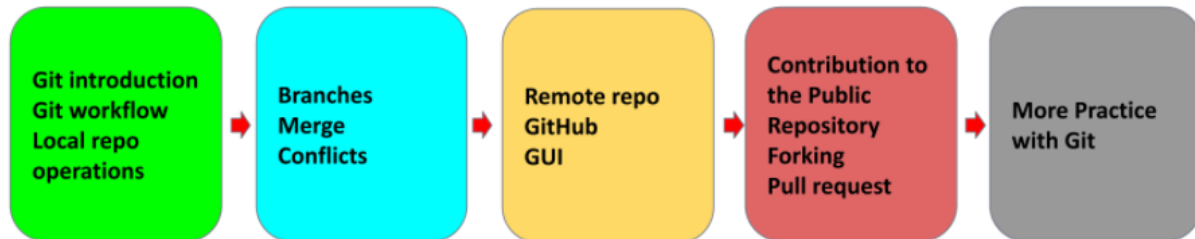


# 05 Mart Git



Git ⇒ proje üzerinde yapılan değişikliği kim ne zaman ne değişiklik yaptı onları takip edebiliriz. Yapılan değişikliklerden önceki versiyona geri dönme imkanı da var. Fazla kişi çalışabilir. Kodu birkez local bilgisayarına indirdikten sonra internet ihtiyacı olmadan çalışılabilir. Open source!!!

Version Control Systems ⇒ Track changes. ne zaman repo oluşturulduysa oraya kadar dönülebilir.

## ► What's Version Control?

- Track changes on text files / source files for you
- Unlimited Undo / Redo
- Time Travel
- Collaborative development environment
- Compare and Blame
  - ◆ What changed
  - ◆ When it changed
  - ◆ Why it changed
  - ◆ Who changed it

Centralized ⇒ merkezi yapıdaki 2kişi aynı anda centrol repository de çalışma imkanı sunmaz. Online olarak merkeze bağlı olması lazım.

Distributed ⇒ birden fazla kişi aynı anda central repository de çalışma imkanı verir. Çalışmaları merkezi bilgisayara gönderir ve oraya yapılan değişikliklerle birlikte kaydedilir. Offline çalışılabilir. İleri veya geri gitme imkanı var.

**Your Daily Tasks** :Create things, Save things, edit things, save the things again  
değişiklikler + ve yeşil ile gösteriliyor. Silinenler kırmızı gösteriliyor.

Git is a software. Linus Torwalds tarafından yazıldı. Repository'nin bir lideri oluyor.

**Local Git** ⇒ Distributed so that connectivity doesn't block work, easy...

**Distributed Git** ⇒ Team-centric so that collaboration happens naturally

Bir kopya günlük çalışan developerların bilgisayarında olur. Git çökerse...

Collaboration ⇒ Task'ler dağıtılıp birden fazla kişi çalışabilir veya bir taski birden çok kişi çalışabilir işlem yapabilir.

What is a repository ⇒ 2türlü oluşturur. GitHub üzerinden veya local de bir klasörle git init komutuyla oluşturulabilir. .git isimli gizli bir klasör oluşuyor ileri geri gitme bilgilerini tutan klasör.

git —version ⇒ versiyonu kontrol eder.

git config —global user.name "marcus"⇒ işlemleri global de yapmamızı sağlar. isim belirler

git config —global user.email "marcus@vv.com" ⇒ mail

git config —global core.editor "nano" ⇒ default kullanılacak olan tercihen editoru seçer.

git config —list ⇒ bilgileri listeler. Bilgileri sadece bir defa giriş yapmak yeterlidir.

git init ⇒ yeni local rapor oluşturur. Hangi dosyada oluşturacaksak oraya girmeliyiz. .git isimli gizli bir dosya oluşturur.

git help ⇒ yardım sayfasını açar... Ezbere gerek yok.

git status ⇒ to see the status of your repo. mevcut durumu için bilgiler verir.

git komut —help ⇒ o komutun help sayfasını açar.

git clone address ⇒ adresteki projeyi clone'lar.

3 Alan önemli ⇒

- Working Directory, çalışılan klasör

- Staging Area(Index) göndermeden önceki hali, çalıştığımız dosyaları öncelikle buraya atıyoruz. git add ile staging area ya atılır. sonra repository ediyoruz.
- Repository ⇒ paylaşım

#### ▼ Track a new file

touch file1.txt

vim file1.txt

git status

git add index.html ⇒ indexi ekle trackler.

veya git add . ⇒ bulunduğum klasörün tümünü ekle

git add -A ⇒ tüm değişiklikleri ekle

veya git add -u

git commit -m "index.html was created" ⇒ commit oluşturma. Açıklama ekliyorsun ne yaptığına dair. Dönüm noktası gibi.

git log ⇒ kim ne zaman hangi açıklamayla değişiklik yaptı..

## ► Commit

→ Commit the files on the stage

```
git commit -m "message"
```

→ Add and commit all tracked files

```
git commit -am "message"
```

→ amend commit message

```
git commit --amend
```

amend ⇒ nano da açtı oradada eklenebilir açıklama.

git commit -am "message" ⇒ trackli dosyayı direk gönderir.

git checkout ilkbeşkarakterveyatamamı ⇒ eski commite geçmeye yarar. HEAD ı taşıyor ileri veya geri gitmemizi sağlıyor. önceki ve sonraki versiyonları ararsında gezinebiliriz.