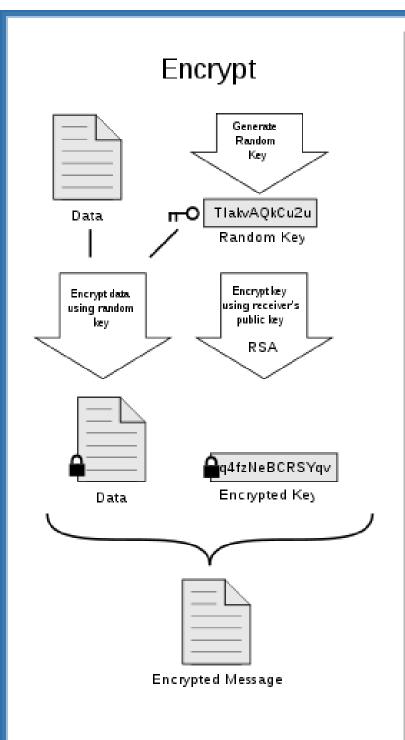
✓ بحث جدید اور دم براتون (ن)

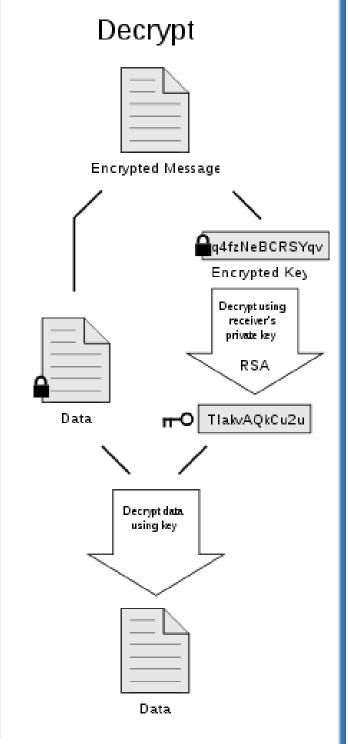
استفاده از امضا برای امنیت خیلی بیشتر یک ریپوزیتوری یا یک کامیت:

# 7.4 Git Tools - Signing Your Work

#### Signing Your Work

Git is cryptographically secure, but it's not foolproof. If you're taking work from others on the internet and want to verify that commits are actually from a trusted source, Git has a few ways to sign and verify work using GPG.





پی جی پی (PGP)، جزء اولین نرم افزارهایی است که به طور گسترده به منظور کاربرد پیاده سازی رمزنگاری کلید عمومی در دسترس است. یک نوع رمزنگاری ترکیبی است که از رمزنگاری متقارن و نامتقارن برای دستیابی به سطح بالایی از امنیت استفاده میکند.

در یک فرآیند اساسی رمزگذاری متن، یک متن ساده (دادههایی که به روشنی قابل درک است) به متن رمزنگاری (دادههای غیرقابل خواندن) تبدیل میشود. اما قبل از اینکه فرآیند رمزگذاری صورت بگیرد، بیشتر سیستمهای پی جی پی فرآیند فشرده سازی دادهها را انجام میدهند. با فشرده سازی پروندههای متون ساده قبل از انتقال آنها، پی جی پی هم فضای اضافی دیسک را افزایش میدهد، هم زمان انتقال را کاهش میدهد و از همه مهمتر، امنیت را نیز بهبود میبخشد.

پس از فشرده سازی فایل، روند واقعی رمزگذاری شروع میشود. در این مرحله، فایل اولیه فشرده شده با یک کلید یکبار مصرف رمزگذاری میشود، که به عنوان کلید جلسه شناخته میشود. این کلید به طور تصادفی از طریق استفاده از رمزنگاری متقارن تولید میشود و هر دورهی ارتباطی پی جی پی دارای یک کلید منحصر به فرد است.

