

:git

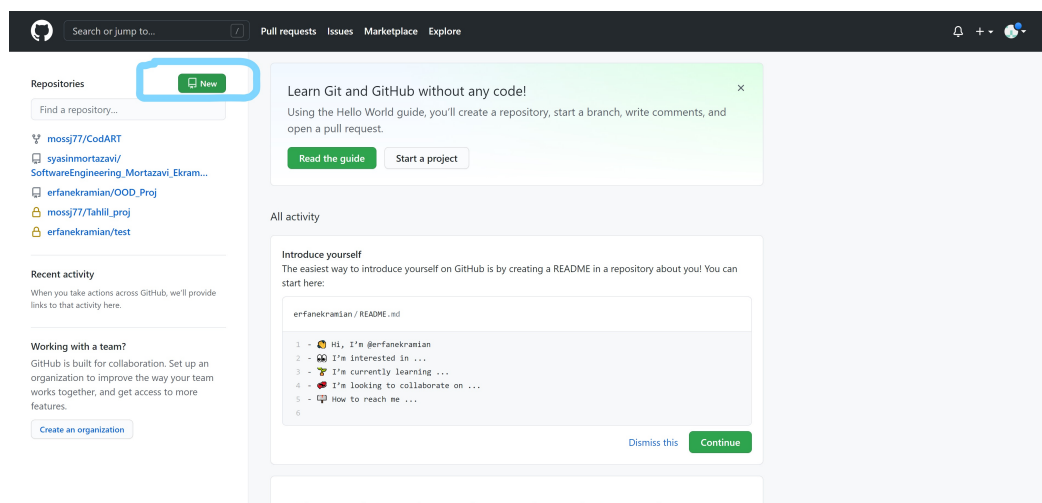
git ابزاری است که به وسیله آن میتوانیم ورژن های مختلف یک پروژه را کنترل کنیم. git وظیفه ردیابی تغییراتی که در فایل های پروژه رخ میدهد و همچنین وظیفه هماهنگ سازی کار بر روی فایل های مختلف درون پروژه برای چندین نفر را بر عهده دارد. git این امکان را برای ما فراهم میکند که نسخه های مختلفی از یک پروژه داشته باشیم و به راحتی بتوانیم نسخه های قبلی را برگردانیم. git امروزه به شدت میان برنامه نویسان و توسعه دهندگان محبوب است و معمولاً در پروژه هایی که بیش از یک نفر آن را توسعه میدهد استفاده میشود.

:repository

برای مدیریت پروژه در git ما نیاز به یک repository داریم. فایل های پروژه درون repository قرار میگیرند و همچنین فایل هایی که در حین پروژه تغییر میکنند یا به پروژه اضافه میشوند همه درون آن قرار دارند.

نحوه ساخت repository:

برای ساخت repository ابتدا باید در سایت www.github.com ثبت نام کنیم. پس از ثبت نام و ورود به سایت میتوانیم از طریق صفحه پروفایل خود به ساخت repository اقدام کنیم. برای این کار مانند شکل زیر در پروفایل خود از قسمت repositories گزینه new را انتخاب میکنیم.



پس از آن صفحه برای ما نمایش داده خواهد شد. در آن صفحه باید اطلاعات مربوط به repository را وارد کنیم.

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner *



Repository name *

typora_git

Great repository names are short and memorable. Need inspiration? How about [solid-garbanzo?](#)

Description (optional)



Public

Anyone on the internet can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.

Initialize this repository with:

Skip this step if you're importing an existing repository.

☐ Add a README file

This is where you can write a long description for your project. [Learn more.](#)

☐ Add .gitignore

Choose which files not to track from a list of templates. [Learn more.](#)

☐ Choose a license

A license tells others what they can and can't do with your code. [Learn more.](#)

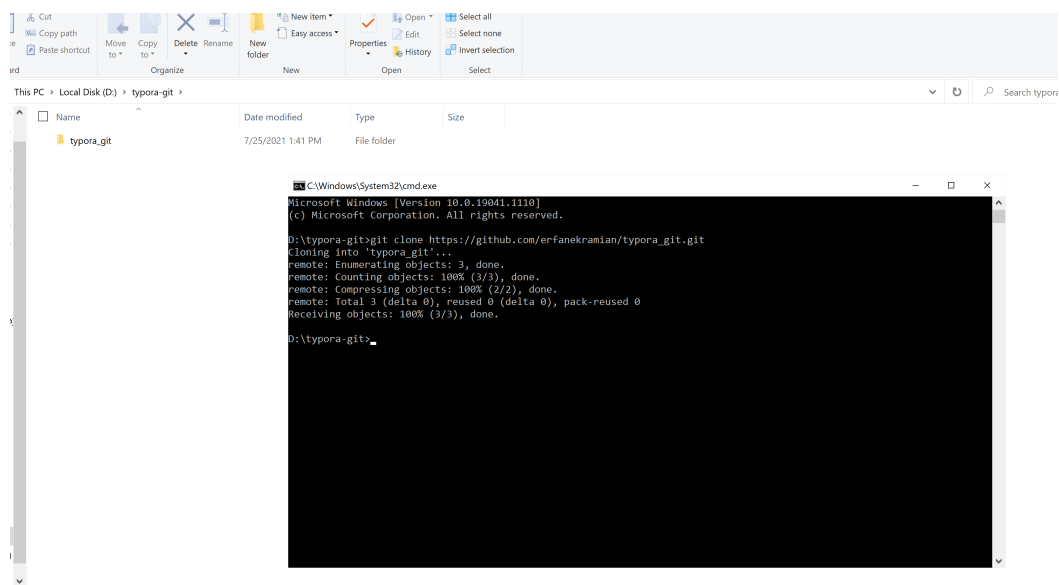
Create repository

پس از وارد کردن اطلاعات مربوط گزینه create repository را انتخاب میکنیم و repository برای ما ساخته میشود.

من در repository ساخته شده فایل مربوط به گزارش typora را آپلود کردم.

حال برای ایجاد کردن repository مورد نظر در کامپیوتر خود از قسمت code لینک repository را کپی میکنیم. پس از آن command prompt را اجرا کرده و به مسیری میرویم که میخواهیم repository را به آنجا منتقل کنیم.

برای اینکار پس از دستور git clone لینکی را که کپی کرده بودیم قرار میدهیم و دستور را اجرا میکنیم. پس از اجرای دستور repository با موفقیت در مسیر مربوطه ایجاد میشود.



توضیح برخی از دستورات ساده git:

ساخت branch:

زمانی که repository را ایجاد میکنیم یک شاخه (branch) اصلی به نام master وجود دارد. برای اینکه تغییرات مختلفی که در فایل ها ایجاد میکنیم روی این شاخه اصلی اعمال نشوند میتوانیم شاخه جدیدی ایجاد کنیم و در درون آن شاخه به صورت مستقل به توسعه بپردازیم. در انتها فایل های نهایی که خالی از خطا هستند را باید در شاخه master ادغام کنیم. برای ایجاد یک شاخه جدید از دستور `git checkout -b` استفاده میکنیم و پس از این دستور نام شاخه ای که میخواهیم ایجاد کنیم را وارد میکنیم و دستور را اجرا میکنیم.

برای مثال با دستور `git checkout -b typora-pdf` یک شاخه جدید به نام `typora-pdf` ایجاد میشود.

کامیت کردن:

پس از اینکه تغییراتی روی فایل ها ایجاد کردیم باید برای آن ها یک پیام کامیت را وارد کنیم. هدف از نوشتن این پیام این است که برای افراد دیگر پروژه مشخص کنیم که چه تغییراتی ایجاد کرده ایم.

به وسیله دستور `"git commit -m"` میتوانیم پیام خود را بنویسیم. در این دستور باید پیام خود را در آخر و بین دو علامت " بنویسیم.

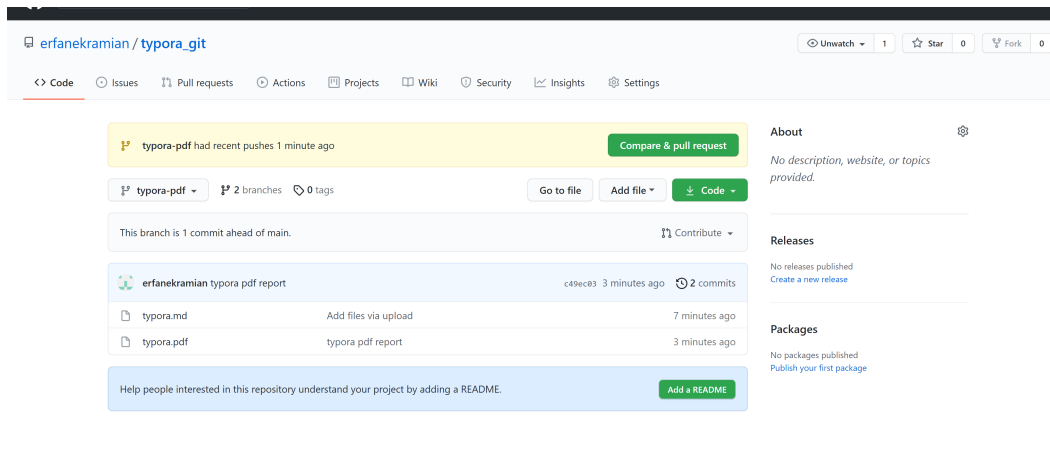
دستور push:

برای اینکه تغییراتی که به صورت محلی روی فایل ها ایجاد کردیم را روی repository آنلاین بفرستیم و ذخیره کنیم از دستور `git push` استفاده میکنیم. برای مثال در تصویر زیر تغییراتی که به صورت محلی ایجاد کرده بودیم را بر شاخه `typora-pdf` در repository آنلاین اعمال میکنیم.

```
D:\typora-git\typora_git>git push origin typora-pdf
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 475.15 KiB | 7.54 MiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'typora-pdf' on GitHub by visiting:
remote:   https://github.com/erfanekramian/typora_git/pull/new/typora-pdf
remote:
To https://github.com/erfanekramian/typora_git.git
 * [new branch]      typora-pdf -> typora-pdf

D:\typora-git\typora_git>
```

حال اگر repository را مشاهده کنیم میبینیم که تغییرات مربوطه به درستی اعمال شده است.



The screenshot shows the GitHub interface for the repository 'erfanekramian/typora_git'. The main content area displays the 'typora-pdf' branch, which is 1 commit ahead of the 'main' branch. It lists two commits: 'erfanekramian typora pdf report' and 'typora.pdf'. The repository has 1 star and 0 forks. The right sidebar shows 'About', 'Releases', and 'Packages' sections.

دستور pull:

دستور pull از نظر کاربرد عکس دستور push عمل میکند. این دستور تغییراتی که در repository آنلاین وجود داشته را روی پروژه که در کامپیوتر خود به صورت محلی ذخیره کرده ایم اعمال میکند.