Commits och Repos

Allt du behöver veta om Commits och Repos!

- Starta ett remote repository
- Starta projekt / lokalt repo
- Lägg till ett remote repo till ditt projekt
- Pusha en commit till ett remote repo
- Hämta ändringar från ett remote repo
- Skapa en commit

Starta ett remote repository

Skapa ditt första remote repo på GitLab (https://git.ssis.nu/)

1. Klicka på “Projects” och sedan på “Your projects”.
2. Där klickar du sedan på skapa nytt projekt. (Alltså ett blankt projekt)
3. Sätt sedan in ett namn på ditt projekt och välj “Internal” på “Visibility Level”.

Nu har du skapat ditt första “remote repository” och den kan nu kommas åt av personer som är inloggade på skolans GitLab.

> Processen för att skapa ett remote repository är ungefär samma på alla Git hemsidor

Starta projekt / lokalt repo

Skapa projektet genom att starta ett nytt med Git.

1. Gå in i din projekt mapp och öppna ditt val av konsol i den mappen. (Git Bash / Powershell / Terminal)
2. Skriv `git init` för att starta ditt projekt.

När du nu startat ditt projekt så kan du börja arbeta.

> Du kan även göra detta på ett projekt som du redan har börjat med!

Lägg till ett remote repo till ditt projekt

För att enkelt kunna dela koden med dina kamrater eller kollegor så behöver du använda ett remote repo.

1. Gå in i din projekt mapp och öppna ditt val av konsol i den mappen. (Git bash / Powershell / Terminal)
2. Skriv “git remote add origin” och stoppa in SSH/HTTPS länken till ditt remote repo och klicka på enter.
3. Skriv "git remote" och så borde du se "origin" komma upp.

> "origin" är det standardnamnet som används på huvud remoten för projektet. men kan självklart bytas ut till vad du själv vill!

> Det är rekommenderat att alltid använda SSH för Git då det oftast är smidigare än användarnamn och lösenord!
>
> Kolla hur du lägger in en SSH nyckel i följande KDBI: SSH Nyckel och dess användning i Git

> HTTPS länken för projekt på https://git.ssis.nu/ kräver ett Admin lösenord och det rekomenderas att använda SSH länken.
>
> Detta kräver då att du har registrerat en SSH nyckel på hemsidan. Se följande KDBI: SSH Nyckel och dess användning i Git

Pusha en commit till ett remote repo

Pusha dina commits genom att pusha dem till ett remote repo.

För att kunna pusha till ett remote repo så måste du ha lagt till den i projektet. Se följande KDBI:

Lägg till ett remote repo till ditt projekt

1. Gå in i din projekt mapp och öppna ditt val av konsol i den mappen. (Git bash / Powershell / Terminal)
2. Skriv "git push RName RBName" där RName är namnet på din remote och RBName är namnet på din remote branch.

> Om det är första gången du ska ladda upp något till ditt remote repo så måste du ha med argumentet "-u"
> Alltså ser det då ut som följande "git push -u RName RBName".

> Om du redan definierat vilken remote och vilken remote branch du vill pusha till så räcker det med att skriva "git push"

Hämta ändringar från ett remote repo

Hämta commits från ett remote repo.

För att kunna hämta commits från ett remote repo så måste du ha lagt till den i projektet. Se följande KDBI: Lägg till ett remote repo till ditt projekt

