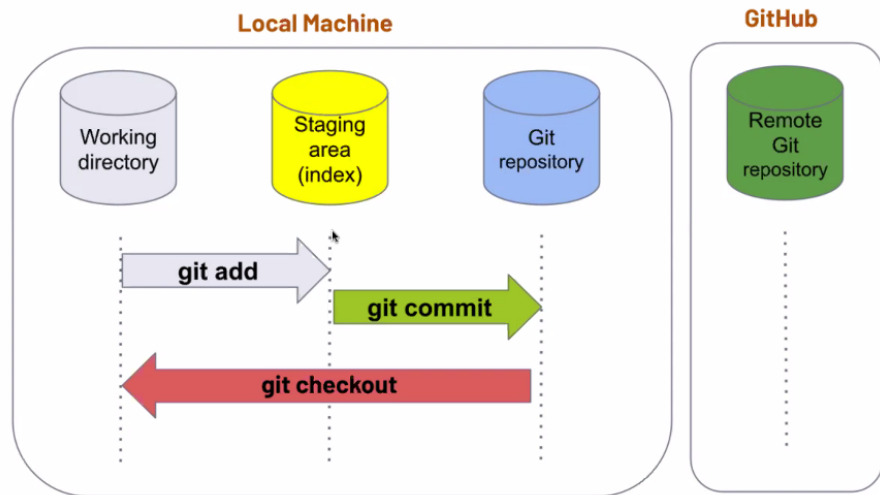


18 March Git/GitHub

git init
git status
git add .
git rm --cached
git commit -m "abc"
git log
git checkout **commitID**



Branch'ler ne için kullanılır:

1. version
2. Birden fazla kişiye çalışma imkanı sunma

git branch -d ⇒ Siliyorduk.

merge yetkisi genelde bir kişidedir.

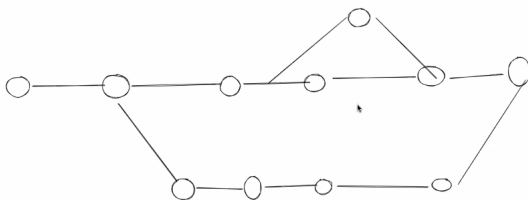
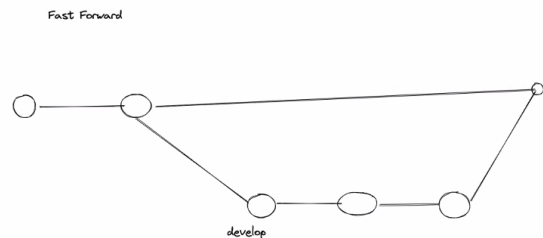
git branch **branch_name**
git branch
git branch -r
git branch -a
git checkout **branch_name**
git checkout -b **branch_name**
git branch -d **branch_name**
git branch -D **branch_name**
git merge **branch_name**

excalidraw.com ⇒ çizim sitesi.

git pull ⇒ en güncelini getirir.

Merge Conflicts ⇒ 2 çeşit.

Fast Forward ⇒ Direk en sona merge edilir. Olması istenilen merge çeşidi.



Fast Forward olmayan ⇒ Birden fazla kişi aynı dosyada farklı çalışmalar yaptığında merge ederken conflicts çıkabilir. Çakışma olabilir. Bu sorunlarını genelde bir kişi senior çözer.

GitHub - Remote Repository

Git, dağıtık bir kontrol sistemini oluşturuyor. GitHub ise merkezi bir sistemde repo oluşturuyor. Her önleme karşı onunda yedeği alınır. GitHub'ı microsoft satın aldı...

Bitbucket GitLab SourceForge GitEnterprise da aynı işlevi yapan başka platformlar.

Şifreyi girmeden önce `git config --global credential.helper store` bunu yazarsak şifre ve kullanıcı adını kaydeder. Bir sonraki işlemlerde istemez.

<https://github.com/marcus-clarusway/Git-Lessons/issues/1>

üstteki siteye bak!!!! oradada yazıyor bu kod.

Profil > Settings > Developer Settings > Personal access tokens > Generate new token

Projects ⇒ Jira tarzı. Öğrenmek faydalı olabilir.

Repositories liste halinde görünüyor.

```
git init
git add .
git commit -m "First commit"
$ git remote add origin <REMOTE_URL>
# Sets the new remote
$ git remote -v
# Verifies the new remote URL
git push -u origin main
# Pushes the changes in your local repository up to the remote repository you specified as
the origin
```

gitignore ⇒ seçilirse bazı dosyaları göndermemek için kullanılır. Bu dosyaları belirleyebiliriz onları tracklamaz ve göndermez.

Silmek için Setting > Danger Zone> Delete this repository

Settings > Repositories > main to master.

```
# Aslında bu komutlar readme dosyasını oluşturmadan repo oluşturduğumuzda bize sunuluyor.B
u ilk yöntem...
...or create a new repository on the command line
```

```
echo "# first-repo" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M master # uzaktaki branch main ise yazılacaktır.
git remote add origin https://github.com/**/first-repo.git # origin takma isim benim repo
nun takma adı...
git push -u origin master # az önce yapmış olduğum commiti gönder.
```

İkinci Yöntem ⇒ clone ile yaparak. ⇒ readme ye ve gitignore a tıklıyoruz. Repo oluşuyor.

git clone repo_linki ⇒ clone u local'e çekiyor. Değişiklik yap. sırasıyla add . sonra git commit -m "we"

sonra git push ile gönderebiliyoruz.

nano .gitconfig ⇒ isim, mail görüntülemek.

[credential]

helper = store ⇒ yazıyorsa bir sonraki işlemlerde şifre istemez.

```
465 git clone https://github.com/***/third-repo.git
466 cd third-repo/
467 ll
468 nano index.html
469 nano deneme.py
470 git add .
471 git commit -m "python and html files were added to repo."
472 git push
473 cd ..
474 ll
475 git clone https://github.com/***/git-session-3.git
476 cd git-session-3/
477 nano index.js
478 nano test.py
479 git add .
480 git commit -m "index.js and test.py were created."
481 git push
482 nano index.html
483 nano README.md
484 git add .
485 git commit -m "index.html created and README.md changed."
486 git push
```

<https://github.com/marcus-clarusway/git-session-3/issues/1>