

## Git

مراحل نصب و اجرا

پوریا شریعت زاده

(۱) ابتدا باید نرم افزار گیت را بر روی سرور و کلاینت ها نصب کرده مراحل نصب را به صورت پیش فرض رد کنید تا گیت به صورت کامل بر روی سیستم شما نصب گردد.

---

(۲) در مرحله بعدی نوبت به تعریف یوزر است

Git bash را باز نموده ، و دستورات زیر را وارد کرده:

→ Git config –global user.name “your name”

→ Git config –global user.email “your Email”

برای اینکه نام کاربری یا ایمیل جاری را ببینید کافی ست دستورات بالا را بدون "..." وارد کنید. مانند:

→ Git config –global user.name

---

نکته:

( در صورتی که یک کاربر هستید و فقط قصد دارید به یک repository متصل بشید مرحله سوم را رد کنید)

(۳) در مرحله بعدی نوبت به ایجاد یک repository خالی برای قرار دادن پروژه هست که این repository باید در server باشد . برای مثال پوشه ای را برای این کار در آدرس زیر در نظر می گیریم :

D:\Projects

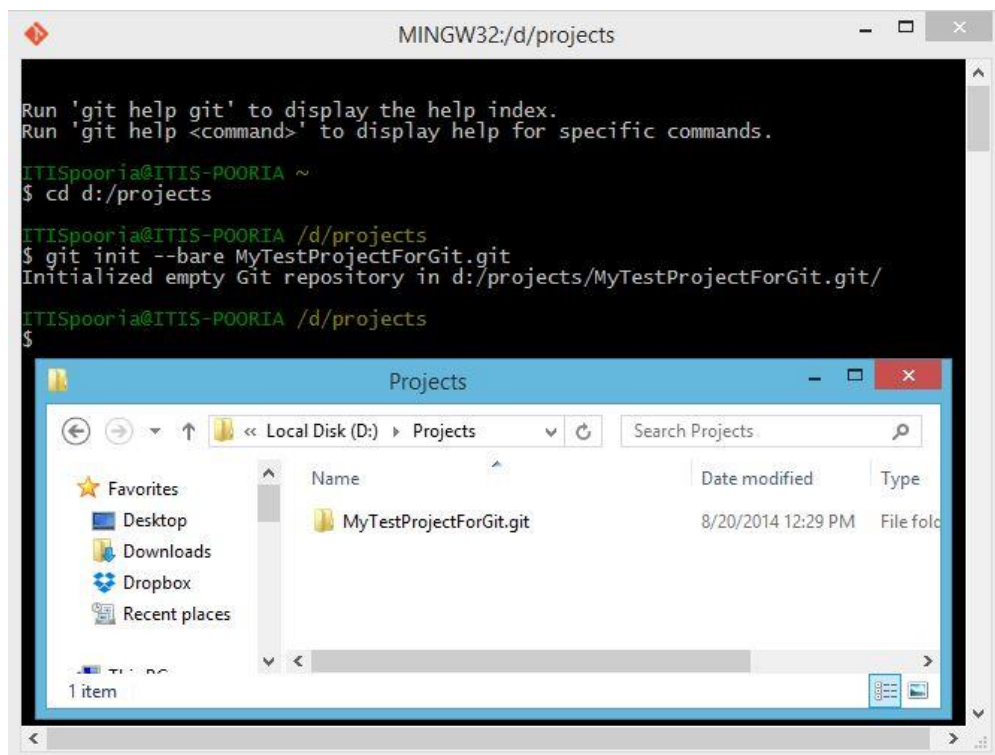
(این پوشه را شیر کرده تا در مراحل بعدی از آن استفاده کنیم)

Git bash را در server باز نموده ابتدا به محل مورد نظر رفته ، و سپس دستورات زیر را وارد کرده:

→ Cd D:\projects

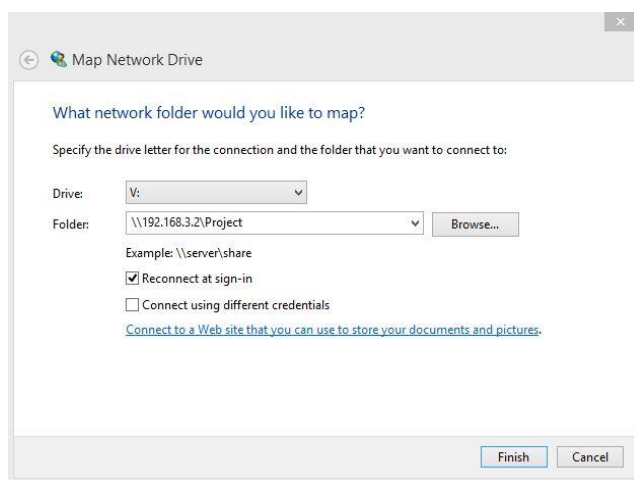
→ git init –bare MyTestProjectForGit.git

نتیجه باید به این صورت باشد:



۴) در مرحله بعدی نوبت به کلاينتها يا يوزر هايي است که در حال کاربر روی پروژه هستند پس اگر شما کاربری هستید که فقط نیاز به وصل شدن به یک repository را دارید قسمت ۳ را رد کرده و مستقیم از قسمت ۴ ادامه ماجرا رو دنبال کنید .

در این مرحله در ابتدا پوشه‌ای را که در مرحله ۳ بر روی server شیر شده بود را برای خود map کرده برای مپ کردن کافی ست بر روی my computer راست کلیک کرده و گزینه map network drive... را انتخاب کرده آدرس فولدر شیر را مانند مثال زیر وارد کرده و بر روی finish کلیک کرده تا فولدر موردنظر برای شما map شود .

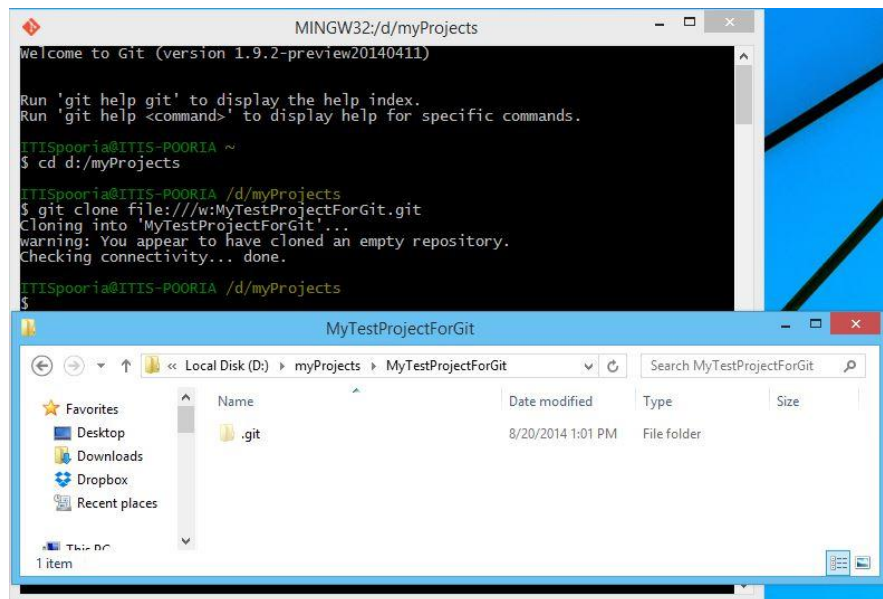


سپس git bash را باز نموده ، به پوشه‌ای که می‌خواهیم پروژه خود را در آن قرار دهیم می‌رویم ، و دستورات زیر را وارد کرده:

➔Cd D:\myProjects

➔Git clone file:///v:/Project/ MyTestProjectForGit.git

حاصل باید شبیه به عکس زیر باشد



۵) در مرحله آخر پروژه را به فولدری که پوشه git در آن قرار دارد اضافه کنید (توجه داشته باشید که در صورتی که پروژه شما فولدرهایی مانند idea. که توسط محیطهای IDE ساخته می شود داشته باشد ممکن است در زمان commit کردن به مشکل برخورید).

تا به اینجا شما موفق شدید تا یک clone از repository موجود بسازید .

- در مراحل بعدی با سه مرحله ADD , Commit , Push or Pull آشنا می شوید.

اما قبل از اینکه وارد مرحله بعدی بشویم اجازه بدید بیشتر با مفهوم این لغات در git آشنا بشیم.

#### Add:

این دستور برای اضافه کردن فایل‌های پروژه به گیت استفاده میشود

#### Commit:

بعد از آن که برنامه نویسان از صحت کدهای خود مطمئن شدند، برای ثبت وضعیت فعلی باید آن‌ها را commit کنند. با این کار یک نسخه جدید از فایل‌ها ایجاد می‌شود. به این ترتیب امکان بازگشت به نقطه فعلی در آینده به وجود خواهد آمد.

