**Felküldés a távoli adattárba**

**Ebben a cikkben**

Használja git push a helyi ágon végrehajtott commit-ok távoli tárolóba küldésére.

**git push**

A git push parancsnak két argumentuma van:

* Egy távoli név, pl.origin
* Egy fióknév, pl.main

Például:

git push REMOTE-NAME BRANCH-NAME

Például általában fut git push origin main, hogy a helyi módosításokat az online tárhelybe helyezze.

**Ágak átnevezése**

Egy ág átnevezéséhez ugyanazt a git pushparancsot kell használnia, de még egy argumentumot kell hozzáadnia: az új ág nevét. Például:

git push REMOTE-NAME LOCAL-BRANCH-NAME:REMOTE-BRANCH-NAME

Ez áthelyezi a -t LOCAL-BRANCH-NAMEa te -re REMOTE-NAME, de átnevezve erre: REMOTE-BRANCH-NAME.

**A "nem előre gyorsított" hibák kezelése**

Ha egy tárhely helyi példánya nincs szinkronban a felfelé irányuló tárhellyel, vagy „mögötte” van, akkor a következő üzenetet kapja: non-fast-forward updates were rejected. Ez azt jelenti, hogy le kell töltenie vagy „le kell hoznia” a felfelé irányuló módosításokat, mielőtt végrehajthatja a helyi módosításokat.

A hibával kapcsolatos további információkért olvassa el a " [Nem gyors előretekeréssel kapcsolatos hibák kezelése](https://docs.github.com/en/github/getting-started-with-github/dealing-with-non-fast-forward-errors) " című részt.

**Címkék tologatása**

Alapértelmezés szerint és további paraméterek nélkül git pushelküldi az összes egyező ágat, amelynek neve megegyezik a távoli ágakkal.

Egyetlen címke leküldéséhez ugyanazt a parancsot adhatja ki, mint egy ág lenyomásakor:

git push REMOTE-NAME TAG-NAME

Az összes címke leküldéséhez írja be a következő parancsot:

git push REMOTE-NAME --tags

**Távoli ág vagy címke törlése**

Az ág törlésének szintaxisa első pillantásra kissé rejtélyes:

git push REMOTE-NAME:BRANCH-NAME

Vegye figyelembe, hogy a kettőspont előtt szóköz van. A parancs ugyanazokhoz a lépésekhez hasonlít, amelyeket egy ág átnevezéséhez tenne. Itt azonban azt mondod Gitnek, hogy *semmit* se toljon bele BRANCH-NAME. REMOTE-NAMEEmiatt git pushtörli az ágat a távoli lerakatból.

**Távirányítók és villák**

Talán már tudja, hogy a GitHubon [„elágazhat” adattárakat](https://guides.github.com/overviews/forking/) .

Amikor klónoz egy saját tárhelyet, egy távoli URL-t ad meg neki, amely megmondja a Gitnek, hogy hol kell letöltenie és leküldenie a frissítéseket. Ha együttműködni szeretne az eredeti tárhellyel, adjon hozzá egy új távoli URL-t, amelyet általában nek hívnak upstream, a helyi Git klónhoz:

git remote add upstream THEIR\_REMOTE\_URL

Most letöltheti a frissítéseket és *az elágazásokat a* villájukból:

git fetch upstream

# Grab the upstream remote's branches

> remote: Counting objects: 75, done.

> remote: Compressing objects: 100% (53/53), done.

> remote: Total 62 (delta 27), reused 44 (delta 9)

> Unpacking objects: 100% (62/62), done.

> From https://github.com/OCTOCAT/REPO

> \* [new branch] main -> upstream/main

Ha végzett a helyi módosításokkal, átküldheti helyi fiókját a GitHubba, és lehívási [kérelmet kezdeményezhet](https://docs.github.com/en/pull-requests/collaborating-with-pull-requests/proposing-changes-to-your-work-with-pull-requests/about-pull-requests) .

A villákkal való munkával kapcsolatos további információkért lásd [a " Villa szinkronizálása](https://docs.github.com/en/pull-requests/collaborating-with-pull-requests/working-with-forks/syncing-a-fork) " című részt.