نرم افزار آزادیا و کسایی که این کاره اند، او مدن و gpg رو ساختن ش که اجازه میده شما امضای خودتون رو یای کاراتون بزنید که بر اساس کلید ها کار میکنه.

```
→ bestoon git:(master) gpg --list-keys
gpg: directory '/Users/jadi/.gnupg' created
gpg: keybox '/Users/jadi/.gnupg/pubring.kbx' created
gpg: /Users/jadi/.gnupg/trustdb.gpg: trustdb created
→ bestoon git:(master) gpg --gen-key
gpg (GnuPG) 2.1.21; Copyright (C) 2017 Free Software Foundation, Inc.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Note: Use "gpg2 --full-generate-key" for a full featured key generation dialog.
GnuPG needs to construct a user ID to identify your key.

Real name: jadi mirmirani
Email address: jadijadi@gmail.com
You selected this USER-ID:
    "jadi mirmirani <jadijadi@gmail.com>"
Change (N)ame, (E)mail, or (0)kay/(Q)uit? o
```

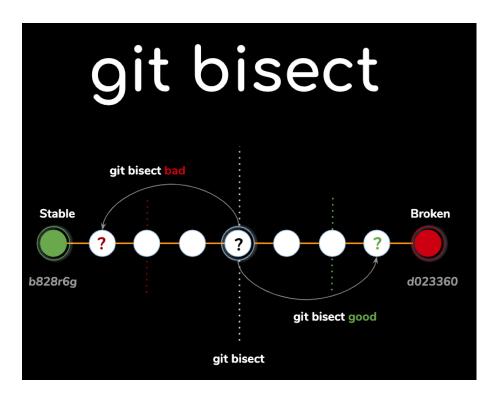
Please enter the passphrase protect your new key	to
Passphrase:	
<0K>	<cancel></cancel>

یه جورایی مثل استفاده از اون دو تا کلید ها میمونه که در کورس اول نام بردم.

جلسه 11

git bisect دستور

git bisect is a tool that allows you to find an offending commit. ... If you can find a commit where the code works properly and a commit where it doesn't, you don't have to trace down the offending commit by hand; git-bisect will do that for you. Khordad 22, 1394 AP



# start up git bisect git bisect start

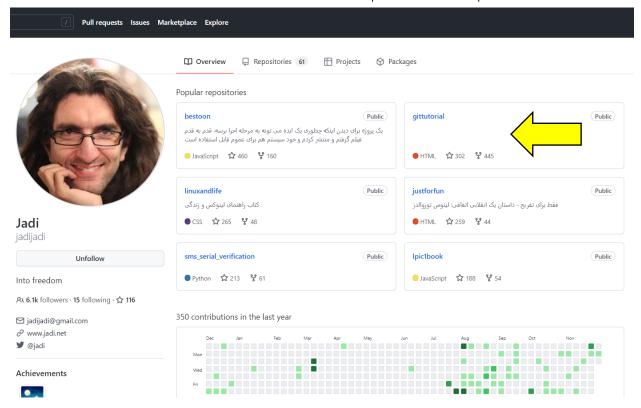
# give git a commit where there is not a bug
git bisect good a09c728

# give git a commit where there is a bug
git bisect bad b6a0692

#### جلسه 12

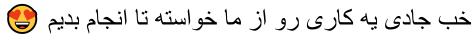
✓در ارتباط با فورک کردن توضیحاتی داده شد.

جادی گفت که یه ریپوزیتوری داره توی گیت هاب خودش که میتونیم خودمون داخلش تغییراتی رو اعمال کنیم؛ وقتشه بریم سراغش.

