

# Git és GitHub

## Git Telepítése

Ahhoz, hogy otthon is használni tudjuk a Git-et, telepíteni kell. Ehhez látogassunk el a következő weboldalra: [Git letöltés](#). A telepítő során csak végig kell kattintani az alapételemezéseket, extra beállításra nincs szükség.

## Kezdő Lépések

### 1. Git Konfigurálása

Ha a Git nincs még beállítva, akkor az alábbi parancsokkal konfigurálhatjuk a felhasználónevet és az email címet:

```
git config --global user.email "you@example.com"
git config --global user.name "Your Name"
```

Itt cseréljük le a `"you@example.com"` és `"Your Name"` részeket a saját email címünkre és névre, amelyeket a GitHub-on is használunk.

## Verziókezelés Beállítása Egy Projektben

### 1. Projekt Inicializálása

Ha verziókezelni szeretnénk egy projektet, először inicializálnunk kell a Git-et a projekt mappájában:

```
git init
```

Ezt a parancsot csak egyszer futtassuk le egy projekten belül. A `git init` létrehoz egy `.git` mappát a projekt gyökérváltárban, amely tartalmazza a verziókezeléshez szükséges adatokat. Ezt a mappát nem kell manuálisan módosítanunk.

### 2. Projekt Állapotának Ellenőrzése

A projekt aktuális állapotának megtekintéséhez használjuk a következő parancsot:

```
git status
```

### 3. Fájlok Hozzáadása a Verziókezeléshez

Ha a `git status` parancs piros színnel jelöl fájlokat, amelyeket követni szeretnénk, akkor azokat hozzá kell adni a verziókezeléshez:

```
git add .
```

A `.` a parancs végén jelzi, hogy az összes módosított fájlt hozzáadjuk a verziókezeléshez.

### 4. Módosítások Rögzítése

Az `add` parancs után a fájlok változásait rögzíthetjük, azaz commitálhatjuk őket:

```
git commit -m "Állapot/változás rövid leírása"
```

A `-m` után a változások rövid leírását írjuk, ami segít azonosítani a commitot.

## Projekt Megosztása GitHub-on

### 1. Repository Létrehozása GitHub-on

Hozunk létre egy új repository-t a GitHub-on, majd másoljuk ki a repository git-es URL-jét. Ez általában így néz ki:

```
https://github.com/user/repo.git
```

Az URL http-vel kezdődik és git szóval zárul.

### 2. Távoli Repository Hozzáadása

Adjunk hozzá a lokális projektünkhöz távoli GitHub repository-t:

```
git remote add origin https://github.com/user/repo.git
```

Az `origin` egy alias név, amely az új repository-t azonosítja. Akármi más is lehetne.

### 3. Távoli Repository-k Ellenőrzése

Ellenőrizzük, hogy a távoli repository megfelelően hozzá lett adva, listázásra kerül-e:

```
git remote -v
```

### 4. Projekt Feltöltése GitHub-ra

A projekt feltöltéséhez a következő parancsot használjuk:

```
git push origin master
```

Az első alkalommal, amikor ezt futtatjuk, autentikációt fog kérni. Ha sikeres, a projekt feltöltődik a távoli GitHub repository-ra

## GitHub Projekt Letöltése

### 1. Projekt Klónozása

Ha egy GitHub projektről szeretnénk másolatot készíteni/letölteni lokális eszközre, pl egy otthoni eszközünkre, használjuk a következő parancsot:

```
git clone https://github.com/user/repo.git
```

Ez letölti a projektet és létrehoz egy mappát a repository nevével.

### 2. Mappába Lépés

A projekt mappájába léphetünk az alábbi paranccsal:

```
cd repo
```

### 3. Projekt Indítása

A projekt elindításához használjuk a következő parancsot:

```
code .
```

Ez megnyitja a projektet a Visual Studio Code szerkesztőben.