nem értem mi a hiba. az előbb megcsináltam az első oldalamat, és az működik. Deployments 1 github-pages 2 hours ago

ChatGPT

Ha az első oldalad működik, és most egy másikkal kapcsolatban van problémád, lehet, hogy néhány egyszerű lépés segíthet a hiba azonosításában és megoldásában. Itt van egy lépésről lépésre útmutató, amit követhetsz a probléma megoldásához:

**1. Ellenőrizd a GitHub Pages Beállításokat**

1. **Navigálj a Tároló Beállításaihoz**:
   * Nyisd meg a GitHub tárolód főoldalát.
   * Kattints a „Settings” fülre a tároló oldalának jobb felső sarkában.
2. **GitHub Pages Beállítások**:
   * Görgess le a „GitHub Pages” szekcióhoz.
   * Ellenőrizd, hogy a „Source” beállításnál a megfelelő ágat választottad-e (pl. main vagy master), és hogy a gyökérkönyvtár (/ (root)) van-e kiválasztva.

**2. Ellenőrizd a Fájlok Elérhetőségét**

1. **Fájlok Elhelyezkedése**:
   * Győződj meg róla, hogy az index.html fájl a tároló gyökérkönyvtárában található. GitHub Pages alapértelmezett belépési oldalként keres egy index.html fájlt.
2. **Fájlok Hivatkozása**:
   * Ellenőrizd, hogy az index.html fájl helyesen hivatkozik-e a CSS, JS, és képfájlokra. Használj relatív hivatkozásokat, például:

html

Copy code

<link rel="stylesheet" href="styles.css">

<img src="project1.jpg" alt="Project 1">

**3. Ellenőrizd a Deployments Számát és Állapotát**

1. **Deployments**:
   * Nézd meg a GitHub Pages oldal „Deployments” szekcióját. Ha az előző deploy (kibővítés) sikeres volt, az utolsó deploy időpontja látható.
   * Kattints a legutolsó deploy hivatkozására a részletek megtekintéséhez.
2. **Deployments Állapota**:
   * Ellenőrizd, hogy volt-e hiba a deploy során. Ha a deploy nem sikerült, keresd meg a hibaüzeneteket.

**4. Böngésző és Cache**

1. **Böngésző Cache Törlése**:
   * Próbáld meg törölni a böngésző cache-ét (Ctrl+Shift+R vagy Cmd+Shift+R Mac-en), hogy biztos legyél benne, hogy a legfrissebb változtatások töltődnek be.
2. **Inkognitó Mód**:
   * Használj inkognitó módot a böngészőben, hogy ellenőrizd, nem a cache okozza-e a problémát.

**5. Hálózati Kérések és Hibák**

1. **Böngésző Fejlesztői Eszközei**:
   * Nyisd meg a böngésző fejlesztői eszközeit (F12 vagy Ctrl+Shift+I), és nézd meg a „Network” fület.
   * Ellenőrizd, hogy minden szükséges fájl betöltődik-e, és nincs-e 404-es hiba a fájlok elérhetőségével kapcsolatban.
2. **Konzol Hibák**:
   * Nézd meg a „Console” fület, hátha hibák vagy figyelmeztetések jelennek meg, amelyek segíthetnek a probléma azonosításában.

**6. Ellenőrizd az URL-t**

1. **GitHub Pages URL**:
   * Győződj meg róla, hogy a helyes URL-t használod, amely általában így néz ki: https://<felhasználóneved>.github.io/<tárolóneved>/.

**7. Egyéb Ellenőrzések**

1. **GitHub Status**:
   * Nézd meg a GitHub [status](https://www.githubstatus.com/) oldalát, hogy nincsenek-e ismert problémák a GitHub Pages szolgáltatással kapcsolatban.
2. **Dokumentáció**:
   * Nézd át a [GitHub Pages dokumentációt](https://docs.github.com/en/pages/getting-started-with-github-pages) további útmutatásért és hibakeresési tippekért.

**Összefoglalás**

Ha minden beállítás helyesnek tűnik, és a fájlok a megfelelő helyen vannak, de az oldal még mindig nem töltődik be, érdemes újra átnézni a lépéseket és biztosítani, hogy nincsenek elgépelések vagy hiányzó fájlok. Ha a probléma továbbra is fennáll, írd le pontosan, hogy mit látsz, és milyen hibák merülnek fel, hogy pontosabban tudjak segíteni.

4o mini

itt kell-e valamit csinálnom? agnes@Andras-Laptop MINGW64 ~ $

ChatGPT

A parancssori környezetben a Git használatakor az alábbiak szerint ellenőrizheted és kezelheted a GitHub tárolódat:

**1. Navigálj a Tároló Mappájába**

Ha helyi másolata van a tárolóról, először is navigálj a tároló mappájába a következő parancs használatával:

bash

Copy code

cd /path/to/your/repository

Cseréld le a /path/to/your/repository részt a saját tárolód elérési útjára.

**2. Ellenőrizd a Tároló Ágainak Listáját**

A tároló ágait a következő paranccsal listázhatod:

bash

Copy code

git branch -a

Ez megmutatja az összes helyi és távoli ágat. Az ágak listájában keresd meg a main vagy master ágat, amelyet a GitHub Pages használ.

**3. Ág Létrehozása és Állapotának Ellenőrzése**

Ha a main vagy master ág nem létezik, létrehozhatod és beállíthatod az alapértelmezett ágként:

* **Hozz létre egy új ágat (ha szükséges):**

bash

Copy code

git checkout -b main

* **Pushold fel az ágat a GitHub-ra:**

bash

Copy code

git push -u origin main

* **Ellenőrizd, hogy az ág most már megjelenik-e a távoli ághoz:**

bash

Copy code

git branch -r

**4. GitHub Pages Beállítása**

1. **Nyisd meg a Tároló Beállításait**:
   * Kattints a „Settings” fülre a GitHub weboldalán.
2. **GitHub Pages Szekció**:
   * Görgess le a „GitHub Pages” szekcióhoz.
   * Válaszd ki a megfelelő ágat a „Source” legördülő menüből (pl. main).
   * Válaszd ki a „/ (root)” könyvtárat, ha a fájlok a tároló gyökérkönyvtárában találhatóak.

**5. Helyi Változtatások Commitálása és Pusholása**

Ha módosítottál valamit a fájlokon, ne felejtsd el commitálni és pusholni a változtatásokat:

* **Add hozzá a fájlokat a commit-hoz:**

bash

Copy code

git add .

* **Készíts egy commit-ot:**

bash

Copy code

git commit -m "Your commit message"

* **Pushold fel a változtatásokat a GitHub-ra:**

bash

Copy code

git push origin main

**6. Hibák Ellenőrzése**

* **Hálózati Kérések Ellenőrzése**:
  + Használj git status parancsot a helyi változtatások állapotának ellenőrzéséhez.
* **Git Log**:
  + Nézd meg a commit előzményeit:

bash

Copy code

git log

Ez a parancs megmutatja az összes korábbi commitot, amely segíthet a problémák nyomozásában.

**7. GitHub Status Oldal**

* **Ellenőrizd a GitHub Status Oldalt**:
  + Nézd meg a [GitHub Status](https://www.githubstatus.com/) oldalt, hogy nincsenek-e ismert problémák a GitHub Pages szolgáltatással kapcsolatban.

**Összefoglalás**

Ha a parancssori környezetben az agnes@Andras-Laptop MINGW64 ~ $ promptot látod, az azt jelenti, hogy a Git Bash vagy más terminál alkalmazás fut. Az itt végrehajtott parancsokkal ellenőrizheted a tároló állapotát, az ágakat és a fájlokat, valamint kezelheted a GitHub Pages beállításait.

Ha bármilyen további kérdésed van, vagy specifikus problémát tapasztalsz, kérlek, írd le pontosan, és szívesen segítek!

4o mini

mi a baj? agnes@Andras-Laptop MINGW64 ~ $ cd/D:\\_mentes\_d\adatok\andras\web\weboldal3 bash: cd/D:\_mentes\_dadatokandraswebweboldal3: No such file or directory

ChatGPT

A hibaüzenet alapján úgy tűnik, hogy a parancsban elgépelés történt, ami miatt a rendszer nem találja az adott könyvtárat. A cd parancs használatakor fontos a helyes szintaxis és a megfelelő elválasztók használata.

**Helyes Parancs Szintaxis**

A cd (change directory) parancs használatakor az elérési utakat az alábbiak szerint kell megadni:

1. **Windows Elérési Út Elválasztók**: A Windows rendszeren a könyvtárak elválasztója a backslash (\). Azonban a Git Bash és a legtöbb Unix-alapú környezetben a slash (/) használatos.
2. **Leírás és Példák**:
   * **Git Bash vagy Unix-Alapú Környezetben**: Használj / elválasztókat, és győződj meg róla, hogy az útvonal teljesen pontos.

bash

Copy code

cd /d/\_mentes\_d/adatok/andras/web/weboldal3

* + **Windows Command Prompt**: Ha Windows Command Promptot használsz, akkor a backslash (\) a megfelelő elválasztó.

cmd

Copy code

cd D:\\_mentes\_d\adatok\andras\web\weboldal3

**Lépések a Hiba Kijavításához**

1. **Győződj Meg a Könyvtár Létezéséről**:
   * Ellenőrizd, hogy a megadott könyvtár létezik-e. Nyisd meg a fájlkezelőt és ellenőrizd, hogy az D:\\_mentes\_d\adatok\andras\web\weboldal3 könyvtár valóban ott van-e.
2. **Használj Helyes Elérési Út Elválasztókat**:
   * Git Bash használatakor a könyvtárak elválasztója /, nem \.
3. **Parancs Példák**:
   * **Git Bash**:

bash

Copy code

cd /d/\_mentes\_d/adatok/andras/web/weboldal3

Ha az elérési út kezdőbetűje D és nem egy teljes útvonal, próbáld a következő parancsot:

bash

Copy code

cd /d/\_mentes\_d/adatok/andras/web/weboldal3

Vagy:

bash

Copy code

cd /mnt/d/\_mentes\_d/adatok/andras/web/weboldal3

* + **Windows Command Prompt**:

cmd

Copy code

cd D:\\_mentes\_d\adatok\andras\web\weboldal3

1. **Fájl és Könyvtár Nézet**:
   * Győződj meg róla, hogy a könyvtár nevében nincsenek elgépelések vagy extra szóközök. Például, ha az elérési út D:\\_mentes\_d\adatok\andras\web\weboldal3, ellenőrizd, hogy minden elérési szint helyesen van megadva.

**Ellenőrzés Git Bash-ban**

* **Listázd a Könyvtárakat**: Használhatod a ls parancsot, hogy listázd az aktuális könyvtár tartalmát, és ellenőrizd, hogy a megcélzott könyvtár létezik-e:

bash

Copy code

ls /d/\_mentes\_d/adatok/andras/web

* **Navigálj a Gyökérkönyvtárhoz**: Próbáld meg a gyökérkönyvtárba navigálni, majd ellenőrizd a további könyvtárakat:

bash

Copy code

cd /

ls

cd d

ls

cd \_mentes\_d

ls

cd adatok

ls

cd andras

ls

cd web

ls

cd weboldal3

Ha továbbra is problémád van, kérlek, írd le a pontos hibaüzenetet és az elérési utat, amelyet használsz. Ezen információk alapján pontosabb segítséget tudok nyújtani.

4o mini

agnes@Andras-Laptop MINGW64 ~ $ cd /d/\_mentes\_d/adatok/andras/web/weboldal3 agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 $ git branch -a fatal: not a git repository (or any of the parent directories): .git agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3

ChatGPT

A „fatal: not a git repository (or any of the parent directories): .git” hibaüzenet azt jelenti, hogy a megadott könyvtár nem tartalmaz Git tárolót. Ennek oka lehet, hogy a könyvtár nem inicializált Git tároló, vagy esetleg nem a megfelelő helyen vagy.

**Lépések a Hiba Kijavításához**

**1. Ellenőrizd a Könyvtárat**

* **Győződj meg arról, hogy a könyvtár valóban létezik és helyesen navigáltál oda**.

Ha a könyvtár nem tartalmaz .git mappát, akkor az nem Git tároló.

**2. Inicializáld a Git Tárolót**

Ha a könyvtár valóban a projekted könyvtára, és szeretnéd Git tárolóvá alakítani, használhatod az alábbi lépéseket:

1. **Navigálj a Könyvtárba**:
   * Lépj be a projekt könyvtárába, ahol a projekt fájljaid találhatóak.
2. **Inicializáld a Git Tárolót**:

bash

Copy code

git init

Ez létrehozza a .git mappát, és az aktuális könyvtár Git tárolóvá válik.

1. **Add hozzá a Fájlokat a Tárolóhoz**:
   * Add hozzá az összes fájlt a Git követéshez:

bash

Copy code

git add .

1. **Készíts egy Commit-ot**:
   * Commitáld az első változtatásokat:

bash

Copy code

git commit -m "Initial commit"

1. **Kapcsold Össze a Tárolót a GitHub-bal**:
   * Ha már van egy tároló a GitHub-on, add hozzá a távoli URL-t:

bash

Copy code

git remote add origin <repository-url>

* + Pushold fel a változtatásokat:

bash

Copy code

git push -u origin main

1. **Megjegyzés**: Ha az alapértelmezett ág nem main, hanem master, cseréld le main-et master-re a parancsokban.

**3. Ellenőrizd a Létező Git Tárolót**

Ha biztos vagy benne, hogy a könyvtárnak Git tárolónak kell lennie, de nem található .git mappa:

1. **Nézd Meg, Hol van a .git Mappa**:
   * Ellenőrizd, hogy a .git mappa valahol a projekt könyvtár fölött létezik-e. Használhatod az alábbi parancsokat a szülő könyvtárak ellenőrzéséhez:

bash

Copy code

cd ..

ls -a

1. **Keresd Meg a .git Mappát**:
   * Nézd meg a szülő könyvtárakban, hogy hol található a .git mappa:

bash

Copy code

find . -name ".git"

1. Ez megmondja, hogy hol található a .git mappa az aktuális könyvtár hierarchiájában.

**4. Töltsd Le a Git Tárolót a GitHub-ról**

Ha a projekt GitHub-on található, és szeretnéd letölteni a tárolót:

1. **Klónozd a Tárolót a GitHub-ról**:
   * Használj git clone parancsot a tároló letöltéséhez:

bash

Copy code

git clone <repository-url>

1. Cseréld le <repository-url>-t a GitHub tároló URL-jére.

**Összefoglalás**

* **Ha a könyvtár nem tartalmaz .git mappát**, inicializáld Git tárolóvá az git init parancs segítségével.
* **Ha a projektet már klónoztad volna a GitHub-ról**, győződj meg róla, hogy a könyvtár a helyes helyen van, és tartalmazza a .git mappát.