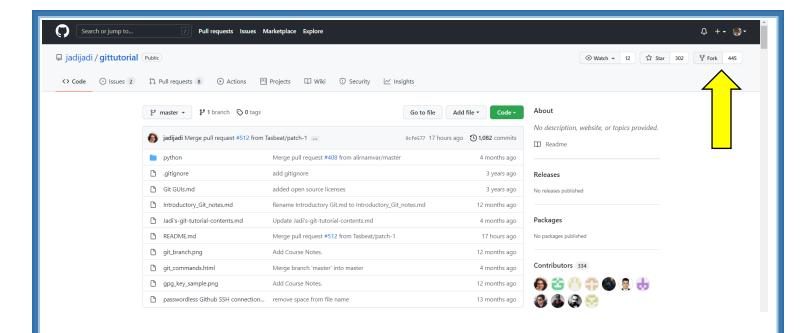


بریم توی این ریپوزیتوری و ببینیم چه کار هایی تاحالا انجام شده یا از ما میخواد تا انجام بدیم.





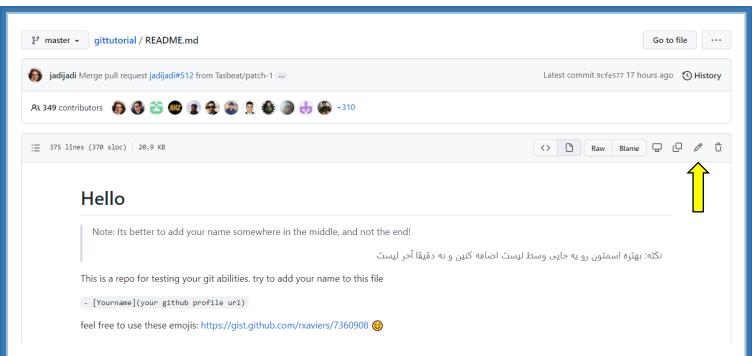
برای اینکه بتونیم تغییراتی رو که جادی گفته انجام بدیم اول از همه باید این ریپوزیتوری رو fork کنیم تا در اکانت خودمون اجازه ی دادن تغییراتی را روی این ریپوزیتوری داشته باشیم.



من به راحتی ریپوزیتوری رو فورک کردم و این هم نتیجه اش:



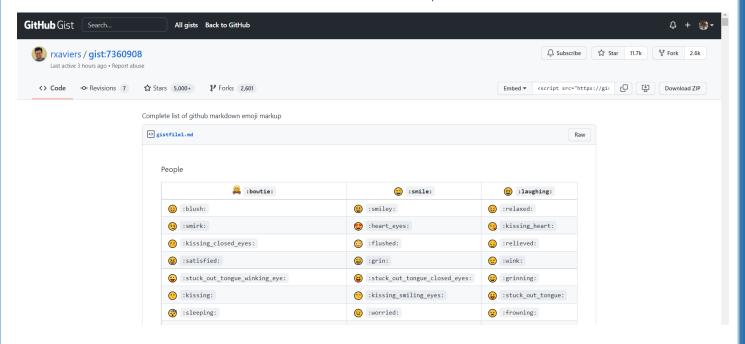
روی فایل README کلیک میکنم تا واردش بشم به علت اینکه برای وارد کردن اسم و لینک گیت هابم، باید وارد این دایرکتوری بشم.



همونطور که مشخص کردم، باید روی ویرایش کلیک کنیم تا بتونیم اسم خودمون رو هم اضافه کنیم.

راستی یادت نره، تا وقتی که ریپوزیتوری رو فورک نکردی، این گزینه ی ویرایش بر ات غیر فعاله.

راستی جادی یه دونه ریپوزیتوری دیگه رو هم برامون مشخص کرده تا بتونیم اموجی ها رو از داخل اون برداریم:



# ولى فعلا بريم سراغ اديت كردن اون فايل و اضافه كردن اسم خودمون:

```
<> Edit file
                Preview
     > Note: Its better to add your name somewhere in the middle, and not the end!
     نکته: بهتره اسمتون رویه جایی وسط لیست اضافه کنین و نه دقیقا آخر لیست <
     This is a repo for testing your git abilities, try to add your name to this file
     `- [Yourname](your github profile url)`
 11 feel free to use these emojis: <a href="https://gist.github.com/rxaviers/7360908">https://gist.github.com/rxaviers/7360908</a> :relaxed:
13 # List of people who know git
    - [Ali Ghasempour =)1(https://github.com/AliGhDeveloper)
    - [Ali Alirahimi](https://github.com/Tasbeat)
     - [Hosein Laki , I love python](https://github.com/hosein-1)
    - [Mobin Rajaei :)](https://github.com/itsdiy0)
    - [Ashkan Shakiba :sunglasses::sparkles:](https://github.com/AshkanShakiba)
    - [Hossein Gholamian :trollface::+1:](https://github.com/HosseinGholamian)
    - [Hesam Omidi :notes:](https://github.com/Tannisam)

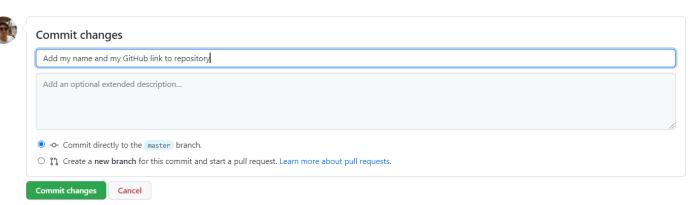
    - [fateme barbod√@](https://github.com/fatemebarbod)

    - [masome papi [] (https://github.com/papimasome)
     - [Hossein-Developer :wolf:](https://github.com/Hossein-Haghshenas)
    - [MicRoB ♥](https://github.com/ALI161)
     - [Amir.H.Shirzad:sunglasses::computer:](https://github.com/Amir-H-Shzd)
    - [ZaraKh :computer: ](https://github.com/Zarakhansari)
    - [mohamad hosein](https://github.com/mmdtoorani)
    - [DavodBeheshti:heart::sunglasses::heart:](https://github.com/davodbeheshti)
    - [Alireza Ebrahimkhani :grin:](https://github.com/AlirezaEbrahimkhani)
     - [Mohrazzr:computer::heart:](https://github.com/mohrazzr)
    - [Jadi](https://jadi.net)
     - [Nima, Linux Lover :)](https://smartnima.com)
    - [Javad Salehi 🕲 🔥](https://github.com/JSalehii)
    - [Ehsan Saadatfar](https://github.com/ehsansdf)
    - [Hedieh Raeisi :blue_heart:](https://github.com/HediyeRaisy)
```

وقتی روی علامن ادیت میزنید، فایلتون اینطوری میشه و تا و قتشه تا اسمم رو اضافه کنم:

- Hossein Gholamian :trolltace::+1: [(https://github.com/HosseinGholamian)
- [Ali Moeinian](https://github.com/AliMoeinian)
- [Hesam Omidi :notes:](https://github.com/Tannisam)

# در پایین صفحه هم کامیت مناسب رو نوشتم:



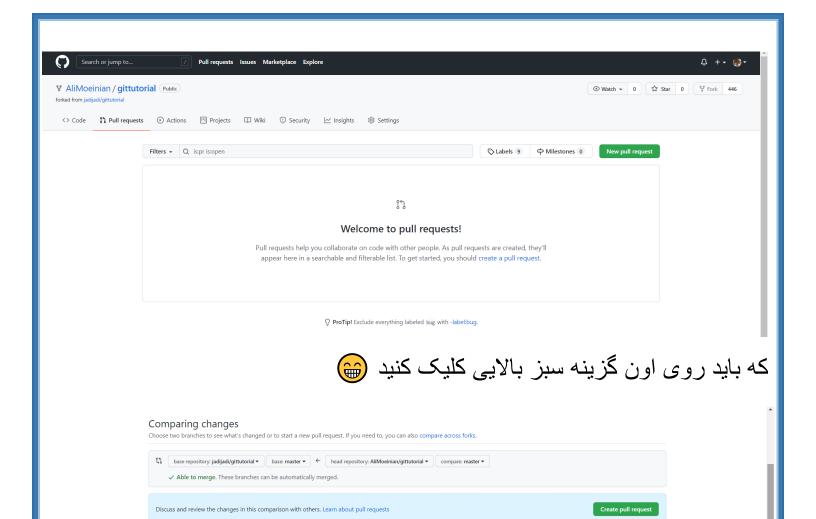
و در نهایت گزینه ی commit changes رو میزنم.