#### Pushing:

بعد از انجام عملیات Commit ، معمولاً برنامه نویسان می‌خواهند کدهای نوشته‌شده را با دیگران به اشتراک بگذارند. این کار به وسیله عملیات Pushing صورت می‌گیرد. بنابراین عبارت است از عملی که با استفاده از آن داده‌ها از یک Repository به Repository دیگر جهت به اشتراک گذاری انتقال می‌یابد. معمولاً به این مخزن Upstream Repository می‌گویند Upstream Repository. یک مخزن عمومی برای تمامی برنامه‌نویسانی است که تغییرات فایل‌های خود را در آنجا push می‌کنند.

#### Pulling:

عملیات Pushing تنها نیمی از آن چیزی است که برنامه نویسان برای حفظ به‌روز بودن کدهای خود به آن احتیاج دارند. در بسیاری از موارد آن‌ها نیاز دارند تا تغییرات فایل‌ها و آخرین به‌روزرسانی‌ها را نیز دریافت کنند. این کار در دو مرحله متفاوت انجام می‌شود:

(1) بازیابی داده‌ها از مخزن عمومی (fetch)

(2) الحاق داده‌های دریافت شده با داده‌های فعلی

معمولاً در بسیاری از سیستم‌های مدیریت کد، چون به هردوی این عملیات توأمان نیاز است، با یک دستور هر دو کار انجام می‌شود. به مجموع عملیات فوق Pulling گویند.

در IDE های مختلف روش های مختلفی برای ۳ عملکرد بالا هست ولی در نهایت همه این‌ها یک کار را انجام می‌دهند در این داکيومنت از خود نرم‌افزار git یعنی git bash برای این دستورات استفاده می‌کنیم.

کار رو از مرحله آخر یعنی مرحله ۵ ادامه می‌دید تا به اینجای کار تمامی فایل‌ها در جای خود قرار گرفتند اما در ادامه ما نیاز پیدا می‌کنیم که فایلی به نام index.html را که تغییری در آن دادیم را share کنیم تا دیگر افراد گروه هم بتوانند از آن استفاده کنند برای این کار به ترتیب دستورات زیر را در git bash وارد می‌کنیم.

(نیاز به ذکر می‌باشد که فولدري که باید دران باشیم فولدر پدر يا فولدري است که پوشه git. ما در ان قرار دارد )

→ git add index.html

(با زدن این دستور "git add" تمام فایل ها یا فولدر هایی که در فولدر پدر هستند add میشوند)

→ Git commit -m "comment"

(در این قسمت توضیحاتی راجع به تغییرات انجام شده می نویسیم : comment)

→ Git push origin master

```
MINGW32:/d/myprojects/MyTestProjectForGit

ITISpooria@ITIS-POORIA /d/myprojects/MyTestProjectForGit (master)
$ git add index.html

ITISpooria@ITIS-POORIA /d/myprojects/MyTestProjectForGit (master)
$ git commit -m "MyFirstCommit"
[master c68c636] MyFirstCommit
1 file changed, 1 insertion(+)

ITISpooria@ITIS-POORIA /d/myprojects/MyTestProjectForGit (master)
$ git push origin master
Counting objects: 5, done.
Writing objects: 100% (3/3), 238 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To file:///w:/MyTestProjectForGit.git
083d6d3..c68c636 master -> master

ITISpooria@ITIS-POORIA /d/myprojects/MyTestProjectForGit (master)
$
```

و در انتها برای گرفتن آخرین تغییرات در سیستم های دیگر می توان از دستور زیر استفاده کرد

→ Git pull

```
MINGW32:/c/MyTestProjectForGit/mytestprojectforgit

Administrator@WIN-JAE05R2V0Q7 /c/MyTestProjectForGit/mytestprojectforgit (master)
$ git pull
remote: Counting objects: 5, done.
remote: Total 3 (delta 0), reused 0 (delta 0)
Unpacking objects: 100% (3/3), done.
From file:///z:/MyTestProjectForGit
c68c636..c02bc63 master -> origin/master
Updating c68c636..c02bc63
Fast-forward
 index.html | 3 ++-
1 file changed, 2 insertions(+), 1 deletion(-)

Administrator@WIN-JAE05R2V0Q7 /c/MyTestProjectForGit/mytestprojectforgit (master)
$
```