Ha további kérdéseid vannak, vagy ha más problémát észleltél, kérlek, írj részletesebben, hogy pontosabban tudjak segíteni.

4o mini

add hozzá a távoli URL-t, ez mit jelent, honnan tudom mi az?

ChatGPT

A távoli URL (remote URL) a Git tárolóhoz kapcsolódó webes cím, amely általában egy GitHub, GitLab vagy Bitbucket tároló URL-je. Ez az URL lehetővé teszi, hogy a helyi tárolódat szinkronizáld a távoli tárolóval, például feltölthesd (push) a változtatásokat, vagy letölthesd (pull) a frissítéseket.

**Hogyan Tudd Meg a Távoli URL-t**

**1. GitHub Tároló URL Megkeresése**

1. **Lépj be a GitHub Fiókodba**:
   * Nyisd meg a [GitHub](https://github.com) weboldalt, és jelentkezz be a fiókodba.
2. **Navigálj a Tárolódhoz**:
   * Keresd meg azt a tárolót, ahová szeretnéd feltölteni a helyi változtatásokat.
3. **Másold az URL-t**:
   * A tároló oldalán kattints a „Code” gombra, amely a jobb oldalon található. Itt látni fogod a távoli URL-t, amely lehet HTTPS vagy SSH formátumban.
     + **HTTPS URL**: https://github.com/username/repository.git
     + **SSH URL**: git@github.com:username/repository.git

Válaszd ki az URL-t (általában a „HTTPS” a leggyakrabban használt), és másold ki.

**2. Git Tárolóhoz Távoli URL Hozzáadása**

Ha létrehoztál egy új Git tárolót és szeretnéd csatlakoztatni azt a GitHub tárolóhoz, kövesd az alábbi lépéseket:

1. **Navigálj a Lokális Tároló Könyvtárába**:
   * Ha már a helyi tárolód könyvtárában vagy, add hozzá a távoli URL-t a következő paranccsal:

bash

Copy code

git remote add origin <repository-url>

Cseréld le <repository-url>-t a GitHub-on másolt URL-re. Például:

bash

Copy code

git remote add origin https://github.com/username/repository.git

1. **Pusholj az Alapértelmezett Ágba**:
   * Most már pusholhatod a változtatásokat a távoli tárolóra:

bash

Copy code

git push -u origin main

**Megjegyzés**: Ha az alapértelmezett ág nem main, hanem master, akkor használj master-t a main helyett.

1. **Ellenőrizd a Távoli URL-t**:
   * Ellenőrizheted, hogy helyesen adtad-e hozzá a távoli URL-t a következő paranccsal:

bash

Copy code

git remote -v

Ez megmutatja az összes hozzáadott távoli URL-t, amelyeket a tároló használ.

**Példák a Távoli URL Használatára**

* **HTTPS URL hozzáadása**:

bash

Copy code

git remote add origin https://github.com/username/repository.git

* **SSH URL hozzáadása** (ha SSH kulcsot használsz):

bash

Copy code

git remote add origin git@github.com:username/repository.git

**További Információk**

* **HTTPS**: Egyszerűen használható, de minden alkalommal jelszót kell megadni, hacsak nem tárolod el a hitelesítő adatokat.
* **SSH**: Biztonságosabb, de SSH kulcsokat kell konfigurálni.

Ha további kérdéseid vannak a távoli URL-ekkel vagy GitHub használatával kapcsolatban, bátran kérdezz!

4o mini

ez? https://github.com/ahabram/adoszpoly.git

ChatGPT

Igen, **https://github.com/ahabram/adoszpoly.git** az a URL, amely a GitHub tárolódra mutat. Ezt az URL-t használhatod a helyi Git tárolódban a GitHub tárolóhoz való kapcsolódáshoz.

**Lépések a Távoli URL Hozzáadásához**

**1. Navigálj a Helyi Tároló Könyvtárába**

Először is, győződj meg róla, hogy a helyi könyvtárad a megfelelő helyen van. Nyisd meg a Git Bash-t vagy a parancssort és navigálj a projekt könyvtárába, ahol a helyi Git tárolód található.

bash

Copy code

cd /d/\_mentes\_d/adatok/andras/web/weboldal3

**2. Inicializálj Git Tárolót (ha még nem tetted meg)**

Ha még nem inicializáltad Git tárolóként, tedd meg a következő parancs segítségével:

bash

Copy code

git init

**3. Add Hozzá a Távoli URL-t**

Most add hozzá a GitHub tároló URL-t a helyi tárolóhoz. A következő parancsot használd:

bash

Copy code

git remote add origin https://github.com/ahabram/adoszpoly.git

Ez a parancs hozzáad egy „origin” nevű távoli URL-t, amely a GitHub tárolódra mutat.

**4. Pushold Fel a Változtatásokat**

Most, hogy a távoli URL hozzá van adva, pusholhatod a helyi változtatásokat a GitHub tárolóra. Ha az első commit-ot még nem készítetted el, először készíts egy commit-ot:

bash

Copy code

git add .

git commit -m "Initial commit"

Ezután pushold fel a változtatásokat:

bash

Copy code

git push -u origin main

**Megjegyzés**: Ha az alapértelmezett ág neve master, akkor használj master-t a main helyett:

bash

Copy code

git push -u origin master

**5. Ellenőrizd a Távoli URL-t**

Ellenőrizheted, hogy helyesen adtad-e hozzá a távoli URL-t a következő parancs segítségével:

bash

Copy code

git remote -v

Ez megmutatja az összes távoli URL-t, amelyeket a Git tároló használ.

**Összefoglalás**

A https://github.com/ahabram/adoszpoly.git URL a GitHub tárolód URL-je, és ezt használhatod a helyi Git tárolód kapcsolódásához. A fenti lépéseket követve beállíthatod a távoli kapcsolatot és pusholhatod a helyi változtatásokat a GitHub tárolóra.

Ha bármilyen probléma merül fel, vagy további kérdéseid vannak, kérlek, jelezd!

4o mini

agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (master) $ git add . agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (master) $ git commit -m "Initial commit" [master (root-commit) 951602b] Initial commit 9 files changed, 168 insertions(+) create mode 100644 20201031\_141942.jpg create mode 100644 Csatlos\_Green\_V04\_001.jpg create mode 100644 index.html create mode 100644 info.txt create mode 100644 project1.jpg create mode 100644 project2.jpg create mode 100644 project3.jpg create mode 100644 project4.jpg create mode 100644 styles.css agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (master) $ git remote add origin https://github.com/ahabram/adoszpoly.git agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (master) $ git add . agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (master) $ git commit -m "Initial commit" On branch master nothing to commit, working tree clean agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (master) $ ^C agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (master) $ git push -u origin main error: src refspec main does not match any error: failed to push some refs to 'https://github.com/ahabram/adoszpoly.git' agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (master) $ git remote -v origin https://github.com/ahabram/adoszpoly.git (fetch) origin https://github.com/ahabram/adoszpoly.git (push) agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (master) $

ChatGPT

A error: src refspec main does not match any hibaüzenet azt jelenti, hogy a main ág nem létezik a helyi tárolódban. Jelenleg a helyi tárolóban az ág neve master, nem main.

**Lépések a Hiba Kijavításához**

**1. Pushold a master Ágat**

Mivel a helyi ág neve master, pushold fel a változtatásokat a master ágon, ne a main ágon:

bash

Copy code

git push -u origin master

Ez feltölti a master ágat a GitHub tárolóra.

**2. Ellenőrizd a Távoli Ágakat**

A távoli tároló állapotát az alábbi paranccsal ellenőrizheted:

bash

Copy code

git branch -r

Ez megmutatja a távoli ágakat, és ellenőrizheted, hogy a master ág szerepel-e.

**3. Ha Szeretnéd Átnevezni az Ágat main-re**

Ha szeretnéd az ág nevét master-ről main-re váltani, kövesd az alábbi lépéseket:

1. **Helyi Ág Átnevezése main-re**:

bash

Copy code

git branch -m master main

1. **Pushold a main ágat a GitHub-ra**:

bash

Copy code

git push -u origin main

1. **Állítsd be az alapértelmezett ágat a GitHub tárolón**:
   * Nyisd meg a GitHub tárolót a böngésződben.
   * Kattints a „Settings” (Beállítások) fülre.
   * A „Branches” (Ágak) szekció alatt állítsd be a „Default branch” (Alapértelmezett ág) értéket main-re.
2. **Töröld a Távoli master Ágat (opcionális)**:

Ha már nincs szükség a master ágra, törölheted azt a GitHub tárolóról:

bash

Copy code

git push origin --delete master

**4. Ellenőrizd a Távoli URL-t és Ágakat**

Győződj meg arról, hogy a távoli URL és ágak helyesen vannak beállítva:

* **Távoli URL**:

bash

Copy code

git remote -v

* **Helyi Ágak**:

bash

Copy code

git branch

* **Távoli Ágak**:

bash

Copy code

git branch -r

**Összefoglalás**

* **Pushold a helyi master ágat a távoli master ágra**:

bash

Copy code

git push -u origin master

* **Átnevezheted az ágat main-re, és beállíthatod azt alapértelmezettnek, ha szükséges**.

Ha további kérdéseid vannak, vagy más problémát észleltél, kérlek, jelezd!

4o mini

agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (master) $ ^C agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (master) $ git push -u origin master fatal: unable to access 'https://github.com/ahabram/adoszpoly.git/': Failed to connect to github.com port 443 after 84 ms: Couldn't connect to server agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (master) $ git branch -r agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (master) $ git branch -m master main agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $ git push -u origin main fatal: unable to access 'https://github.com/ahabram/adoszpoly.git/': Failed to connect to github.com port 443 after 84 ms: Couldn't connect to server agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $

ChatGPT

A hibaüzenet, amit kapsz (Failed to connect to github.com port 443: Couldn't connect to server), azt jelzi, hogy a Git nem tud kapcsolódni a GitHub szerverhez. Ennek több lehetséges oka lehet, és az alábbi lépésekkel próbálhatod meg orvosolni a problémát:

**1. Ellenőrizd a Internet Kapcsolatot**

* **Győződj meg róla, hogy az internetkapcsolat működik**. Próbálj meg megnyitni más weboldalakat a böngészőben, és ellenőrizd, hogy hozzáférsz az internethez.

**2. Ellenőrizd a GitHub Weboldalt**

* **Ellenőrizd, hogy a GitHub elérhető-e**. Nézd meg a [GitHub Status oldalát](https://www.githubstatus.com/), hogy van-e bármilyen ismert probléma a GitHub szerverekkel.

**3. Frissítsd a Git Verziót**

* **Győződj meg róla, hogy a Git naprakész**. Ha régi verziót használsz, frissítheted a Git-et a legújabb verzióra.

**4. Ellenőrizd a Távoli URL-t**

* **Ellenőrizd, hogy a távoli URL helyes-e**. Használhatod a következő parancsot az URL ellenőrzésére:

bash

Copy code

git remote -v

Győződj meg arról, hogy az URL helyes és a GitHub tárolóra mutat.

* **Ha az URL helytelen, frissítheted**:

bash

Copy code

git remote set-url origin https://github.com/ahabram/adoszpoly.git

**5. Ellenőrizd a Proxy Beállításokat**

* **Ha proxy-t használsz**, győződj meg arról, hogy a proxy beállítások helyesek. A Git proxy beállításait az alábbi parancsokkal ellenőrizheted és módosíthatod:
  + Ellenőrizd a beállításokat:

bash

Copy code

git config --global --get http.proxy

* + Ha szükséges, állítsd be a proxy-t:

bash

Copy code

git config --global http.proxy http://proxyuser:proxypassword@proxy.server.com:port

* + Ha proxy-t nem használsz, győződj meg arról, hogy nincsenek proxy beállítások:

bash

Copy code

git config --global --unset http.proxy

**6. Ellenőrizd a GitHub Hitelesítést**

* **Győződj meg arról, hogy a GitHub hitelesítési adatokat helyesen állítottad-e be**. Ha HTTPS URL-t használsz, szükséged lehet GitHub felhasználói névre és jelszóra, vagy egy [personal access token](https://github.com/settings/tokens) használatára.

**7. Próbáld Újra a Kapcsolódást**

* **Próbálj meg újra kapcsolódni**. Néha a probléma csak átmeneti, és a próbálkozások ismétlése segíthet.

**8. Hálózati Beállítások**

* **Ellenőrizd a tűzfal vagy víruskereső beállításokat**, amelyek esetleg blokkolhatják a GitHub elérését.

**Példák a Gyakori Parancsokra**

* **Ellenőrizd az URL-t**:

bash

Copy code

git remote -v

* **Frissítsd az URL-t**:

bash

Copy code

git remote set-url origin https://github.com/ahabram/adoszpoly.git

* **Ellenőrizd a Proxy Beállításokat**:

bash

Copy code

git config --global --get http.proxy

* **Állítsd be a Proxy-t** (ha szükséges):

bash

Copy code

git config --global http.proxy http://proxyuser:proxypassword@proxy.server.com:port

* **Töröld a Proxy Beállításokat** (ha nem szükséges):

bash

Copy code

git config --global --unset http.proxy

Ha ezek a lépések nem oldják meg a problémát, kérlek, írd le a további részleteket a problémáról, és igyekszem további segítséget nyújtani!

4o mini

agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $ git remote -v origin https://github.com/ahabram/adoszpoly.git (fetch) origin https://github.com/ahabram/adoszpoly.git (push)

ChatGPT

A git remote -v parancs kimenete azt mutatja, hogy a távoli URL helyesen van beállítva: https://github.com/ahabram/adoszpoly.git. Azonban a fatal: unable to access 'https://github.com/ahabram/adoszpoly.git/': Failed to connect to github.com port 443 after 84 ms: Couldn't connect to server hibaüzenet továbbra is azt jelzi, hogy a Git nem tud kapcsolódni a GitHub szerverhez.

**Ellenőrzési és Hibaelhárítási Lépések**

**1. Internetkapcsolat Ellenőrzése**

* **Ellenőrizd, hogy más weboldalak elérhetők-e**. Nyisd meg a böngészőt, és látogass el néhány más weboldalra, hogy megbizonyosodj róla, hogy van internetkapcsolatod.

**2. GitHub Elérhetőségének Ellenőrzése**

* **Nyisd meg a GitHub weboldalt**: Próbáld meg elérni a GitHub-ot a böngészőben [GitHub weboldal](https://github.com). Ha a weboldal elérhető, akkor a GitHub szerverek működnek.
* **Ellenőrizd a GitHub állapotát**: Látogass el a [GitHub Status oldalára](https://www.githubstatus.com/), hogy megtudd, van-e bármilyen ismert probléma a GitHub szolgáltatásaival.

**3. Proxy és Tűzfal Ellenőrzése**

* **Ellenőrizd a Proxy beállításokat**: Ha proxy-t használsz, győződj meg róla, hogy a proxy beállítások helyesen vannak konfigurálva.

bash

Copy code

git config --global --get http.proxy

Ha proxy-t nem használsz, győződj meg arról, hogy nincsenek proxy beállítások:

bash

Copy code

git config --global --unset http.proxy

* **Ellenőrizd a tűzfal beállításokat**: A tűzfal vagy biztonsági szoftver esetleg blokkolhatja a GitHub elérését. Ellenőrizd a tűzfal és biztonsági beállításokat, hogy biztosítsd, hogy a GitHub nem blokkolva van.

**4. Git és SSL/TLS Beállítások**

* **Frissítsd a Git verziót**: Győződj meg arról, hogy a Git legújabb verzióját használod. Ha nem, frissítsd a Git-et a legújabb verzióra.
* **SSL/TLS beállítások ellenőrzése**: Ha problémád van az SSL/TLS kapcsolattal, akkor az alábbi parancsok segíthetnek.

**Ellenőrizd az SSL/TLS beállításokat**:

bash

Copy code

git config --global http.sslVerify

Ha a SSL/TLS ellenőrzés problémás, ideiglenesen kikapcsolhatod a következő parancs használatával (ne használj ilyen beállítást hosszú távon, mivel ez biztonsági kockázatot jelenthet):

bash

Copy code

git config --global http.sslVerify false

**5. GitHub SSH URL Használata**

* Ha a HTTPS URL nem működik, próbálkozhatsz az SSH URL használatával. Ehhez SSH kulcsot kell beállítanod.

**SSH URL beállítása**:

* 1. Generálj egy SSH kulcsot, ha még nem rendelkezel vele:

bash

Copy code

ssh-keygen -t ed25519 -C "your\_email@example.com"

* 1. Add hozzá az SSH kulcsot a GitHub-hoz. Másold ki az SSH kulcsot:

bash

Copy code

cat ~/.ssh/id\_ed25519.pub

Lépj be a GitHub fiókodba, és navigálj a „Settings” > „SSH and GPG keys” részhez. Add hozzá az új SSH kulcsot.

* 1. Állítsd be az SSH URL-t a távoli URL helyett:

bash

Copy code

git remote set-url origin git@github.com:ahabram/adoszpoly.git

* 1. Próbáld meg újra a git push parancsot:

bash

Copy code

git push -u origin main

**Összefoglalás**

* **Ellenőrizd az internetkapcsolatot** és **a GitHub elérhetőségét**.
* **Nézd meg a proxy és tűzfal beállításokat**.
* **Frissítsd a Git verzióját**, és ellenőrizd az SSL/TLS beállításokat.
* **Próbálj ki SSH URL-t**, ha a HTTPS nem működik.

Ha ezek a lépések nem oldják meg a problémát, vagy további információra van szükséged, kérlek, írd le a további részleteket!

4o mini

Settings” > „SSH and GPG keys hol van?

ChatGPT

Az „SSH and GPG keys” beállítást a GitHub-on találhatod, ahol hozzáadhatod az SSH kulcsokat, hogy biztonságosan csatlakozhass a tárolóidhoz. Itt van, hogyan navigálhatsz hozzá a GitHub felületén:

**SSH Kulcs Hozzáadása GitHub-hoz**

1. **Jelentkezz be GitHub Fiókodba**
   * Nyisd meg a GitHub weboldalt: [GitHub](https://github.com)
   * Jelentkezz be a GitHub fiókoddal.
2. **Navigálj a Beállításokhoz**
   * Kattints a profilképedre a jobb felső sarokban.
   * A legördülő menüből válaszd ki a „**Settings**” (Beállítások) opciót.
3. **Menj az SSH és GPG Kulcsokhoz**
   * A bal oldali menüben keresd meg a „**SSH and GPG keys**” szekciót. Általában a „**Access**” rész alatt található, vagy keresheted is a „**SSH**” kifejezést a bal oldali menü keresőjében.
   * Kattints a „**SSH and GPG keys**” linkre.
4. **Add hozzá az SSH Kulcsot**
   * Kattints a „**New SSH key**” vagy „**Add SSH key**” gombra.
   * Add meg az SSH kulcs nevét (például „My Laptop”).
   * Másold be az SSH kulcsot a id\_ed25519.pub fájlból a „**Key**” mezőbe. Ezt a következő paranccsal tekintheted meg a terminálban:

bash

Copy code

cat ~/.ssh/id\_ed25519.pub

* + Kattints a „**Add SSH key**” gombra a kulcs hozzáadásához.

**SSH Kulcs Generálása**

Ha még nem rendelkezel SSH kulccsal, kövesd az alábbi lépéseket az új kulcs generálásához:

1. **Generálj új SSH kulcsot**:

bash

Copy code

ssh-keygen -t ed25519 -C "your\_email@example.com"

* + Kövesd az utasításokat, és hagyd az alapértelmezett helyen (~/.ssh/id\_ed25519).

1. **Másold ki az SSH kulcsot**:

bash

Copy code

cat ~/.ssh/id\_ed25519.pub

Másold ki az itt megjelenő szöveget, és illeszd be a GitHub „**Key**” mezőjébe.

**SSH Kulcsok Ellenőrzése**

Győződj meg arról, hogy az SSH kulcs működik, a következő paranccsal:

bash

Copy code

ssh -T git@github.com

Ha a beállítás helyes, a következő választ kell kapnod:

plaintext

Copy code

Hi username! You've successfully authenticated, but GitHub does not provide shell access.

**SSH URL Beállítása**

Miután hozzáadtad az SSH kulcsot, állítsd be a GitHub távoli URL-t SSH-ra a következő paranccsal:

bash

Copy code

git remote set-url origin git@github.com:ahabram/adoszpoly.git

Ezután próbáld meg újra a push parancsot:

bash

Copy code

git push -u origin main

**Összefoglalás**

* **Navigálj a GitHub „SSH and GPG keys” szekcióhoz** a fiók beállításaiban.
* **Generálj és add hozzá az SSH kulcsot**.
* **Állítsd be a távoli URL-t SSH-ra**, és próbáld meg újra a push-ot.

Ha további kérdéseid vannak vagy problémáid merülnek fel, kérlek, jelezd!

4o mini

agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $ ssh -T git@github.com ssh: connect to host github.com port 22: Permission denied

ChatGPT

A Permission denied hibaüzenet azt jelzi, hogy a SSH kapcsolaton keresztül nem tudsz kapcsolódni a GitHub szerverhez. Ez a probléma több dologból fakadhat:

**1. Ellenőrizd a SSH Kulcsot**

* Győződj meg róla, hogy a SSH kulcsot helyesen generáltad és hozzáadtad a GitHub fiókodhoz.
* Ellenőrizd, hogy a megfelelő SSH kulcsot használod. A kulcsok alapértelmezés szerint az ~/.ssh könyvtárban találhatóak (id\_ed25519 és id\_ed25519.pub).

**2. Ellenőrizd a SSH Agent Beállításokat**

* **Add hozzá a SSH kulcsot az SSH agenthez**, ha még nem tetted meg:

bash

Copy code

eval "$(ssh-agent -s)"

ssh-add ~/.ssh/id\_ed25519

* **Ellenőrizd, hogy a kulcs hozzá van adva az agenthez**:

bash

Copy code

ssh-add -l

Ennek a parancsnak listáznia kell az aktuálisan hozzáadott kulcsokat.

**3. Ellenőrizd a Tűzfal és Proxy Beállításokat**

* **Tűzfal vagy hálózati proxy problémák**: A vállalati vagy iskolai hálózatok gyakran blokkolják a 22-es portot, amelyet az SSH használ. Ha lehetséges, próbálj meg egy másik hálózathoz csatlakozni (pl. otthoni hálózat) vagy beszélj a hálózati adminisztrátorral.

**4. Használj Alternatív Portot az SSH-hoz**

* **GitHub támogatja az SSH alternatív portját (443)**. Próbáld meg ezt a portot:

bash

Copy code

ssh -T -p 443 git@ssh.github.com

**5. Ellenőrizd a SSH Kulcs Formátumát és Helyét**

* **Győződj meg arról, hogy a kulcs helyesen van formázva és a megfelelő helyen van**. Az alapértelmezett hely ~/.ssh/id\_ed25519, de ha más helyet használsz, győződj meg arról, hogy azt is megfelelően konfiguráltad.

**6. Frissítsd a Git és SSH Klienst**

* **Frissítsd a Git-et és az SSH klienst** a legújabb verzióra, mivel a régi verziók problémákat okozhatnak.

**Példák a Parancsokra**

1. **SSH Agent Inicializálása és Kulcs Hozzáadása**:

bash

Copy code

eval "$(ssh-agent -s)"

ssh-add ~/.ssh/id\_ed25519

1. **SSH Kulcsok Ellenőrzése**:

bash

Copy code

ssh-add -l

1. **Alternatív Port Használata**:

bash

Copy code

ssh -T -p 443 git@ssh.github.com

1. **SSH Kulcs Generálása és Ellenőrzése**:

Ha még nem generáltál SSH kulcsot, ezt a parancsot használhatod:

bash

Copy code

ssh-keygen -t ed25519 -C "your\_email@example.com"

Ellenőrizd a kulcsokat a következő paranccsal:

bash

Copy code

cat ~/.ssh/id\_ed25519.pub

Ezt a kulcsot másold be a GitHub „SSH and GPG keys” részébe.

**Összefoglalás**

* **Ellenőrizd a SSH kulcsot** és az SSH agent beállításait.
* **Nézd meg a tűzfal és proxy beállításokat**.
* **Próbálj meg alternatív portot használni** az SSH kapcsolat számára.
* **Frissítsd a Git és SSH klienst**, ha szükséges.

Ha ezek a lépések nem oldják meg a problémát, kérlek, jelezd, és további segítséget nyújtok!

4o mini

agnes/.ssh/id\_ed25519

ChatGPT

Ha a .ssh/id\_ed25519 fájl a helyes helyen van, itt van, hogyan folytathatod az SSH kulcs konfigurálását és ellenőrzését a GitHub-hoz való kapcsolódás érdekében.

**1. SSH Kulcs hozzáadása az SSH Agenthez**

Győződj meg róla, hogy az SSH agent fut, és add hozzá az SSH kulcsot:

bash

Copy code

eval "$(ssh-agent -s)"

ssh-add ~/.ssh/id\_ed25519

Ezután ellenőrizheted, hogy a kulcsot hozzáadtad-e az agenthez:

bash

Copy code

ssh-add -l

**2. Ellenőrizd a SSH Kulcsot**

Győződj meg róla, hogy a kulcs helyesen van formázva és helyén van. A kulcsot a következő paranccsal tekintheted meg:

bash

Copy code

cat ~/.ssh/id\_ed25519.pub

A kulcsot a GitHub „SSH and GPG keys” oldalán kell hozzáadni. Ha még nem tetted meg, itt van, hogyan adhatod hozzá:

**GitHub SSH Kulcs Hozzáadása**

1. **Jelentkezz be GitHub Fiókodba**
   * Lépj be a GitHub fiókodba a [GitHub weboldalán](https://github.com).
2. **Navigálj a Beállításokhoz**
   * Kattints a profilképedre a jobb felső sarokban.
   * Válaszd ki a „**Settings**” (Beállítások) opciót.
3. **Menj az SSH és GPG Kulcsokhoz**
   * A bal oldali menüben keresd meg a „**SSH and GPG keys**” szekciót.
   * Kattints a „**New SSH key**” gombra.
4. **Add hozzá az SSH Kulcsot**
   * Írd be a kulcs nevét a „**Title**” mezőbe.
   * Másold be az SSH kulcsot, amit a cat ~/.ssh/id\_ed25519.pub parancs mutat, a „**Key**” mezőbe.
   * Kattints a „**Add SSH key**” gombra.

**3. SSH Kapcsolat Tesztelése**

Teszteld az SSH kapcsolatot a következő paranccsal:

bash

Copy code

ssh -T git@github.com

Ha a kapcsolat sikeres, a következő üzenetet kell látnod:

plaintext

Copy code

Hi username! You've successfully authenticated, but GitHub does not provide shell access.

**4. Alternatív Port Használata**

Ha a 22-es port blokkolva van a hálózatodon, próbálkozz az SSH alternatív portján (443):

bash

Copy code

ssh -T -p 443 git@ssh.github.com

**5. Ellenőrizd a Tűzfal és Proxy Beállításokat**

A tűzfal vagy proxy beállítások is okozhatják a problémát. Győződj meg róla, hogy a GitHub elérése nem blokkolva van.

**6. Frissítsd a Git és SSH Klienst**

Győződj meg róla, hogy a Git és az SSH kliens a legújabb verziókat használja. Esetleg frissítheted őket, ha szükséges.

**Összefoglalás**

* **Győződj meg róla, hogy az SSH kulcs helyesen van konfigurálva** és hozzá van adva az SSH agenthez.
* **Ellenőrizd, hogy az SSH kulcsot hozzáadtad a GitHub fiókodhoz**.
* **Teszteld az SSH kapcsolatot** a GitHub szerverhez.
* **Próbálj meg alternatív portot** használni, ha szükséges.
* **Ellenőrizd a tűzfal és proxy beállításokat**, ha problémád van a kapcsolat létrehozásával.

Ha további kérdéseid vannak, vagy a probléma továbbra is fennáll, kérlek, jelezd, és további segítséget nyújtok!

4o mini

mi az ssh kulcs? (/c/Users/agnes/.ssh/id\_ed25519):

ChatGPT

Az SSH kulcs egy kriptográfiai kulcspár, amelyet az SSH (Secure Shell) protokoll használ a biztonságos kapcsolatok létrehozásához két számítógép között. A kulcspár két részből áll: egy privát kulcsból és egy nyilvános kulcsból.

**SSH Kulcsok Jellemzői**

1. **Privát Kulcs** (id\_ed25519):
   * **Jelentése**: Ez a kulcs a saját gépeden található, és titokban kell tartani. Ezt a kulcsot nem szabad megosztani senkivel.
   * **Helye**: A privát kulcs általában az ~/.ssh/ könyvtárban található (például /c/Users/agnes/.ssh/id\_ed25519 Windows rendszeren).
   * **Funkció**: Az SSH kapcsolatok során a privát kulcsot használják az autentikációs folyamat során.
2. **Nyilvános Kulcs** (id\_ed25519.pub):
   * **Jelentése**: Ez a kulcs bárkivel megosztható, és azt a szerverek használják az autentikációs folyamat során. Ezt a kulcsot hozzá kell adni a szerverhez, amelyhez csatlakozni szeretnél.
   * **Helye**: A nyilvános kulcs az ~/.ssh/ könyvtárban található (például /c/Users/agnes/.ssh/id\_ed25519.pub Windows rendszeren).
   * **Funkció**: A nyilvános kulcsot az SSH szerverek használják annak ellenőrzésére, hogy a privát kulcs tulajdonosa jogosult-e a kapcsolatra.

**SSH Kulcsok Generálása**

Az SSH kulcsokat általában a ssh-keygen paranccsal generálhatod. Például:

bash

Copy code

ssh-keygen -t ed25519 -C "your\_email@example.com"

Ez létrehoz egy új kulcspárt. A rendszer megkérdezi, hogy hova szeretnéd menteni a kulcsot (alapértelmezett hely: ~/.ssh/id\_ed25519), és lehetőséged lesz egy jelszót is beállítani, amely hozzáad egy extra rétegű biztonságot.

**SSH Kulcs Használata**

1. **Privát Kulcs Hozzáadása az SSH Agenthez**:
   * Az SSH agent egy háttérszolgáltatás, amely a privát kulcsot tárolja és kezeli a hitelesítést az SSH kapcsolat során.
   * Indítsd el az SSH agentet és add hozzá a privát kulcsot:

bash

Copy code

eval "$(ssh-agent -s)"

ssh-add ~/.ssh/id\_ed25519

1. **Nyilvános Kulcs Hozzáadása a GitHub-hoz**:
   * Másold ki a nyilvános kulcs tartalmát:

bash

Copy code

cat ~/.ssh/id\_ed25519.pub

* + Lépj be a GitHub fiókodba, majd navigálj a „**Settings**” > „**SSH and GPG keys**” részhez.
  + Kattints a „**New SSH key**” gombra, add meg a kulcs nevét, és illeszd be a nyilvános kulcsot a „**Key**” mezőbe.

1. **Kapcsolódás a GitHub-hoz**:
   * Ellenőrizd a kapcsolatot a következő paranccsal:

bash

Copy code

ssh -T git@github.com

* + Ha a kapcsolat sikeres, a következő üzenetet kell látnod:

plaintext

Copy code

Hi username! You've successfully authenticated, but GitHub does not provide shell access.

**Összefoglalás**

* **Privát Kulcs**: Titkos, a saját gépeden található.
* **Nyilvános Kulcs**: Megosztható, és a szervereken tárolható.
* **Generálás**: ssh-keygen paranccsal történik.
* **Használat**: A privát kulcsot az SSH agenthez adod hozzá, a nyilvános kulcsot pedig hozzáadod a GitHub-hoz vagy más szerverekhez.

Ha bármilyen kérdésed van a SSH kulcsokkal vagy azok használatával kapcsolatban, szívesen segítek!

4o mini

Enter file in which to save the key (/c/Users/agnes/.ssh/id\_ed25519):

ChatGPT

Amikor az ssh-keygen parancsot futtatod, a rendszer megkérdezi, hogy hova szeretnéd menteni az új SSH kulcsot. Az alapértelmezett hely az ~/.ssh/id\_ed25519, de ha nem szeretnéd ezt az alapértelmezett helyet használni, akkor itt megadhatsz egy másik elérési utat is.

**Útmutató az SSH Kulcsok Generálásához**

1. **Futtasd az SSH Kulcs Generáló Parancsot**

Használhatod az alábbi parancsot az új SSH kulcs generálásához:

bash

Copy code

ssh-keygen -t ed25519 -C "your\_email@example.com"

Az -t ed25519 opció az ed25519 típusú kulcsot jelöli, ami biztonságos és modern. Az -C "your\_email@example.com" opció pedig egy megjegyzést ad a kulcshoz, ami általában az e-mail címed.

1. **Amikor a Rendszer Megkérdezi a Fájl Helyét**

Ha nem adsz meg másik elérési utat, az alapértelmezett hely lesz felajánlva:

bash

Copy code

Enter file in which to save the key (/c/Users/agnes/.ssh/id\_ed25519):

* + **Az Alapértelmezett Hely Használata**: Ha nyomsz egy Enter-t, az SSH kulcs a /c/Users/agnes/.ssh/id\_ed25519 helyre lesz mentve. Ez a hely általában megfelelő a legtöbb esetben.
  + **Másik Hely Megadása**: Ha másik helyet szeretnél használni, írj be egy új elérési utat. Például, ha az SSH kulcsot a C:/Users/agnes/ssh\_keys/my\_key helyre szeretnéd menteni, akkor ezt írd be:

bash

Copy code

/c/Users/agnes/ssh\_keys/my\_key

* + **Fájlnevek**: Ha egy meglévő fájl nevére szeretnéd menteni, a rendszer kérni fogja, hogy írd be a fájl új neve előtt a fájl kiterjesztését. Például my\_key helyett my\_key\_ed25519.

1. **Jelszó Beállítása**

A következő lépésben a rendszer megkérdezi, hogy szeretnél-e jelszót beállítani a kulcshoz:

java

Copy code

Enter passphrase (empty for no passphrase):

* + **Jelszó Beállítása**: Írj be egy jelszót, majd nyomj Enter-t. Ezzel egy extra réteg biztonságot adsz a kulcshoz.
  + **Jelszó Nélkül**: Ha nem szeretnél jelszót beállítani, egyszerűen nyomj Enter-t.

1. **SSH Kulcsok Ellenőrzése**

Miután a kulcsokat generáltad, az alábbi parancsokkal ellenőrizheted őket:

* + **Privát Kulcs**: ~/.ssh/id\_ed25519
  + **Nyilvános Kulcs**: ~/.ssh/id\_ed25519.pub

A nyilvános kulcsot a következő parancs segítségével tekintheted meg:

bash

Copy code

cat ~/.ssh/id\_ed25519.pub

Másold ki a nyilvános kulcsot, és add hozzá a GitHub-hoz vagy más szolgáltatókhoz a biztonságos hozzáférés érdekében.

**Összefoglalás**

* **Alapértelmezett hely**: /c/Users/agnes/.ssh/id\_ed25519 (Windows rendszeren).
* **Másik hely megadása**: Írd be az új elérési utat, ha nem az alapértelmezett helyet szeretnéd használni.
* **Jelszó beállítása**: Opcionális, de növeli a kulcs biztonságát.

Ha további kérdéseid vannak a kulcsok generálásával vagy használatával kapcsolatban, szívesen segítek!

4o mini

$ cat ~/.ssh/id\_ed25519.pub cat: /c/Users/agnes/.ssh/id\_ed25519.pub: No such file or directory

ChatGPT

A No such file or directory hiba azt jelzi, hogy a ~/.ssh/id\_ed25519.pub fájl nem létezik. Ennek a problémának több oka lehet:

1. **SSH Kulcs Nem Lett Generálva**
2. **Helytelen Fájlútvonal**
3. **SSH Kulcs Másik Helyen Van**

**Ellenőrizd az SSH Kulcsok Létét**

1. **Járj utána, hogy léteznek-e SSH kulcsok az alapértelmezett helyen**

Navigálj az .ssh könyvtárba és ellenőrizd, hogy létezik-e a id\_ed25519 és id\_ed25519.pub fájl:

bash

Copy code

cd ~/.ssh

ls -la

Ez a parancs listázza az .ssh könyvtár tartalmát. Ha itt nem látod a kulcsokat, akkor lehet, hogy nem lettek generálva.

1. **Ha nem találod a kulcsokat, generálj újakat**

Használj ssh-keygen parancsot az SSH kulcsok generálásához. Győződj meg róla, hogy a megfelelő könyvtárban vagy:

bash

Copy code

ssh-keygen -t ed25519 -C "your\_email@example.com"

Amikor megkérdezi a fájl elérési útját, hagyd meg az alapértelmezett helyet, ha nem szeretnél más helyet megadni:

bash

Copy code

Enter file in which to save the key (/c/Users/agnes/.ssh/id\_ed25519):

Ha a rendszer nem találja az alapértelmezett helyet, vagy egy másik könyvtárat szeretnél használni, megadhatsz egy új helyet is.

1. **Ellenőrizd az SSH Kulcsok Meglévőségét**

Miután generáltad a kulcsokat, ellenőrizd újra az .ssh könyvtár tartalmát, hogy megbizonyosodj róla, a kulcsok megfelelően létrejöttek:

bash

Copy code

ls -la ~/.ssh

1. **Fájlok Másik Helyen**

Ha nem az alapértelmezett helyen generáltad a kulcsokat, navigálj ahhoz a helyhez, ahol elmentetted őket, és ott ellenőrizd a kulcsokat.

1. **Jelentkezz be a GitHubra**

Amint megtaláltad vagy létrehoztad a kulcsokat, ellenőrizd, hogy hozzáadtad-e őket a GitHub fiókodhoz:

* + Másold ki a nyilvános kulcs tartalmát (például a id\_ed25519.pub fájl tartalmát).
  + Navigálj a GitHub „**Settings**” > „**SSH and GPG keys**” szekcióba.
  + Kattints a „**New SSH key**” gombra, és add hozzá a kulcsot.

**További Ellenőrzési Lépések**

* **Ellenőrizd a kulcsok generálásához használt parancsot**. Ha másik típusú kulcsot generáltál (például rsa helyett ed25519), győződj meg róla, hogy az az általad használt könyvtárban található.
* **Nézd meg a fájlokat és jogosultságokat**. Az .ssh könyvtár jogosultságai lehetnek problémásak, ha nem tudsz hozzáférni a fájlokhoz. Győződj meg róla, hogy az .ssh könyvtár és a kulcsfájlok jogosultságai megfelelőek:

bash

Copy code

chmod 700 ~/.ssh

chmod 600 ~/.ssh/id\_ed25519

chmod 644 ~/.ssh/id\_ed25519.pub

**Összefoglalás**

* **Ellenőrizd az SSH kulcsokat** az alapértelmezett helyen vagy a megadott helyen.
* **Generálj új kulcsokat**, ha a fájlok nem léteznek.
* **Ellenőrizd a fájlokat és jogosultságokat**.
* **Hozzáadd a nyilvános kulcsot a GitHub fiókodhoz**.

Ha további kérdéseid vannak, vagy a probléma továbbra is fennáll, jelezd, és segítek a megoldásban!

4o mini

agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $ cat ~/.ssh/id\_ed25519.pub cat: /c/Users/agnes/.ssh/id\_ed25519.pub: No such file or directory agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $ ^C agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $ cd ~/.ssh bash: cd: /c/Users/agnes/.ssh: No such file or directory agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $ ls -la total 27452 drwxr-xr-x 1 agnes 197611 0 Jul 20 02:21 ./ drwxr-xr-x 1 agnes 197611 0 Jul 20 00:24 ../ drwxr-xr-x 1 agnes 197611 0 Jul 20 02:26 .git/ -rw-r--r-- 1 agnes 197611 3557299 Sep 2 2023 20201031\_141942.jpg -rw-r--r-- 1 agnes 197611 3562654 Jul 20 00:47 Csatlos\_Green\_V04\_001.jpg -rw-r--r-- 1 agnes 197611 7025 Jul 20 01:33 index.html -rw-r--r-- 1 agnes 197611 2965 Jul 20 02:15 info.txt -rw-r--r-- 1 agnes 197611 15157361 Mar 5 2019 project1.jpg -rw-r--r-- 1 agnes 197611 3390909 Mar 20 2018 project2.jpg -rw-r--r-- 1 agnes 197611 417850 Jul 20 00:57 project3.jpg -rw-r--r-- 1 agnes 197611 1984530 Jul 20 00:53 project4.jpg -rw-r--r-- 1 agnes 197611 1127 Jul 20 00:25 styles.css agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main)

ChatGPT

A No such file or directory hiba azt jelzi, hogy az .ssh könyvtár nem létezik, vagy nem a megfelelő helyen keresed. Az alábbi lépések segítenek megoldani ezt a problémát:

**1. Ellenőrizd az SSH könyvtár létezését**

Lehet, hogy az .ssh könyvtár nem létezik a felhasználói könyvtáradban, vagy egy másik helyen van. Ellenőrizd, hogy létezik-e az .ssh könyvtár a felhasználói könyvtáradban:

bash

Copy code

cd ~

ls -la

Ezzel a parancssal látni fogod a felhasználói könyvtárad tartalmát. Ha itt nincs .ssh könyvtár, akkor az még nem lett létrehozva.

**2. SSH Kulcsok Generálása**

Ha az .ssh könyvtár nem létezik, az alábbi lépésekkel generálhatsz SSH kulcsokat, és létrehozhatod az .ssh könyvtárat:

1. **SSH Kulcs Generálása**

Futtasd a következő parancsot az SSH kulcs generálásához:

bash

Copy code

ssh-keygen -t ed25519 -C "your\_email@example.com"

Ha nem adsz meg másik helyet, az alapértelmezett helyen (~/.ssh/id\_ed25519 és ~/.ssh/id\_ed25519.pub) fogja menteni a kulcsokat.

1. **Könyvtár Ellenőrzése**

Miután futtattad az ssh-keygen parancsot, ellenőrizd az .ssh könyvtárat:

bash

Copy code

cd ~/.ssh

ls -la

Ha a kulcsok sikeresen létrejöttek, itt kell látnod a id\_ed25519 és id\_ed25519.pub fájlokat.

**3. SSH Kulcsok Másolása**

Ha sikerült létrehozni a kulcsokat, másold ki a nyilvános kulcsot:

bash

Copy code

cat ~/.ssh/id\_ed25519.pub

Ezután add hozzá ezt a kulcsot a GitHub-hoz:

* **GitHub-on**:
  + Navigálj a GitHub „**Settings**” > „**SSH and GPG keys**” szekcióba.
  + Kattints a „**New SSH key**” gombra.
  + Illeszd be a nyilvános kulcsot (amit a cat parancs kiadott) a „**Key**” mezőbe, és add meg a kulcs nevét.

**4. Ha SSH Kulcsot Már Létrehoztál Másik Helyen**

Ha tudod, hogy az SSH kulcsok már léteznek valahol a rendszereden, de nem az alapértelmezett helyen, navigálj ahhoz a könyvtárhoz, ahol elmentetted őket, és ott futtasd a cat parancsot. Például:

bash

Copy code

cd /path/to/your/keys

cat id\_ed25519.pub

**5. Hálózati Problémák Ellenőrzése**

Mivel a git push parancsnál hálózati problémát tapasztaltál, győződj meg róla, hogy:

* Az internetkapcsolatod működik.
* Nincsenek tűzfalak vagy proxyk, amelyek blokkolják a GitHub-hoz való hozzáférést.
* A GitHub szerverek elérhetők.

Ha a probléma továbbra is fennáll, próbálj meg csatlakozni egy másik hálózaton keresztül, vagy használd a GitHub Web felületet az SSH kulcsok hozzáadásához.

**Összefoglalás**

* **Ellenőrizd** az .ssh könyvtár létezését a felhasználói könyvtárban.
* **Generálj új SSH kulcsokat**, ha szükséges.
* **Add hozzá a nyilvános kulcsot** a GitHub-hoz.
* **Ellenőrizd** az internetkapcsolatot és a hálózati beállításokat, ha problémákat tapasztalsz.

Ha további kérdéseid vannak vagy elakadtál valamelyik lépésben, kérlek, jelezd, és szívesen segítek tovább!

4o mini

agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $ cd ~ agnes@Andras-Laptop MINGW64 ~ $ ls -la total 218164 drwxr-xr-x 1 agnes 197611 0 Jul 20 02:29 ./ drwxr-xr-x 1 agnes 197611 0 Oct 27 2023 ../ drwxr-xr-x 1 agnes 197611 0 May 31 2021 .abevjava/ -rw-r--r-- 1 agnes 197611 773 Jul 20 02:16 .bash\_history -rw-r--r-- 1 agnes 197611 57 Jul 19 23:41 .gitconfig -rw-r--r-- 1 agnes 197611 20 Jul 20 02:26 .lesshst drwxr-xr-x 1 agnes 197611 0 Nov 5 2020 .thumbnails/ -rw-r--r-- 1 agnes 197611 88 Feb 25 2021 .ubnt-discovery.properties drwxr-xr-x 1 agnes 197611 0 Oct 30 2022 '3D Objects'/ drwxr-xr-x 1 agnes 197611 0 Oct 29 2022 AppData/ lrwxrwxrwx 1 agnes 197611 30 Oct 29 2022 'Application Data' -> /c/Users/agnes/AppData/Roaming/ drwxr-xr-x 1 agnes 197611 0 Oct 7 2020 Autodesk/ drwxr-xr-x 1 agnes 197611 0 Jun 12 2023 'Calibre könyvtár'/ drwxr-xr-x 1 agnes 197611 0 Oct 30 2022 Contacts/ lrwxrwxrwx 1 agnes 197611 58 Oct 29 2022 Cookies -> /c/Users/agnes/AppData/Local/Microsoft/Windows/INetCookies/ drwxr-xr-x 1 agnes 197611 0 Sep 5 2023 Desktop/ drwxr-xr-x 1 agnes 197611 0 Jul 18 16:34 Documents/ drwxr-xr-x 1 agnes 197611 0 Jul 20 00:53 Downloads/ drwxr-xr-x 1 agnes 197611 0 Oct 30 2022 Favorites/ drwxr-xr-x 1 agnes 197611 0 Apr 17 20:42 GRAPHISOFT/ drwxr-xr-x 1 agnes 197611 0 Oct 30 2022 Links/ lrwxrwxrwx 1 agnes 197611 28 Oct 29 2022 'Local Settings' -> /c/Users/agnes/AppData/Local/ drwxr-xr-x 1 agnes 197611 0 Oct 30 2022 Music/ lrwxrwxrwx 1 agnes 197611 24 Oct 29 2022 'My Documents' -> /c/Users/agnes/Documents/ -rw-r--r-- 1 agnes 197611 14155776 Jul 18 23:11 NTUSER.DAT -rw-r--r-- 1 agnes 197611 65536 Oct 29 2022 NTUSER.DAT{3402a1dc-57dc-11ed-9ebc-843497222191}.TM.blf -rw-r--r-- 1 agnes 197611 524288 Oct 29 2022 NTUSER.DAT{3402a1dc-57dc-11ed-9ebc-843497222191}.TMContainer00000000000000000001.regtrans-ms -rw-r--r-- 1 agnes 197611 524288 Oct 29 2022 NTUSER.DAT{3402a1dc-57dc-11ed-9ebc-843497222191}.TMContainer00000000000000000002.regtrans-ms lrwxrwxrwx 1 agnes 197611 66 Oct 29 2022 NetHood -> '/c/Users/agnes/AppData/Roaming/Microsoft/Windows/Network Shortcuts'/ drwxr-xr-x 1 agnes 197611 0 Dec 7 2020 OpenVPN/ drwxr-xr-x 1 agnes 197611 0 Jul 16 08:40 Pictures/ lrwxrwxrwx 1 agnes 197611 66 Oct 29 2022 PrintHood -> '/c/Users/agnes/AppData/Roaming/Microsoft/Windows/Printer Shortcuts'/ lrwxrwxrwx 1 agnes 197611 55 Oct 29 2022 Recent -> /c/Users/agnes/AppData/Roaming/Microsoft/Windows/Recent/ drwxr-xr-x 1 agnes 197611 0 Oct 30 2022 'Saved Games'/ drwxr-xr-x 1 agnes 197611 0 Oct 30 2022 Searches/ lrwxrwxrwx 1 agnes 197611 55 Oct 29 2022 SendTo -> /c/Users/agnes/AppData/Roaming/Microsoft/Windows/SendTo/ lrwxrwxrwx 1 agnes 197611 59 Oct 29 2022 'Start Menu' -> '/c/Users/agnes/AppData/Roaming/Microsoft/Windows/Start Menu'/ lrwxrwxrwx 1 agnes 197611 58 Oct 29 2022 Templates -> /c/Users/agnes/AppData/Roaming/Microsoft/Windows/Templates/ drwxr-xr-x 1 agnes 197611 0 Feb 25 2021 'Ubiquiti UniFi'/ drwxr-xr-x 1 agnes 197611 0 Nov 11 2023 Videos/ drwxr-xr-x 1 agnes 197611 0 May 31 2021 abevjava/ -rw-r--r-- 1 agnes 197611 5158 Feb 28 00:03 abevjava\_agnes.log -rwxr-xr-x 1 agnes 197611 200998888 Jul 29 2022 iTunes64Setup.exe\* -rw-r--r-- 1 agnes 197611 3412992 Oct 29 2022 ntuser.dat.LOG1 -rw-r--r-- 1 agnes 197611 3551232 Oct 29 2022 ntuser.dat.LOG2 -rw-r--r-- 1 agnes 197611 20 Oct 30 2022 ntuser.ini agnes@Andras-Laptop MINGW64 ~

ChatGPT

Úgy tűnik, hogy az .ssh könyvtár nem létezik a felhasználói könyvtáradban. Ha az .ssh könyvtárat és annak tartalmát szeretnéd létrehozni, kövesd az alábbi lépéseket:

**1. SSH Kulcsok Generálása**

1. **Nyisd meg a terminált** (amit használni szoktál).
2. **Futtasd az SSH kulcs generáló parancsot** az alábbiak szerint:

bash

Copy code

ssh-keygen -t ed25519 -C "your\_email@example.com"

Cseréld ki a "your\_email@example.com" részt a saját e-mail címedre. Ha az alapértelmezett helyen (/c/Users/agnes/.ssh/id\_ed25519) szeretnéd menteni a kulcsokat, nyomj Entert a Enter file in which to save the key kérdésnél.

1. **Kövesd az utasításokat** a jelszó megadásához (opcionális).
   * **Amikor azt kérdezi, hogy "Enter passphrase"**, itt adhatsz meg egy jelszót a kulcs védelméhez. Ha nem akarsz jelszót, nyomj Entert.

**2. Ellenőrizd a Kulcsok Létrejöttét**

Ha a kulcsok sikeresen létrejöttek, ellenőrizd, hogy a fájlok léteznek az .ssh könyvtárban:

bash

Copy code

cd ~/.ssh

ls -la

Ha sikeresen létrejöttek a kulcsok, látnod kell a id\_ed25519 és id\_ed25519.pub fájlokat.

**3. Másold Ki a Nyilvános Kulcsot**

Használd a cat parancsot a nyilvános kulcs megtekintéséhez:

bash

Copy code

cat ~/.ssh/id\_ed25519.pub

Másold ki a megjelenő szöveget. Ez az SSH kulcs fogja azonosítani a gépedet a GitHub-on.

**4. SSH Kulcs Hozzáadása a GitHub-hoz**

1. **Jelentkezz be a GitHub fiókodba.**
2. **Navigálj a GitHub „Settings” > „SSH and GPG keys” oldalra.**
3. **Kattints a „New SSH key” gombra.**
4. **Illeszd be** a id\_ed25519.pub fájl tartalmát a „**Key**” mezőbe.
5. **Add meg** a kulcs nevét a „**Title**” mezőben (például: „My Laptop SSH Key”).
6. **Kattints a „Add SSH key” gombra** a kulcs hozzáadásához.

**5. Ellenőrizd az SSH Kapcsolatot**

Miután hozzáadtad az SSH kulcsot a GitHub-hoz, ellenőrizd, hogy a kapcsolat működik-e:

bash

Copy code

ssh -T git@github.com

Ha sikeresen csatlakozol, látni fogsz egy üzenetet, amely megerősíti, hogy a kulcsot sikeresen hozzáadtad.

**6. Git Push Parancs**

Most már próbálkozhatsz a git push parancsot:

bash

Copy code

git push -u origin main

**Összefoglalva:**

* **Generálj új SSH kulcsot**, ha az .ssh könyvtár nem létezik.
* **Ellenőrizd a kulcsok létrejöttét**.
* **Add hozzá** a nyilvános kulcsot a GitHub-hoz.
* **Ellenőrizd az SSH kapcsolatot**.
* **Használj Git parancsokat**, mint a git push, hogy frissítsd a távoli repository-t.

Ha bármilyen kérdésed van a lépések egyikével kapcsolatban, vagy ha további problémákat tapasztalsz, jelezd bátran!

4o mini

agnes@Andras-Laptop MINGW64 ~ $ ssh-keygen -t ed25519 -C "agnes.habram@outlook.com" Generating public/private ed25519 key pair. Enter file in which to save the key (/c/Users/agnes/.ssh/id\_ed25519):

ChatGPT

Az SSH kulcs generálásához szükséges lépéseket most részletesebben bemutatom:

**SSH Kulcs Generálása**

1. **Futtasd a következő parancsot** az SSH kulcsok generálásához:

bash

Copy code

ssh-keygen -t ed25519 -C "agnes.habram@outlook.com"

1. **A következő kérdésnél** (Enter file in which to save the key), ha az alapértelmezett helyen (/c/Users/agnes/.ssh/id\_ed25519) szeretnéd menteni a kulcsot, egyszerűen nyomj Enter-t.

plaintext

Copy code

Enter file in which to save the key (/c/Users/agnes/.ssh/id\_ed25519):

* + Ha más helyet szeretnél megadni, írd be az elérési utat és nyomj Enter-t.

1. **A következő kérdésnél** (Enter passphrase), adhatsz meg egy jelszót a kulcs védelméhez, vagy ha nem szeretnél jelszót, nyomj Enter-t.

plaintext

Copy code

Enter passphrase (empty for no passphrase):

* + Ha jelszót szeretnél beállítani, írd be, majd nyomj Enter-t.
  + Ha nem akarsz jelszót, csak nyomj Enter-t.

1. **Ismételd meg** a jelszó beírását, ha jelszót adtál meg.

plaintext

Copy code

Enter same passphrase again:

**Kulcsok Ellenőrzése**

Miután a kulcsok generálása befejeződött, ellenőrizd, hogy az SSH könyvtár tartalmazza-e a szükséges fájlokat:

bash

Copy code

cd ~/.ssh

ls -la

Két fájlt kell látnod:

* id\_ed25519 (a privát kulcs)
* id\_ed25519.pub (a nyilvános kulcs)

**Nyilvános Kulcs Megtekintése**

A nyilvános kulcsot az alábbi paranccsal tekintheted meg:

bash

Copy code

cat ~/.ssh/id\_ed25519.pub

A kimenet a nyilvános kulcs tartalmát mutatja, amit a GitHub-on kell hozzáadnod. Másold ki ezt a szöveget.

**SSH Kulcs Hozzáadása a GitHub-hoz**

1. **Jelentkezz be a GitHub fiókodba**.
2. **Navigálj a GitHub beállítások oldalára**: [GitHub SSH and GPG keys](https://github.com/settings/keys).
3. **Kattints a „New SSH key” gombra**.
4. **Illeszd be a id\_ed25519.pub fájl tartalmát** a „Key” mezőbe.
5. **Add meg a kulcs nevét** a „Title” mezőben (például: „My Laptop SSH Key”).
6. **Kattints a „Add SSH key” gombra**.

**SSH Kapcsolat Ellenőrzése**

Most ellenőrizd, hogy az SSH kapcsolat működik-e a GitHub-bal:

bash

Copy code

ssh -T git@github.com

Ha minden jól van beállítva, látni fogsz egy üzenetet, amely megerősíti, hogy sikeresen csatlakoztál a GitHub-hoz.

**Git Push Parancs**

Miután hozzáadtad a nyilvános kulcsot, próbáld meg újra a git push parancsot:

bash

Copy code

git push -u origin main

Ha további kérdéseid vagy problémáid vannak, ne habozz jelezni!

4o mini

$ cd ~/.ssh agnes@Andras-Laptop MINGW64 ~/.ssh $ ls -la total 18 drwxr-xr-x 1 agnes 197611 0 Jul 20 02:46 ./ drwxr-xr-x 1 agnes 197611 0 Jul 20 02:46 ../ -rw-r--r-- 1 agnes 197611 419 Jul 20 02:46 id\_ed25519 -rw-r--r-- 1 agnes 197611 106 Jul 20 02:46 id\_ed25519.pub

ChatGPT

Jól néz ki! Az SSH kulcsaid jelenleg ott vannak:

* **Privát kulcs:** id\_ed25519
* **Nyilvános kulcs:** id\_ed25519.pub

Most nézzük meg, hogyan kell hozzáadni a nyilvános kulcsot a GitHub-hoz, és hogyan tudod ellenőrizni, hogy a kapcsolat működik-e.

**Nyilvános Kulcs Másolása**

1. **Nézd meg a nyilvános kulcs tartalmát:**

bash

Copy code

cat ~/.ssh/id\_ed25519.pub

Ezzel megjelenik a nyilvános kulcs szövege. Másold ki ezt a szöveget, mert szükséged lesz rá a GitHub-on.

**SSH Kulcs Hozzáadása a GitHub-hoz**

1. **Jelentkezz be a GitHub fiókodba**.
2. **Navigálj a GitHub beállítások oldalára**: [GitHub SSH and GPG keys](https://github.com/settings/keys).
3. **Kattints a „New SSH key” gombra**.
4. **Illeszd be a nyilvános kulcsot a „Key” mezőbe**.
5. **Add meg a kulcs nevét** a „Title” mezőben (például: „My Laptop SSH Key”).
6. **Kattints a „Add SSH key” gombra**.

**SSH Kapcsolat Ellenőrzése**

Miután hozzáadtad a kulcsot a GitHub-hoz, ellenőrizd, hogy az SSH kapcsolat működik-e:

bash

Copy code

ssh -T git@github.com

Ha sikeresen csatlakoztál, egy üzenetet fogsz látni, amely így nézhet ki:

plaintext

Copy code

Hi <username>! You've successfully authenticated, but GitHub does not provide shell access.

**Git Push Parancs**

Most próbáld meg újra a git push parancsot a projekted könyvtárában:

bash

Copy code

git push -u origin main

Ez a parancs feltölti a helyi változásokat a GitHub tárolóra.

**Ha További Problémáid Vannak**

* Ha továbbra is problémákba ütközöl, ellenőrizd a tűzfal beállításaidat, vagy próbálj meg másik hálózaton csatlakozni.
* Győződj meg arról, hogy a GitHub URL megfelelő és elérhető.
* Ellenőrizd, hogy a helyi Git tároló helyesen van konfigurálva.

Ha bármi más kérdésed van, nyugodtan jelezd!

4o mini

$ cat ~/.ssh/id\_ed25519.pub ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIHTARDSI+r+Md8cSR+c6pzon/MwjJMY1ld+tCSzR4lG0 agnes.habram@outlook.com

ChatGPT

Nagyszerű! Most, hogy megvan a nyilvános SSH kulcsod, a következő lépés, hogy hozzáadd a GitHub fiókodhoz. Itt vannak a pontos lépések:

**Nyilvános Kulcs Hozzáadása a GitHub-hoz**

1. **Jelentkezz be GitHub fiókodba.**
2. **Navigálj a GitHub beállítások oldalára:**

[GitHub SSH and GPG keys](https://github.com/settings/keys)

1. **Kattints a „New SSH key” gombra.**
2. **A „Title” mezőbe** adj meg egy nevet az SSH kulcsodnak. Például: „My Laptop SSH Key”.
3. **A „Key” mezőbe** illeszd be a nyilvános kulcsot, amit az alábbi parancs kimenetében láttál:

plaintext

Copy code

ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIHTARDSI+r+Md8cSR+c6pzon/MwjJMY1ld+tCSzR4lG0 agnes.habram@outlook.com

1. **Kattints a „Add SSH key” gombra.**

**SSH Kapcsolat Ellenőrzése**

Miután hozzáadtad a kulcsot, ellenőrizd, hogy az SSH kapcsolat működik-e:

bash

Copy code

ssh -T git@github.com

Ha sikeresen csatlakoztál, látni fogsz egy üzenetet, amely így néz ki:

plaintext

Copy code

Hi <username>! You've successfully authenticated, but GitHub does not provide shell access.

**Git Push Parancs**

Most, hogy a kulcs hozzá van adva, próbáld meg újra feltölteni a projektedet a GitHub-ra:

1. **Navigálj a projekted könyvtárába:**

bash

Copy code

cd /d/\_mentes\_d/adatok/andras/web/weboldal3

1. **Töltsd fel a helyi változásokat:**

bash

Copy code

git push -u origin main

Ha a parancs sikeresen fut, akkor a változásaid feltöltődnek a GitHub tárolódba.

**Ha További Kérdéseid Vannak**

* **Ha problémába ütközöl a git push során,** ellenőrizd, hogy az origin távoli URL helyesen van beállítva és próbáld meg újra.
* **Ha más hibaüzenetet kapsz,** nézd át a hibaüzenetet és annak részleteit, és próbálj meg annak megfelelően lépéseket tenni.

Ne habozz bárminemű kérdéseket feltenni, ha további segítségre van szükséged!

4o mini

ssh -T git@github.com ssh: connect to host github.com port 22: Permission denied

ChatGPT

Ha az ssh -T git@github.com parancs „Permission denied” hibát ad, az általában azt jelenti, hogy az SSH kapcsolat nem tud létrejönni. Ennek több oka lehet, például hálózati vagy tűzfal beállítások, vagy helytelen SSH konfiguráció. Itt van néhány lépés, amellyel megoldhatod a problémát:

**1. Ellenőrizd, hogy az SSH Kulcsok Helyesen Vannak Beállítva**

Győződj meg róla, hogy a privát kulcs helyesen van elhelyezve és olvasható a ~/.ssh/id\_ed25519 fájlban.

**2. SSH Agent Beállítása**

Az SSH agent biztosítja, hogy az SSH kulcsok megfelelően használhatóak legyenek az SSH kapcsolatok során. Indítsd el az SSH agent-et és add hozzá a privát kulcsot:

bash

Copy code

eval $(ssh-agent -s)

ssh-add ~/.ssh/id\_ed25519

**3. Ellenőrizd, hogy Nincs Tűzfal vagy Hálózati Szűrő**

Néhány hálózaton (pl. vállalati vagy iskolai hálózatokon) a port 22-t blokkolhatják. Ha a tűzfal vagy hálózati szabályok blokkolják a portot, próbáld meg az SSH kapcsolatot HTTPS-en keresztül:

bash

Copy code

ssh -T git@ssh.github.com

**4. HTTPS Protokoll Használata**

Ha SSH kapcsolatot nem tudsz létrehozni, használhatsz HTTPS-t is a GitHub eléréséhez. A Git tároló URL-jét HTTPS-re állíthatod be:

1. **Állítsd be az HTTPS URL-t:**

bash

Copy code

git remote set-url origin https://github.com/ahabram/adoszpoly.git

1. **Feltöltés HTTPS-en keresztül:**

bash

Copy code

git push -u origin main

**5. SSH Kulcs Ellenőrzése és GitHub Konfiguráció**

1. **Győződj meg róla, hogy a nyilvános kulcs hozzá van adva a GitHub fiókodhoz.**
2. **Ellenőrizd, hogy a GitHub fiókod beállításaiban** a kulcs tényleg hozzá van adva, és hogy a kulcs megfelelően lett bemásolva.

**6. SSH Config Fájl Ellenőrzése**

Ellenőrizd, hogy van-e ~/.ssh/config fájl, és hogy tartalmaz-e megfelelő bejegyzéseket. Ha nem létezik, próbáld meg létrehozni és hozzáadni az alábbi konfigurációt:

plaintext

Copy code

Host github.com

Hostname ssh.github.com

Port 443

User git

Ez a konfiguráció lehetővé teszi, hogy SSH kapcsolaton keresztül port 443-on csatlakozz a GitHubhoz, ami sok esetben átmegy a tűzfalakon, ahol a port 22 le van tiltva.

**7. SSH Kulcs Újra Generálása és Hozzáadása**

Ha a fenti lépések nem segítenek, próbálj meg új SSH kulcsot generálni és hozzáadni a GitHub-hoz:

1. Generálj új SSH kulcsot:

bash

Copy code

ssh-keygen -t ed25519 -C "your\_email@example.com"

1. Másold ki a nyilvános kulcsot, és add hozzá a GitHub-hoz, ahogy korábban is.

**További Ellenőrzési Lépések**

* **Hálózati probléma:** Próbálj meg csatlakozni egy másik hálózaton (pl. mobil adatkapcsolaton).
* **GitHub Állapot:** Ellenőrizd, hogy a GitHub szolgáltatások működnek-e: [GitHub Status](https://www.githubstatus.com/).

Ha továbbra is problémáid vannak, oszd meg a pontos hibaüzenetet és a lépéseket, amiket már kipróbáltál, és további segítséget tudok nyújtani.

4o mini

agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $ eval $(ssh-agent -s) Agent pid 371 agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $ ssh-add ~/.ssh/id\_ed25519 Identity added: /c/Users/agnes/.ssh/id\_ed25519 (agnes.habram@outlook.com) agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $ ^C agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $ ssh -T git@ssh.github.com ssh: connect to host ssh.github.com port 22: Permission denied agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $ ^C agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $ git remote set-url origin https://github.com/ahabram/adoszpoly.git agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $ git push -u origin main fatal: unable to access 'https://github.com/ahabram/adoszpoly.git/': Failed to connect to github.com port 443 after 83 ms: Couldn't connect to server agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $ ^C agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $ Host github.com Hostname ssh.github.com Port 443 User git bash: Host: command not found Hostname: cannot set name to ‘ssh.github.com’: Invalid argument bash: Port: command not found bash: User: command not found agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $

ChatGPT

Úgy tűnik, hogy problémáid vannak az SSH és HTTPS kapcsolatokkal a GitHub-hoz. Az alábbiakban bemutatom, hogyan állíthatod be az SSH kulcsot és a HTTPS kapcsolatot lépésről lépésre, és néhány alternatív megoldást is javaslok a problémád megoldására.

**1. SSH Kulcs és Kapcsolat Ellenőrzése**

Az SSH kulcs beállítását és a GitHub-hoz való kapcsolódást az alábbi lépésekkel tudod ellenőrizni:

**SSH Kulcs Beállítása**

1. **Ellenőrizd az SSH kulcsot:**

Győződj meg arról, hogy az SSH kulcsot sikeresen hozzáadtad az SSH agenthez:

bash

Copy code

ssh-add -l

Ekkor a rendszer vissza kell adja az általad hozzáadott kulcsot.

1. **GitHub SSH Beállítás:**

Ha nem tudsz SSH-n keresztül csatlakozni, próbáld meg az SSH kapcsolatot a port 443-on:

bash

Copy code

ssh -T git@ssh.github.com

Ha ez nem működik, akkor próbálj meg kapcsolódni a port 443-on az alábbi beállításokkal:

plaintext

Copy code

Host github.com

Hostname ssh.github.com

Port 443

User git

Ezt a konfigurációs fájlt az ~/.ssh/config fájlba kell elhelyezned. A fájlt a következőképpen szerkesztheted:

bash

Copy code

nano ~/.ssh/config

Add hozzá a következő tartalmat, majd mentsd el és zárd be a fájlt:

plaintext

Copy code

Host github.com

Hostname ssh.github.com

Port 443

User git

**2. HTTPS Kapcsolat Beállítása**

Ha az SSH kapcsolat nem működik, használhatsz HTTPS-et is:

1. **Állítsd be a távoli URL-t HTTPS-re:**

Már beállítottad a távoli URL-t HTTPS-re, de győződj meg róla, hogy helyesen van beállítva:

bash

Copy code

git remote set-url origin https://github.com/ahabram/adoszpoly.git

1. **Ellenőrizd a hálózati kapcsolatot:**

Győződj meg arról, hogy a port 443 (HTTPS) nincs blokkolva a hálózati beállításaid által. Próbálj meg csatlakozni egy másik hálózaton, például mobil adathálózaton keresztül, hogy teszteld a kapcsolatot.

**3. Állapot Ellenőrzése és Alternatívák**

1. **GitHub Állapot Ellenőrzése:**

Ellenőrizd, hogy a GitHub szolgáltatások működnek-e: [GitHub Status](https://www.githubstatus.com/).

1. **Proxy és VPN Használata:**

Ha tűzfal vagy proxy blokkolja a kapcsolatot, próbálj meg VPN-t használni, vagy beszélj a hálózati adminisztrátoroddal.

1. **Hálózati Beállítások Ellenőrzése:**

Győződj meg róla, hogy a helyi hálózati beállítások nem blokkolják a GitHub-hoz való kapcsolódást. Ha a hálózati beállítások helyesek, próbáld meg az alábbi lépéseket:

bash

Copy code

curl -I https://github.com

Ez megmutatja, hogy el tudsz-e érni a GitHub-ot HTTPS-en keresztül.

**4. GitHub-tól Támogatás Kérése**

Ha a fentiek után sem tudsz csatlakozni, akkor próbálj meg kapcsolatba lépni a GitHub támogatási csapatával, mivel lehet, hogy valami speciális probléma merült fel a fiókoddal vagy a GitHub rendszerével.

Ha további kérdéseid vannak, vagy új hibaüzeneteket kapsz, kérlek, jelezd, és segítek a további lépésekben!

4o mini

agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $ ssh-add -l 256 SHA256:Hu1WMN+06Ye5ak5tNeamQZ8gYTOTD3ABYs+nIfFOLO4 agnes.habram@outlook.com (ED25519)

ChatGPT

Jól látszik, hogy az SSH kulcsod sikeresen hozzá lett adva az SSH agenthez, és működnie kellene a GitHub-hoz való kapcsolódásnak. Azonban a port 22-es kapcsolódási problémát továbbra is tapasztalod. Néhány további lépést javasolnék a probléma elhárítására:

**1. SSH Konfiguráció Ellenőrzése**

Mivel a port 22-es kapcsolódási probléma merült fel, próbálkozzunk az SSH konfiguráció módosításával a port 443 használatára, ha a port 22 blokkolva van a hálózaton:

1. **SSH Config Fájl Létrehozása vagy Szerkesztése:**

Nyisd meg vagy hozd létre a ~/.ssh/config fájlt:

bash

Copy code

nano ~/.ssh/config

1. **Adj hozzá egy új bejegyzést a fájlhoz:**

plaintext

Copy code

Host github.com

Hostname ssh.github.com

Port 443

User git

Mentsd el a fájlt (Ctrl+O, majd Enter) és zárd be (Ctrl+X).

1. **Teszteld az SSH kapcsolatot:**

bash

Copy code

ssh -T git@github.com

Ha a beállítások helyesek, akkor a kapcsolatnak most már működnie kellene.

**2. HTTPS Kapcsolat Tesztelése**

Ha az SSH kapcsolatok továbbra sem működnek, tesztelhetjük a HTTPS kapcsolatot:

1. **Állítsd be a távoli URL-t HTTPS-re (ha még nem tetted volna meg):**

bash

Copy code

git remote set-url origin https://github.com/ahabram/adoszpoly.git

1. **Próbáld meg a push parancsot:**

bash

Copy code

git push -u origin main

Ha a hálózati probléma továbbra is fennáll, a probléma lehet a helyi hálózati beállításokkal.

**3. Ellenőrizd a Hálózati Beállításokat**

1. **Futtass egy curl parancsot a GitHub HTTPS elérhetőségének tesztelésére:**

bash

Copy code

curl -I https://github.com

Ez megmutatja, hogy el tudsz-e érni a GitHub-ot HTTPS-en keresztül. Ha nem, akkor lehet, hogy a hálózati beállítások blokkolják a kapcsolatot.

**4. Proxy és VPN Beállítások**

Ha VPN-t vagy proxy-t használsz, próbáld meg kikapcsolni vagy átkonfigurálni, hogy az ne blokkolja a GitHub-hoz való kapcsolatot.

**5. GitHub Status Ellenőrzése**

Ellenőrizd, hogy a GitHub szolgáltatások működnek-e: [GitHub Status](https://www.githubstatus.com/).

**6. Tűzfal és Biztonsági Beállítások Ellenőrzése**

Ellenőrizd, hogy a tűzfal vagy más biztonsági beállítások nem blokkolják-e a GitHub-hoz való hozzáférést.

**7. Helyi Git Konfiguráció Ellenőrzése**

Ellenőrizd, hogy a git konfiguráció helyesen van-e beállítva:

bash

Copy code

git config --global --get user.name

git config --global --get user.email

Ha szükséges, állítsd be a helyes felhasználónevet és email címet:

bash

Copy code

git config --global user.name "Your Name"

git config --global user.email "you@example.com"

Ha a fenti lépések után sem tudsz csatlakozni, lehet, hogy a probléma a helyi hálózati beállításokkal van. Kérlek, jelezd, ha további kérdéseid vannak, vagy új hibákat tapasztalsz!

4o mini

$ nano ~/.ssh/config agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $ ssh -T git@github.com ssh: connect to host ssh.github.com port 443: Permission denied

ChatGPT

Mivel az SSH kapcsolat továbbra sem működik a ssh.github.com portján, érdemes próbálkozni a következő lépésekkel, hogy megoldjuk a problémát.

**1. GitHub SSH Kulcs Ellenőrzés**

Először is, győződj meg róla, hogy a GitHub SSH kulcs helyesen van beállítva és a kulcsot hozzáadtad a GitHub fiókodhoz.

1. **Nyisd meg a GitHub SSH beállításokat:**
   * Lépj be a GitHub oldaladra.
   * Menj a Settings (Beállítások) -> SSH and GPG keys részhez.
   * Győződj meg róla, hogy a id\_ed25519.pub kulcs ott van hozzáadva.
2. **Ha még nem adtad hozzá a kulcsot, akkor tedd meg:**
   * Másold ki a id\_ed25519.pub tartalmát:

bash

Copy code

cat ~/.ssh/id\_ed25519.pub

* + Másold a kulcs tartalmát a GitHub SSH and GPG keys részébe, és add hozzá új SSH kulcsként.

**2. GitHub SSH Port Ellenőrzése**

Az SSH kapcsolatok problémáit a helyi hálózati beállítások vagy tűzfal is okozhatja, különösen a port 22 vagy 443 blokkolása miatt. Próbáld ki az alábbiakat:

1. **Állítsd be az SSH kapcsolatot a 443-as porton:**

Nyisd meg vagy szerkeszd a ~/.ssh/config fájlt:

bash

Copy code

nano ~/.ssh/config

Add hozzá az alábbi bejegyzést:

plaintext

Copy code

Host github.com

Hostname ssh.github.com

Port 443

User git

Mentsd el a fájlt (Ctrl+O, Enter), és zárd be (Ctrl+X).

1. **Ellenőrizd, hogy a ssh -T git@github.com parancs helyesen működik:**

bash

Copy code

ssh -T git@github.com

Ha ez nem működik, próbálkozz az alábbi parancsokkal is:

bash

Copy code

ssh -vT git@github.com

Ez több információt ad arról, mi történik a kapcsolat során.

**3. HTTPS Kapcsolat Tesztelése**

Ha az SSH kapcsolat nem működik, próbáld meg az HTTPS kapcsolatot:

1. **Állítsd be a távoli URL-t HTTPS-re (ha még nem tetted volna meg):**

bash

Copy code

git remote set-url origin https://github.com/ahabram/adoszpoly.git

1. **Próbáld meg a push parancsot:**

bash

Copy code

git push -u origin main

**4. Hálózati Beállítások Ellenőrzése**

1. **Futtass egy curl parancsot a GitHub HTTPS elérhetőségének tesztelésére:**

bash

Copy code

curl -I https://github.com

Ez megmutatja, hogy el tudsz-e érni a GitHub-ot HTTPS-en keresztül.

1. **Ellenőrizd a tűzfal és hálózati beállításokat:**

Győződj meg róla, hogy a tűzfal nem blokkolja a portokat. A port 443 általában nyitva kell lennie HTTPS forgalom számára.

**5. VPN és Proxy Ellenőrzése**

Ha VPN-t vagy proxy-t használsz, próbáld meg kikapcsolni, és nézd meg, hogy a probléma megszűnik-e.

**6. GitHub Status Ellenőrzése**

Ellenőrizd, hogy a GitHub szolgáltatások működnek-e: [GitHub Status](https://www.githubstatus.com/).

Ha továbbra is problémák merülnek fel, próbálj meg további hibaelhárítást, és oszd meg velem a legújabb hibaüzeneteket.

4o mini

agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $ ssh -T git@github.com ssh: connect to host ssh.github.com port 443: Permission denied agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $ ssh -vT git@github.com OpenSSH\_9.7p1, OpenSSL 3.2.1 30 Jan 2024 debug1: Reading configuration data /c/Users/agnes/.ssh/config debug1: /c/Users/agnes/.ssh/config line 1: Applying options for github.com debug1: Reading configuration data /etc/ssh/ssh\_config debug1: Connecting to ssh.github.com [140.82.121.35] port 443. debug1: connect to address 140.82.121.35 port 443: Permission denied ssh: connect to host ssh.github.com port 443: Permission denied agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $

ChatGPT

Mivel a ssh parancs továbbra is blokkolva van a ssh.github.com portján, próbálkozzunk más megoldásokkal a GitHub SSH kapcsolatának helyreállítására.

**1. Ellenőrizd a Hálózati Beállításokat**

1. **Tűzfal és Hálózati Beállítások:**
   * Győződj meg róla, hogy a tűzfal nem blokkolja a portokat.
   * Ha vállalati hálózaton vagy, előfordulhat, hogy a tűzfal blokkolja az SSH kapcsolatokat. Próbálj meg csatlakozni egy másik hálózathoz, például otthoni Wi-Fi-n keresztül.
2. **Proxy és VPN:**
   * Ha proxy-t vagy VPN-t használsz, próbáld meg kikapcsolni, és nézd meg, hogy ez megoldja-e a problémát.

**2. GitHub SSH Kulcs Ellenőrzése**

1. **GitHub SSH Kulcs Hozzáadása:**
   * Győződj meg róla, hogy az SSH kulcsot helyesen adtad hozzá a GitHub fiókhoz:
     + Nyisd meg a GitHub-on a Settings -> SSH and GPG keys részt.
     + Győződj meg róla, hogy az új kulcsot hozzáadtad.

**3. HTTPS Használata**

Ha az SSH továbbra is problémát okoz, használhatod az HTTPS kapcsolatot a push műveletekhez:

1. **Állítsd be a távoli URL-t HTTPS-re:**

bash

Copy code

git remote set-url origin https://github.com/ahabram/adoszpoly.git

1. **Próbáld meg újra a push parancsot:**

bash

Copy code

git push -u origin main

**4. Hálózati Tesztelés**

1. **Curl parancs futtatása GitHub HTTPS elérhetőségének ellenőrzésére:**

bash

Copy code

curl -I https://github.com

Ellenőrizd, hogy a GitHub elérhető-e HTTPS-en keresztül.

1. **Ping teszt GitHub szerverekre:**

bash

Copy code

ping github.com

Ez ellenőrzi, hogy a géped képes-e kommunikálni a GitHub szerverekkel.

**5. GitHub Status Ellenőrzése**

Ellenőrizd, hogy a GitHub szolgáltatások működnek-e: [GitHub Status](https://www.githubstatus.com/).

**6. SSH Konfiguráció Ellenőrzése**

Győződj meg róla, hogy az ~/.ssh/config fájl helyesen van beállítva:

1. **Nyisd meg a ~/.ssh/config fájlt:**

bash

Copy code

nano ~/.ssh/config

1. **Győződj meg róla, hogy az alábbi bejegyzés van benne:**

plaintext

Copy code

Host github.com

Hostname ssh.github.com

Port 443

User git

1. **Mentsd el a fájlt (Ctrl+O, Enter), és zárd be (Ctrl+X).**

**7. SSH Agent Ellenőrzése**

1. **Győződj meg róla, hogy az SSH agent fut:**

bash

Copy code

eval $(ssh-agent -s)

1. **Add hozzá az SSH kulcsot az agenthez:**

bash

Copy code

ssh-add ~/.ssh/id\_ed25519

**8. GitHub CLI Tesztelése**

Használhatod a GitHub CLI-t (gh), ha telepítve van, a gh auth status paranccsal ellenőrizheted az autentikációs állapotot.

bash

Copy code

gh auth status

**9. GitHub Support**

Ha ezek után is problémák merülnek fel, és biztos vagy benne, hogy a helyi hálózat nem blokkolja a GitHub SSH kapcsolatokat, lehet, hogy érdemes felvenni a kapcsolatot a GitHub support-tal, hogy segítsenek a probléma megoldásában.

Ha további információra van szükséged, vagy ha új hibaüzenetek jelennek meg, kérlek, oszd meg velem!

4o mini

kikapcsoltam a tűzfalat

ChatGPT

Ha kikapcsoltad a tűzfalat és még mindig problémáid vannak a GitHub SSH kapcsolatával, próbálj ki néhány további lépést a probléma megoldására:

**1. SSH Kulcs Újraellenőrzése**

* **Győződj meg róla, hogy az SSH kulcsok megfelelően vannak beállítva és hozzáadva a GitHub-hoz:**
  1. **Másold az SSH kulcsot:**

bash

Copy code

cat ~/.ssh/id\_ed25519.pub

* 1. **Lépj be a GitHub fiókodba.**
  2. **Navigálj a Settings -> SSH and GPG keys részhez.**
  3. **Ha szükséges, add hozzá az új kulcsot a New SSH key gombbal.**

**2. HTTPS Kapcsolat Tesztelése**

Mivel az SSH kapcsolattal problémák vannak, ellenőrizd az HTTPS kapcsolatot:

* **Állítsd be a távoli URL-t HTTPS-re:**

bash

Copy code

git remote set-url origin https://github.com/ahabram/adoszpoly.git

* **Próbáld meg újra a push parancsot:**

bash

Copy code

git push -u origin main

**3. GitHub HTTPS Tesztelése**

* **Próbáld meg a GitHub-ot elérni HTTPS-en keresztül a curl paranccsal:**

bash

Copy code

curl -I https://github.com

* + **Ha nem kapsz választ, akkor lehet, hogy hálózati probléma van, vagy a GitHub szolgáltatása nem elérhető.**

**4. Hálózati Kapcsolatok Ellenőrzése**

* **Pingeld meg a GitHub szervereket:**

bash

Copy code

ping github.com

* **Ellenőrizd a GitHub státuszoldalát:**
  + [GitHub Status](https://www.githubstatus.com/)

**5. SSH Agent és Kulcsok Ellenőrzése**

* **Győződj meg róla, hogy az SSH agent fut és a kulcs hozzá van adva:**

bash

Copy code

eval $(ssh-agent -s)

ssh-add ~/.ssh/id\_ed25519

* **Ellenőrizd, hogy a kulcs hozzá lett-e adva:**

bash

Copy code

ssh-add -l

**6. SSH Config Ellenőrzése**

* **Nyisd meg és ellenőrizd a ~/.ssh/config fájlt:**

bash

Copy code

nano ~/.ssh/config

* **Győződj meg róla, hogy helyes bejegyzéseket tartalmaz:**

plaintext

Copy code

Host github.com

Hostname ssh.github.com

Port 443

User git

**7. SSH Verbose Mód**

* **Futtass ssh parancsot verbose módban a hibaelhárításhoz:**

bash

Copy code

ssh -vT git@github.com

* + **Ellenőrizd a kimenetet, hogy milyen hibaüzenetek jelennek meg.**

**8. GitHub CLI Tesztelése**

* **Ha telepítve van a GitHub CLI (gh), ellenőrizd az autentikációs állapotot:**

bash

Copy code

gh auth status

**9. További Lépések**

Ha a fentiek nem oldják meg a problémát, érdemes lehet újraindítani a gépet, ellenőrizni a hálózati beállításokat, vagy felvenni a kapcsolatot a GitHub támogatásával.

Ha további kérdéseid vannak, vagy új hibaüzenetek jelennek meg, kérlek, oszd meg velem!

4o mini

$ nano ~/.ssh/config agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $ ssh -T git@github.com The authenticity of host '[ssh.github.com]:443 ([140.82.121.36]:443)' can't be established. ED25519 key fingerprint is SHA256:+DiY3wvvV6TuJJhbpZisF/zLDA0zPMSvHdkr4UvCOqU. This key is not known by any other names. Are you sure you want to continue connecting (yes/no/[fingerprint])?

ChatGPT

Ez a hibaüzenet azt jelenti, hogy az SSH kliensed először találkozik a GitHub szerverrel ezen a porton, és meg kell erősítened a kapcsolatot. Itt van, amit tenned kell:

**1. Kapcsolat Elfogadása**

Az üzenetben látható ED25519 key fingerprint azonosítja a GitHub szerverét. A következő lépés az, hogy megerősíted az SSH kapcsolatot.

**Írd be yes a megerősítéshez:**

plaintext

Copy code

Are you sure you want to continue connecting (yes/no/[fingerprint])? yes

**2. SSH Config Ellenőrzése**

Győződj meg arról, hogy az SSH config fájl helyesen van beállítva. Az SSH config fájlban az alábbi bejegyzésnek kell szerepelnie:

plaintext

Copy code

Host github.com

Hostname ssh.github.com

Port 443

User git

**3. SSH Agent Ellenőrzése**

Győződj meg arról, hogy az SSH kulcsot hozzáadtad az SSH agenthez:

bash

Copy code

eval $(ssh-agent -s)

ssh-add ~/.ssh/id\_ed25519

**4. GitHub Kulcs Ellenőrzése**

Ha a kapcsolat sikeresen létrejön, az SSH parancsnak a következő üzenetet kellene adnia:

plaintext

Copy code

Hi <username>! You've successfully authenticated, but GitHub does not provide shell access.

**5. További Lépesek**

Ha a kapcsolat sikeres, próbálj meg újra git push parancsot futtatni:

bash

Copy code

git push -u origin main

**6. Hibaelhárítás**

Ha a probléma továbbra is fennáll, próbálj meg alábbiakat:

* **Ellenőrizd a GitHub státuszt** a [GitHub Status](https://www.githubstatus.com/) oldalon.
* **Győződj meg arról, hogy nincsenek hálózati problémák** a gépeden vagy a környezetedben.

Kérlek, oszd meg velem, ha a fentiek segítségével sikerült-e megoldani a problémát, vagy ha további hibaüzenetekkel találkozol!

4o mini

agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $ ssh -T git@github.com The authenticity of host '[ssh.github.com]:443 ([140.82.121.36]:443)' can't be established. ED25519 key fingerprint is SHA256:+DiY3wvvV6TuJJhbpZisF/zLDA0zPMSvHdkr4UvCOqU. This key is not known by any other names. Are you sure you want to continue connecting (yes/no/[fingerprint])? yes Warning: Permanently added '[ssh.github.com]:443' (ED25519) to the list of known hosts. Hi ahabram! You've successfully authenticated, but GitHub does not provide shell access.

ChatGPT

Nagyszerű, úgy tűnik, hogy az SSH kapcsolat most már sikeresen létrejött, és a GitHub autentikáció is rendben van!

**Most, hogy az SSH kapcsolat működik, a következő lépéseket végezd el:**

1. **Git Remote URL Ellenőrzése és Beállítása**

Ellenőrizd, hogy a git remote -v parancs a megfelelő URL-t mutatja:

bash

Copy code

git remote -v

Ha az URL nem megfelelő (például, ha HTTPS URL-t használsz, de SSH-t szeretnél használni), állítsd be újra az URL-t:

bash

Copy code

git remote set-url origin git@github.com:ahabram/adoszpoly.git

1. **Feltöltés (Push) a GitHubra**

Próbáld meg újra feltölteni a változtatásokat a GitHub tárolóra:

bash

Copy code

git push -u origin main

**Rövid Áttekintés:**

* **SSH kapcsolat ellenőrzése:** Sikeresen csatlakoztál a GitHubhoz SSH-n keresztül, ami azt jelenti, hogy a kulcsok és a beállítások helyesek.
* **Git Remote URL beállítása:** Győződj meg arról, hogy az SSH URL van beállítva a git remote parancs által.
* **Push parancs:** Az git push parancs használatával a változtatásokat feltöltheted a GitHub tárolóra.

Ha bárminemű problémába ütközöl, kérlek jelezd, és segítek további lépésekben!

4o mini

$ git remote -v origin https://github.com/ahabram/adoszpoly.git (fetch) origin https://github.com/ahabram/adoszpoly.git (push)

ChatGPT

Mivel a távoli URL HTTPS formátumban van, de SSH-t használsz, ezért a következő lépéseket kövesd, hogy SSH URL-t állíts be a GitHub tárolóra:

1. **Állítsd be a megfelelő SSH URL-t:**

Most már frissíteni kell a távoli URL-t, hogy SSH-t használjon a HTTPS helyett. Használd az alábbi parancsot:

bash

Copy code

git remote set-url origin git@github.com:ahabram/adoszpoly.git

1. **Ellenőrizd újra a távoli URL-t:**

Győződj meg róla, hogy az URL most már az SSH formátumú:

bash

Copy code

git remote -v

Az eredménynek így kell kinéznie:

scss

Copy code

origin git@github.com:ahabram/adoszpoly.git (fetch)

origin git@github.com:ahabram/adoszpoly.git (push)

1. **Push a GitHubra:**

Most próbáld meg újra feltölteni a változtatásokat a GitHub tárolóra:

bash

Copy code

git push -u origin main

Ha minden rendben van, a változtatások sikeresen feltöltődnek a GitHub tárolóra.

**További lépések, ha problémák merülnek fel:**

* **Hálózati problémák:** Ha még mindig problémáid vannak a kapcsolattal, ellenőrizd, hogy nincsenek-e hálózati problémák, tűzfal beállítások, vagy proxy beállítások, amelyek blokkolhatják a GitHub-hoz való kapcsolódást.
* **SSH Kulcsok:** Győződj meg róla, hogy az SSH kulcsod megfelelően van hozzáadva a GitHub fiókodhoz és a helyi SSH ügynök is fut.

Ha további kérdéseid vagy problémáid vannak, ne habozz jelezni!

4o mini

$ git remote -v origin https://github.com/ahabram/adoszpoly.git (fetch) origin https://github.com/ahabram/adoszpoly.git (push) agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $ ^C agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $ git remote set-url origin git@github.com:ahabram/adoszpoly.git agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $ git remote -v origin git@github.com:ahabram/adoszpoly.git (fetch) origin git@github.com:ahabram/adoszpoly.git (push)

ChatGPT

Most, hogy az SSH URL-t beállítottad, próbálj meg újra feltölteni a változtatásokat a GitHub tárolóra. Itt van, hogyan:

1. **Push a GitHubra:**

bash

Copy code

git push -u origin main

Ezzel a paranccsal feltöltöd a main ágat a GitHub tárolóra. Az -u opció beállítja az upstream ágat, így a jövőbeni git push és git pull parancsok automatikusan a main ágat fogják használni a origin távoli tárolóra.

1. **Ellenőrizd az állapotot:**

Ha a git push parancs sikeresen lefut, akkor a GitHub tárolódon meg kell jelennie a változtatásoknak. Ellenőrizd a GitHub weboldalon, hogy a fájlok és a commitok helyesen megjelennek-e.

1. **Hibák kezelése:**
   * **Hálózati problémák:** Ha még mindig hibaüzenetet kapsz, például "Couldn't connect to server," győződj meg róla, hogy nincsenek hálózati problémák, proxy beállítások vagy tűzfal, amelyek blokkolhatják a kapcsolatot a GitHubbal.
   * **SSH hozzáférés ellenőrzése:** Futass újra egy SSH tesztet, hogy ellenőrizd a kapcsolatot:

bash

Copy code

ssh -T git@github.com

Ha sikeresen csatlakozol, akkor az üzenetnek valami ilyesminek kell lennie: Hi <your-username>! You've successfully authenticated, but GitHub does not provide shell access.

* + **SSH Kulcs hozzáadása:** Győződj meg róla, hogy az SSH kulcsodat hozzáadtad a GitHub fiókodhoz a GitHub beállítások "SSH and GPG keys" részén keresztül.

1. **Ha további problémák merülnek fel:**

Ha a probléma továbbra is fennáll, ellenőrizd a GitHub dokumentációját a kapcsolódási problémákról, és győződj meg róla, hogy az SSH kulcsod helyesen van beállítva és a GitHub fiókhoz van rendelve.

Ne habozz jelezni, ha további segítségre van szükséged!

4o mini

agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $ git push -u origin main To github.com:ahabram/adoszpoly.git ! [rejected] main -> main (fetch first) error: failed to push some refs to 'github.com:ahabram/adoszpoly.git' hint: Updates were rejected because the remote contains work that you do not hint: have locally. This is usually caused by another repository pushing to hint: the same ref. If you want to integrate the remote changes, use hint: 'git pull' before pushing again. hint: See the 'Note about fast-forwards' in 'git push --help' for details. agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main)

ChatGPT

A hibaüzenet azt jelzi, hogy a távoli tároló (GitHub) tartalmaz olyan változtatásokat, amelyek nincsenek jelen a helyi másolatodban. Ennek az oka lehet, hogy valaki más már tett fel commitokat a távoli tárolóra, vagy te korábban nem szinkronizáltad a távoli tárolót a helyi repóval.

A probléma megoldásához kövesd az alábbi lépéseket:

1. **Töltsd le a távoli változtatásokat és egyesítsd őket:**

Először is, futtass egy git pull parancsot, hogy letöltsd a távoli változtatásokat és egyesítsd őket a helyi repóddal.

bash

Copy code

git pull origin main

Ez a parancs letölti a main ágon lévő távoli változtatásokat, és próbálja meg egyesíteni őket a helyi repóddal.

1. **Oldd meg az esetleges konfliktusokat:**

Ha a git pull során ütközések (conflict) lépnek fel, git értesítést ad róluk. A konfliktusokat manuálisan kell megoldanod. A fájlokban található konfliktusokat a <<<<<<<, ======= és >>>>>>> jelölések segítenek azonosítani. Miután megoldottad a konfliktusokat, ne felejts el commitálni:

bash

Copy code

git add <file>

git commit -m "Resolved merge conflicts"

1. **Pushold vissza a változtatásokat:**

Miután sikeresen letöltötted és egyesítetted a változtatásokat, próbálj meg újra push-olni:

bash

Copy code

git push -u origin main

1. **Ellenőrizd a távoli repót:**

Győződj meg róla, hogy a távoli repóban (GitHub-on) láthatóak a változtatások, és hogy minden megfelelően van szinkronizálva.

**Ha továbbra is problémáid vannak:**

* **Nézd meg a git status parancs kimenetét:** Ez segíthet megérteni a repó állapotát és a fennálló problémákat.
* **Használj git fetch és git merge helyett git pull parancsot:** A git fetch letölti a változásokat, míg a git merge a helyi repóddal való egyesítést végzi el.

bash

Copy code

git fetch origin

git merge origin/main

Ne habozz visszajelzni, ha további segítségre van szükséged!

4o mini

agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $ git pull origin main remote: Enumerating objects: 10, done. remote: Counting objects: 100% (10/10), done. remote: Compressing objects: 100% (10/10), done. remote: Total 10 (delta 0), reused 0 (delta 0), pack-reused 0 Unpacking objects: 100% (10/10), 26.62 MiB | 8.66 MiB/s, done. From github.com:ahabram/adoszpoly \* branch main -> FETCH\_HEAD \* [new branch] main -> origin/main fatal: refusing to merge unrelated histories agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $

ChatGPT

A fatal: refusing to merge unrelated histories hiba azt jelzi, hogy a Git nem tudja egyesíteni a két történetet, mert azok függetlenek egymástól. Ez általában akkor fordul elő, amikor két különböző repót próbálsz összeolvasztani, amelyeket korábban nem kapcsolt össze a közös történet.

Ennek a problémának a megoldásához használhatod az --allow-unrelated-histories opciót, amely lehetővé teszi a különálló történetek egyesítését. Íme, hogyan tudod ezt alkalmazni:

1. **Használj git pull parancsot az --allow-unrelated-histories opcióval:**

bash

Copy code

git pull origin main --allow-unrelated-histories

Ez a parancs letölti a távoli változásokat és egyesíti azokat a helyi repóddal, még ha azok története teljesen különböző is.

1. **Ellenőrizd a változtatásokat:**

Miután a pull befejeződött, nézd meg a változásokat és ellenőrizd, hogy minden rendben van-e. Használhatod a git status és git log parancsokat:

bash

Copy code

git status

git log

1. **Oldd meg az esetleges konfliktusokat:**

Ha a merge során konfliktusok lépnek fel, a Git jelzi őket. Oldd meg a konfliktusokat a fájlokban található jelölések alapján (<<<<<<<, =======, >>>>>>>). Miután megoldottad a konfliktusokat, commitálj:

bash

Copy code

git add <file>

git commit -m "Resolved merge conflicts"

1. **Pushold vissza a változtatásokat:**

Miután sikeresen letöltötted és egyesítetted a változtatásokat, próbálj meg újra push-olni:

bash

Copy code

git push -u origin main

Ezek a lépések segítenek abban, hogy a helyi és a távoli repó történeteit egyesíteni tudd, még ha azok korábban nem voltak összehangban. Ha bármilyen további problémád van, szólj bátran!

4o mini

agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main) $ git pull origin main --allow-unrelated-histories From github.com:ahabram/adoszpoly \* branch main -> FETCH\_HEAD Auto-merging index.html CONFLICT (add/add): Merge conflict in index.html Auto-merging styles.css CONFLICT (add/add): Merge conflict in styles.css Automatic merge failed; fix conflicts and then commit the result. agnes@Andras-Laptop MINGW64 /d/\_mentes\_d/adatok/andras/web/weboldal3 (main|MERGING) $

ChatGPT

A CONFLICT jelzés azt mutatja, hogy a Git nem tudta automatikusan összeolvasztani a változtatásokat a index.html és styles.css fájlokban. Most kézzel kell megoldanod ezeket a konfliktusokat.

Íme a lépések a konfliktusok megoldásához:

1. **Nyisd meg a konfliktusokat tartalmazó fájlokat:**

Használj egy szövegszerkesztőt (például nano, vim, code, stb.) az alábbi fájlok megnyitásához:

bash

Copy code

nano index.html

nano styles.css