اما خب الان تغییری که ایجاد شده صرفا روی ریپوزیتوری خودمه؛ پس باید چیکار کنم تا این تغییرات روی گیت هاب جادی هم اضافه بشه؟ (یعنی میخوام تغییری که خودم دادم رو در ریپوزیتوری اصلی هم اعمال کنم)

باید pull request بکنیم.



وقتی روی این گزینه کلیک کنید، با این صفحه مواجه خواهید شد:



1 file changed

Add my name and my GitHub link to repository

AliMoeinian committed 3 minutes ago

.... @@ -17,6 +17,7 @@ feel free to use these emojis: https://gist.github.com/rxaviers/7360908 :relaxed

- [Hossein Gholamian :trollface::+1:](https://github.com/HosseinGholamian)

🖹 Showing 1 changed file with 1 addition and 0 deletions.

- [Mobin Rajaei :)](https://github.com/itsdiy0)

[Hesam Omidi :notes:](https://github.com/Tannisam)

- [fateme barbod√⊚](https://github.com/fatemebarbod)

- [masome papi@](https://github.com/papimasome)

✓ 💠 1 ■ README.md 📮

21

و در آخر هم باید روی create pull request بزنید.

- [Ashkan Shakiba :sunglasses::sparkles:](https://github.com/AshkanShakiba)

- [Hossein Gholamian :trollface::+1:](https://github.com/HosseinGholamian)

- [Mobin Rajaei :)](https://github.com/itsdiy0)

- [Hesam Omidi :notes:](https://github.com/Tannisam)

- [masome papi@](https://github.com/papimasome)

- [fateme\_barbody@](https://github.com/fatemebarbod)

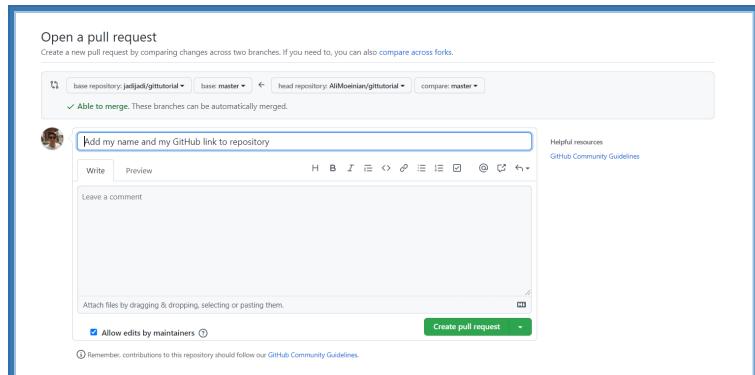
20 + - [Ali Moeinian](https://github.com/AliMoeinian)

At 1 contributor

Verified ☐ 3eb50f7 〈> ▲ Tip

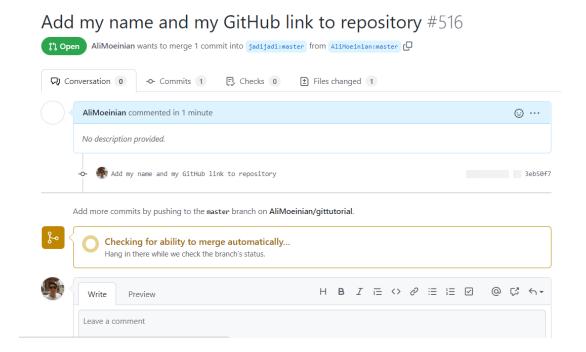
Split Unified

<> □ …

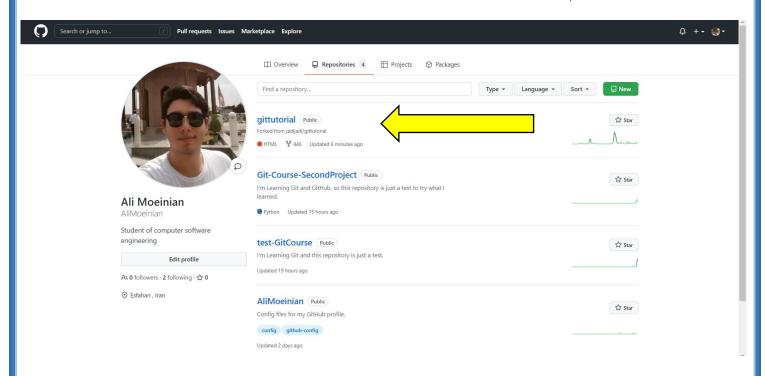


و در اخر : create pull request

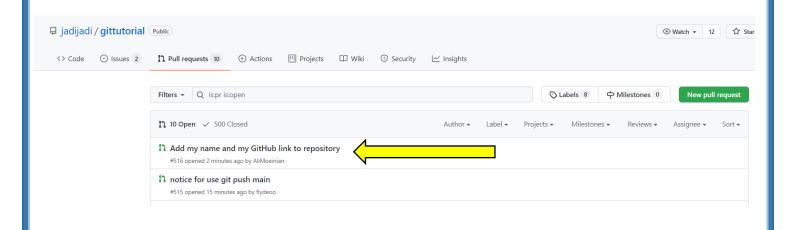
این هم نتیجه ی کار:



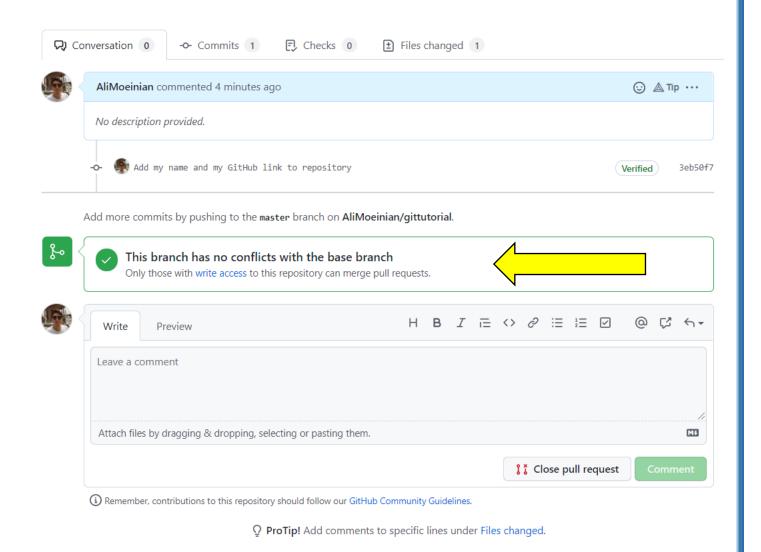
# خب الان بر میگردیم به اکانت گیت هاب خودمون :



حالا اگر برگردیم به ریپوزیتوری اصلی خود جادی و بریم توی قسمت .pull req متیونیم ببینیم که درخواست ما اونجا هست و کارمون رو به درستی انجام دادیم :



اگر روی پول ریکوئست کلیک کنید میتونید اطلاعات جالبی رو ببینید:



# خب چند روز گذشت از درخواست پول ریکوئست و نتیجه اش به ایمیلم اومد:

Re: [jadijadi/gittutorial] Add my name and my GitHub link to repository (PR #516) > Inbox ×

**Jadi** <notifications@github.com> <u>Unsubscribe</u> to jadijadi/gittutorial, me, Author ▼

Merged #516 into master.

You are receiving this because you authored the thread.

Reply to this email directly, view it on GitHub, or unsubscribe.

Triage notifications on the go with GitHub Mobile for iOS or Android.

# بریم ریپوزیتوری اصلی رو چک کنیم و خودمون رو پیدا کنیم:

