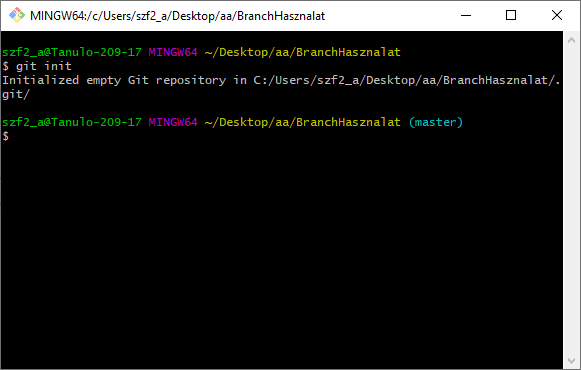
Lokális repository létrehozása:



A gitignore.io weblapon keresztül generálunk egy gitignore fájlt. A *Netbeans* és *Java* kulcsszavakat kell begépelni, majd a *Create* gombra kattintani.

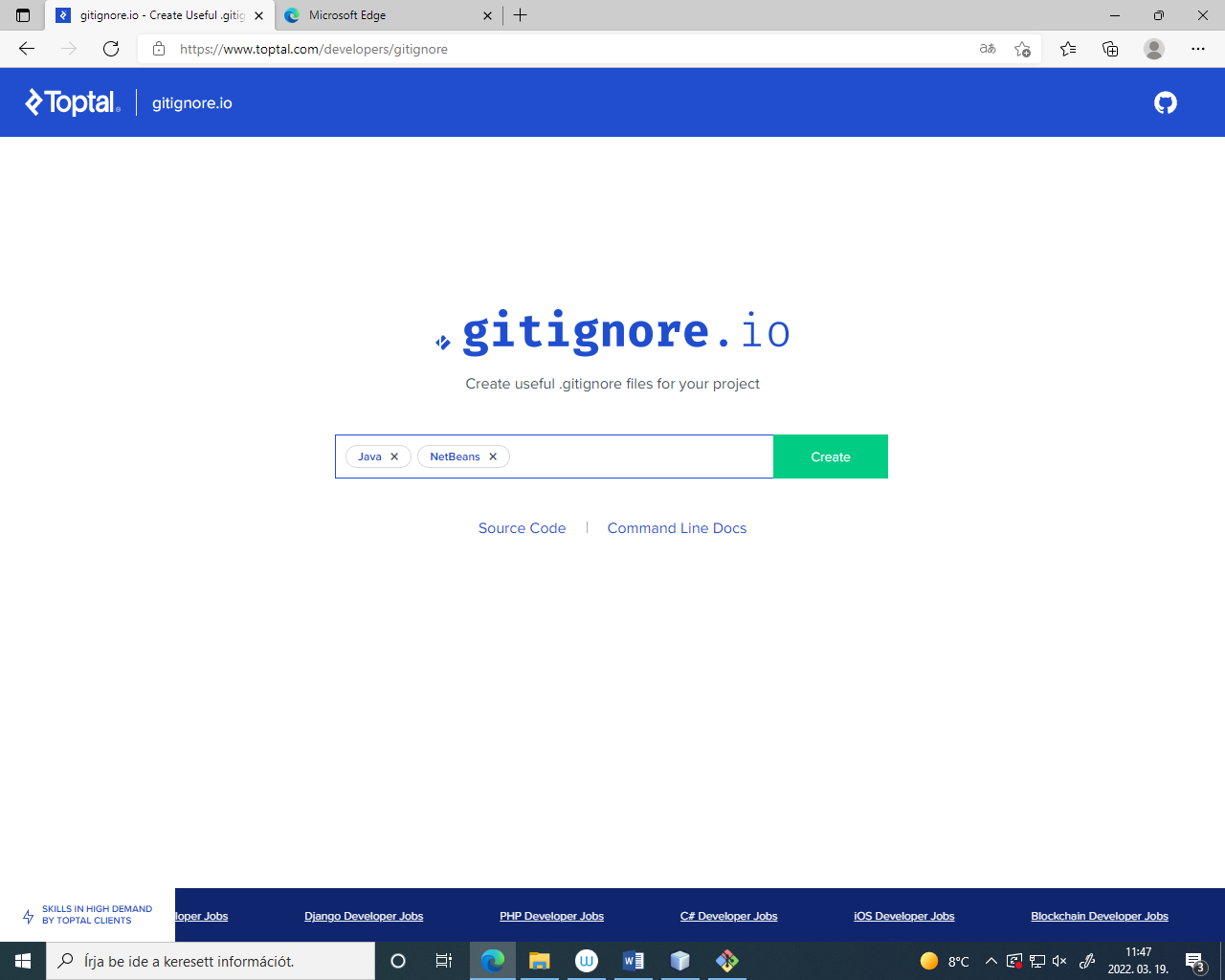
Bővebben a gitignore-ról:

*When a Git project is compiled it may generate binary files, lock files, temporary files and metadata files. These are intermediate files that are not part of the source code. For example, IDE config files or intermediate files like “.class”,”.exe”, “.bin” etc. that are generated during compilation should be ignored by the version control system as they will be different for each developer machine and will unnecessarily increase the size of the repository.*

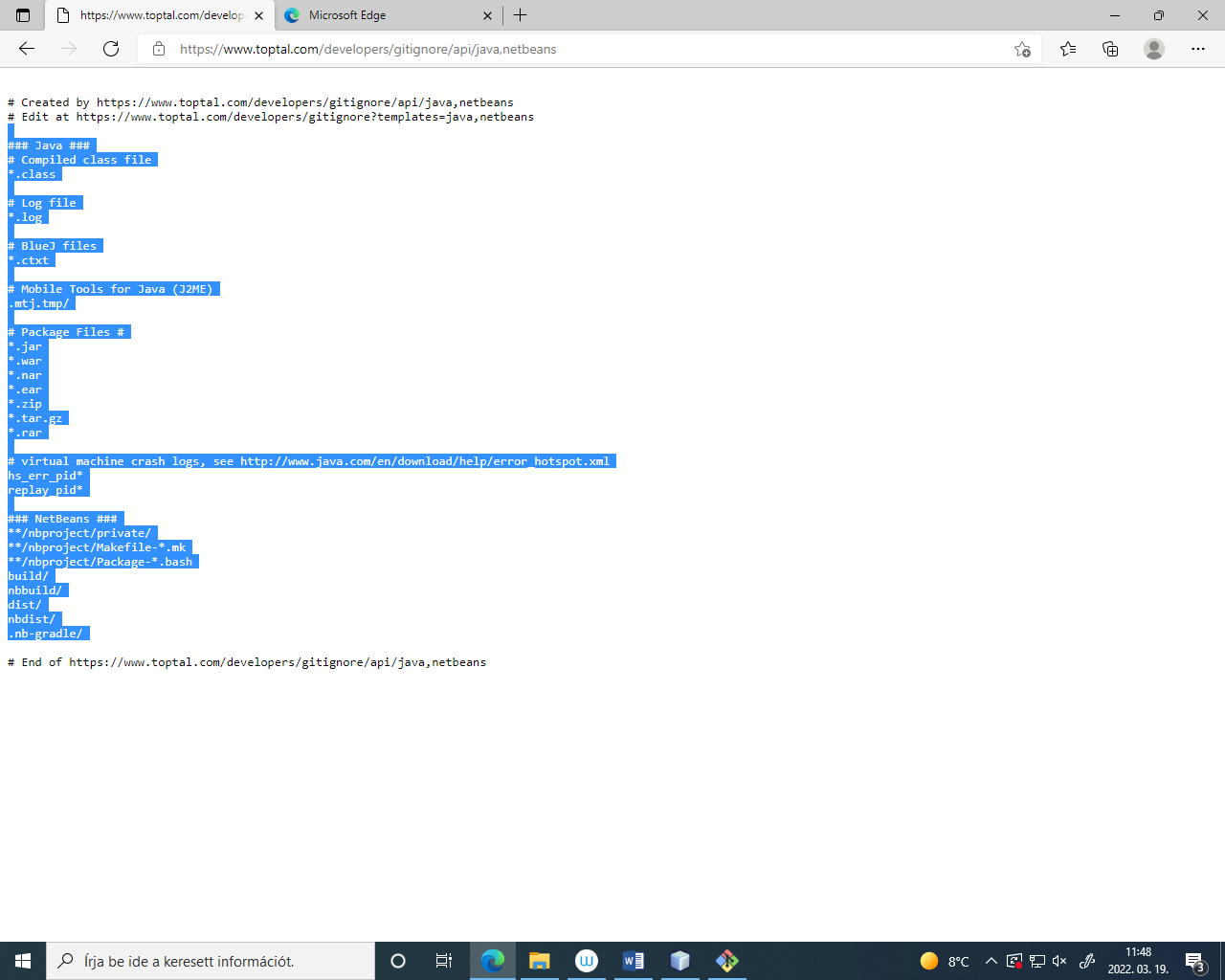
*The .gitignore is a text file where each line contains a pattern for files or directories to ignore. It is usually placed at the root of the project folder. Alternatively, you can put it in any folder in the repository and a project can have multiple .gitignore files.*