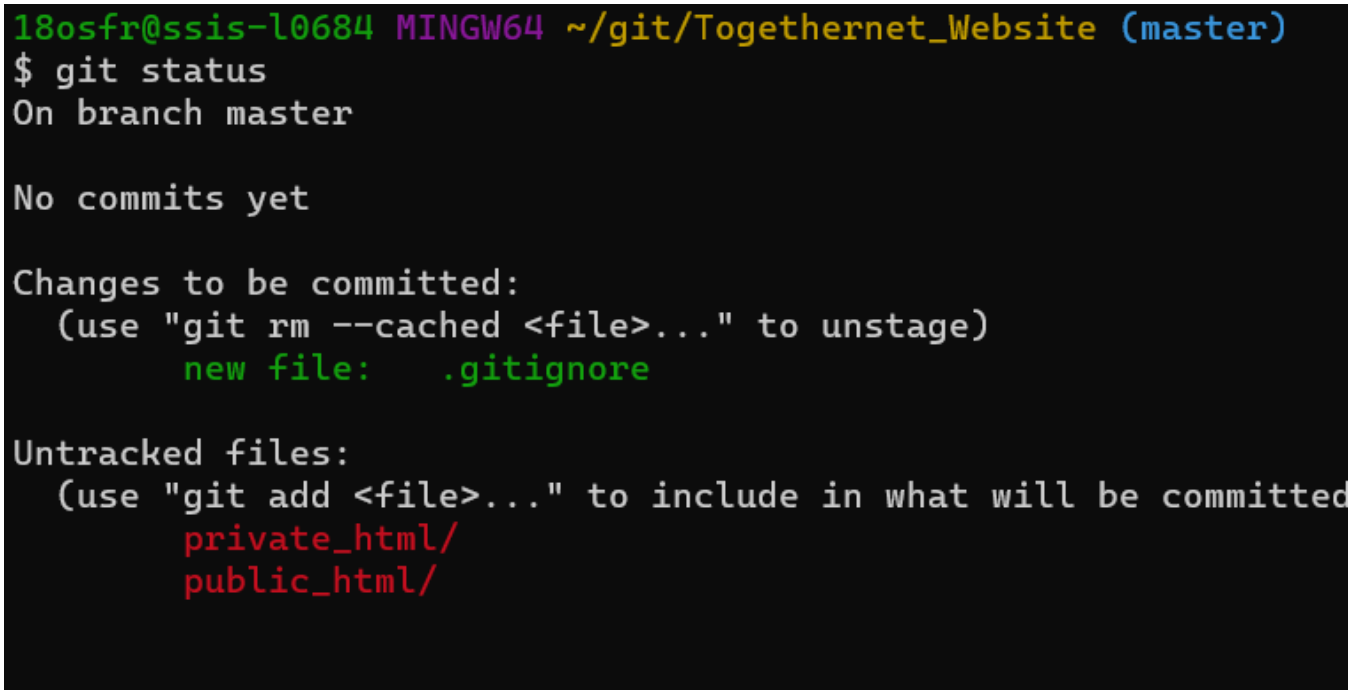
1. Gå in i din projekt mapp och öppna ditt val av konsol i den mappen. (Git bash / Powershell / Terminal)
2. Skriv "git pull" och då kommer den att tanka ner alla commits som är framför din nuvarande.

> Om du vill hämta men inte hoppa framåt så kan du göra "git fetch" och ladda ner commits utan att hoppa frammåt.

Skapa en commit

Skapa en commit med dina ändringar.

1. Gå in i din projekt mapp och öppna ditt val av konsol i den mappen. (Git bash / Powershell / Terminal)
2. Skriv `git status`. Då borde du få fram något som liknar bilden nedan:



3. Skriv `git add FILEName` där FILEName är den relativa vägen till filen.
   Detta kan du göra för varje fil eller använda `git add .` för att lägga till alla filer som är antingen untracked eller modified.
4. Skriv `git commit -m "TITLE"` där TITEL är titeln på din commit.
5. Skriv `git log` för att se dina commits. Följande bild är ett exempel på hur det kan se ut:

```
18osfr@ssis-l0684 MINGW64 ~/git/Sweeper_UnityEdition (master)
$ git log
commit 50da193e088fe801168978e38a68e0627f38eae7 (HEAD -> master, origin/master, origin/HEAD)
Author: Oscar Freij <18osfr@ad.ssis.nu>
Date:   Mon Mar 29 16:45:41 2021 +0200

    Added custom models and camera movement script.

    Also fixed problem where hover color overwrote the flag color on tiles

commit 3fab98d3801acef2c067380180f2e818b16a05bc
Author: Oscar Freij <otg020313@gmail.com>
Date:   Mon Mar 29 00:57:55 2021 +0200

    Added hover color, remade color change system and fixed error with bomb generation creating inf loo

commit 8e79ae07218bbbd6ff4774c7926723f7baad316a
Author: Oscar Freij <otg020313@gmail.com>
Date:   Sat Mar 27 19:11:20 2021 +0100

    First commit with code

commit 5f853b64f8b79f670db2e7ce9dda454d059b7c7e
Author: Oscar Freij <otg020313@gmail.com>
Date:   Sat Mar 27 19:09:05 2021 +0100

    Initial commit
```

Du kan stoppa in flera `-m` i din `git commit` för att lägga till flera rader.
Exempelvis: `git commit -m "TITLE" -m "MESSAGE"`

`git commit` tillåter ASCII så som `\n` `\r` och flera andra.