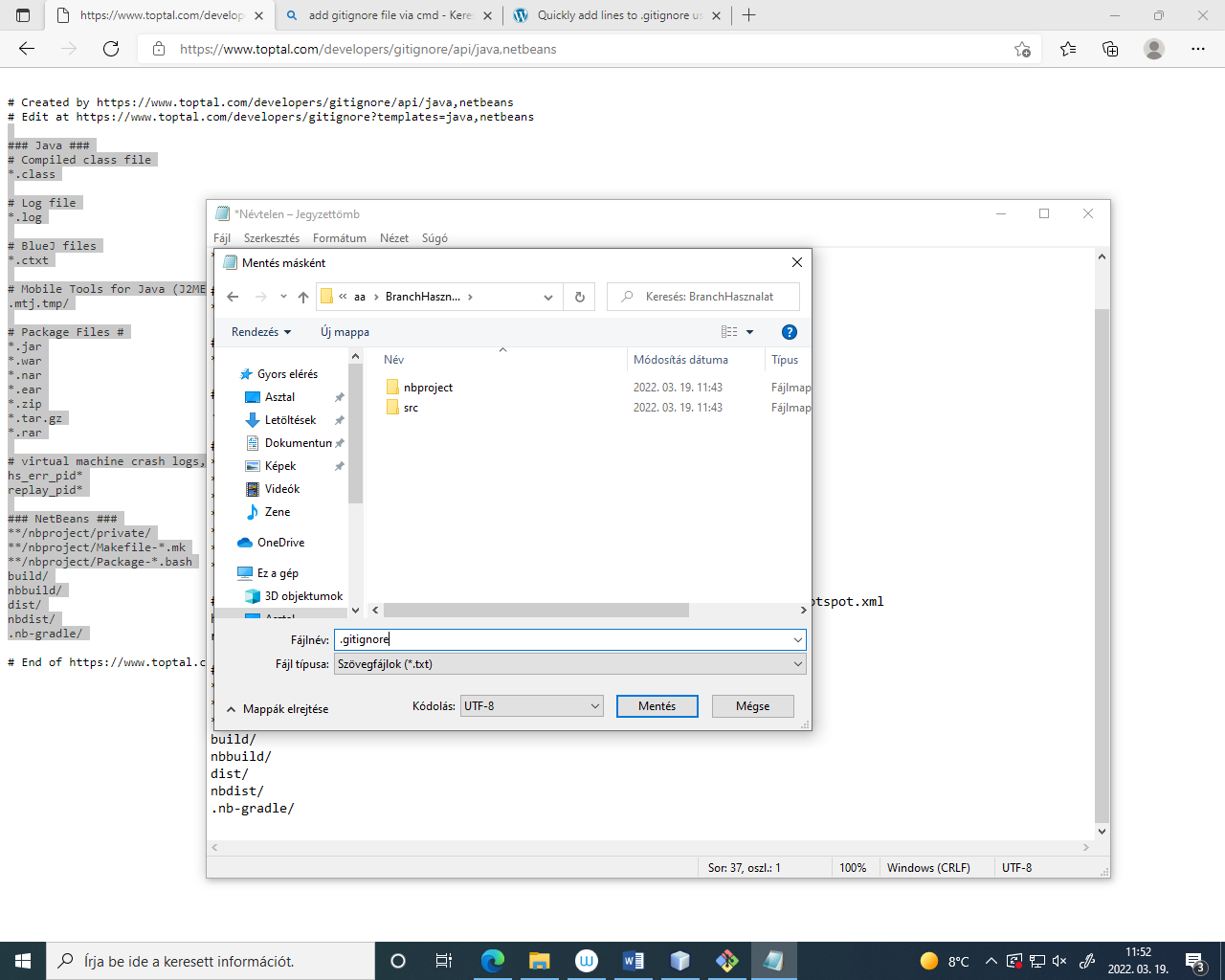
*This file lists files that are intentionally untracked and should be ignored by Git. It is important to note that changes to files that were staged before being added to the .gitignore file will continue to be tracked by Git. In other words, .gitignore will only ignore files in the working directory.*

Forrás: <https://www.tutorialspoint.com/what-is-the-purpose-of-the-gitignore-file>



Egy új tabon megjelenik a gitignore kód. Azt bemásolom egy txt fájlba, és .gitignore néven elmentem.



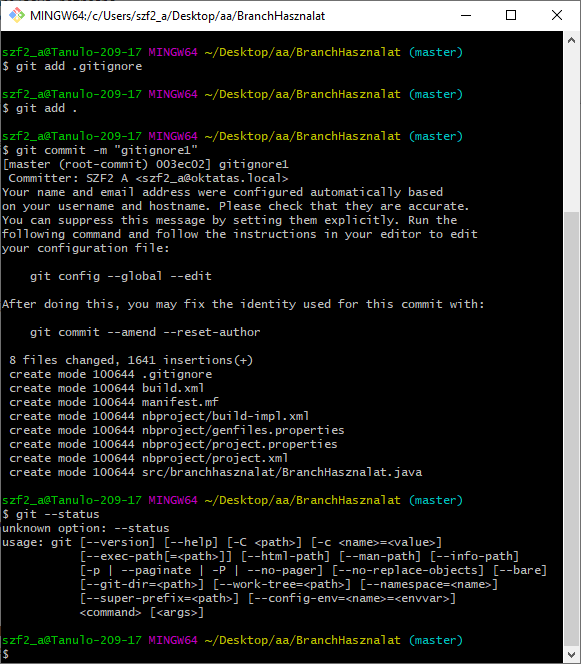


A kezdeti változat committolása:

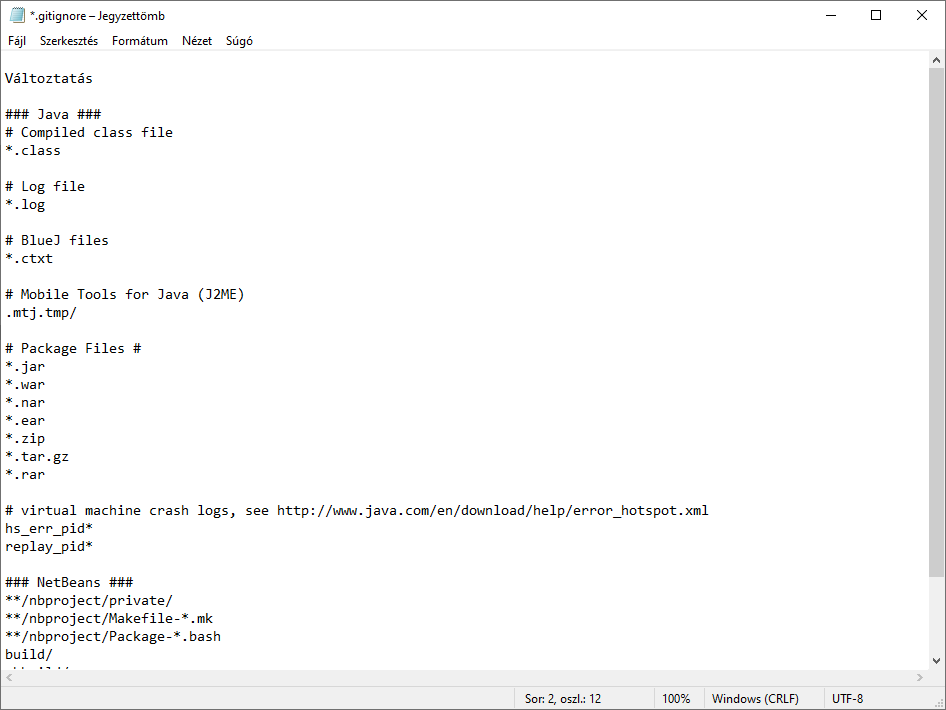
$ git add .gitignore

$ git add .

$ git commit −m ‘add gitignore1’



Megváltoztatom a fájl tartalmát:



Lekérem a státuszt 🡪 pl. kiírja, ha módosítást érzékel valamely állományban a korábbi commit állapotához képest:

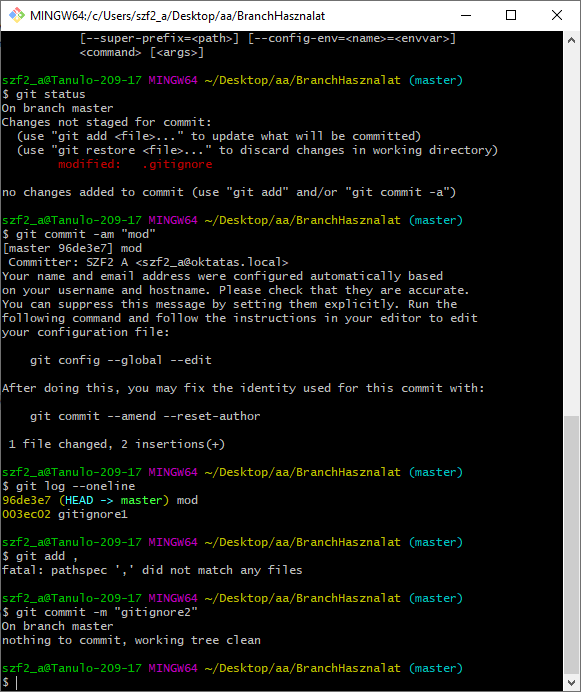
$ git status

Eddigi commitok kilistázása:

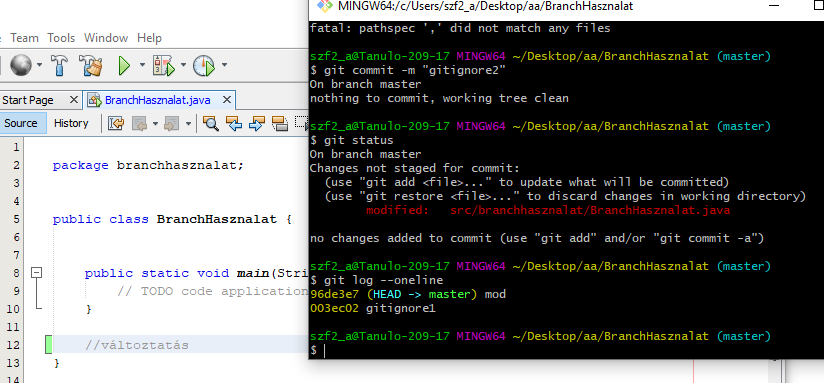
$ git log --oneline

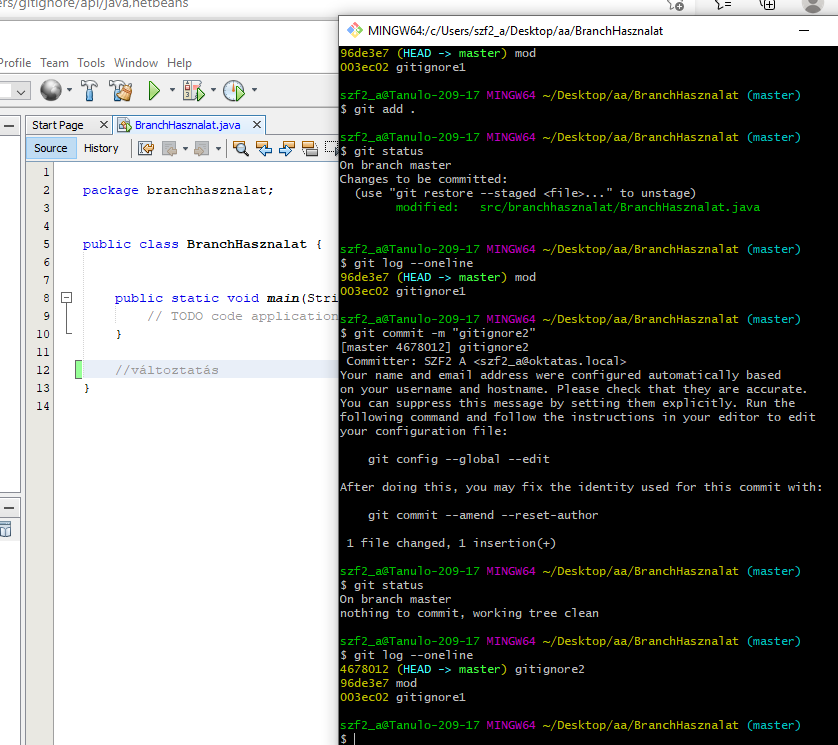
A gitignore fájlban történt módosítást jelzi a *git status* parancs futtatásakor megjelenő válasz.

Felcommitoljuk a gitignore-ban eszközölt változtatást, és feltöltjük a Java projektben létrehozott fájlokat is. Látható, hogy 1 fájl módosítva lett, +2 pedig felkerült a repoba.



Most a Java fájl tartalmát változtatjuk meg:





amend-es parancs: utolsó komment javítása

git commit –am „modositott komment”

git checkout 11111– visszalépés korábbi állapotra

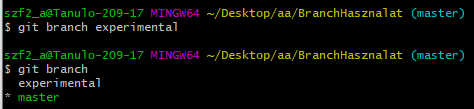
git reflog

git log --oneline 🡪 commit history

git switch –

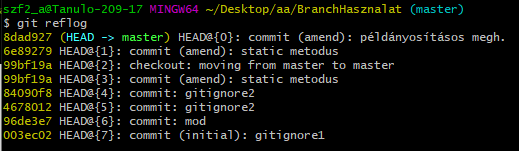
git branch 🡪 hány ág van

git branch new\_branch 🡪 új br. lérehozása



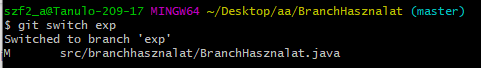
git branch –D experimental 🡪 exp. ág kitörlése





git merge exp

Másik ágra kapcs.:



Vissza a Masterre:

