

Tartalom

Mi is a TypeScript?.....	2
Legfontosabb funkciói	2
TypeScript beüzemelése.....	2
Node.JS csomagkezelőjének letöltése és telepítése	2
1. telepítőablak - A szoftver üdvözlő képernyője	3
2. telepítőablak – Licenz szerződés	3
3. telepítőablak – Telepítés helye.....	4
4. telepítőablak – Összetevők	4
5. telepítőablak – Kiegészítő modulok	5
6. telepítőablak – Telepítésre kész	5
7. telepítőablak – Telepítés állapot	6
8. telepítőablak – Sikeres telepítés visszajelzése.....	6
9. Telepítés ellenőrzése – Terminál ablak indítása	7
10. Telepítés ellenőrzése – node -v parancs futtatása	7
TypeScript letöltése	7
Telepítő parancs	7
Verzió és telepítés sikerességének ellenőrzése	7
TypeScript használata.....	8
Alapvető filestruktúra.....	8
Terminal.....	8
TypeScript file fordítása.....	9
jQuery használata TypeScript segítségével	9
További kiegészítők információkat az alábbi oldalakon találhatsz még:	10
https://www.typescriptlang.org/	10
https://www.w3schools.com/typescript/index.php	10
https://www.typescriptlang.org/play/	10

Mi is a TypeScript?

- Microsoft által fejlesztett úgynevezett JavaScript szuperszett, a JavaScript fejlesztett verziója
- Transpiler – Nem futtat, hanem fordít, minden benne megírt kódot JavaScripté fordítja le
- Mivel JS kódot generál ezért könnyen integrálható más JS projektekbe
- Összetett projektek esetén használják, mivel már fordításkor figyelmeztet az előforduló hibákra (erről később)

Legfontosabb funkciói

- Statikus típusellenőrzés (fordítási időben, kiszűrhető)
- Modulokra való bontási lehetőség
- Osztályok használata
- Package Managerek használata (JavaScript csomagok egyszerű használata)

TypeScript beüzemelése

Node.JS csomagkezelőjének letöltése és telepítése


Elérhető az alábbi oldalon: <https://nodejs.org/en/>

Itt elérhető az **LTS (Long Time Support verzió)** vagy a **Current**, ami az aktuálisan legújabb feature-ket is tartalmazza.

Mivel mi az alapokat sajátítjuk el elég az LTS verzió, ha gyakorlott leszel, a későbbiekben érdemes mindig az aktuálisan legfrissebb verziót feltölteni. De ügyel arra, hogy az aktuális verziók gyakrabban tartalmazhatnak hibákat/biztonsági réseket

Run JavaScript Everywhere

Node.js® is a free, open-source, cross-platform JavaScript runtime environment that lets developers create servers, web apps, command line tools and scripts.

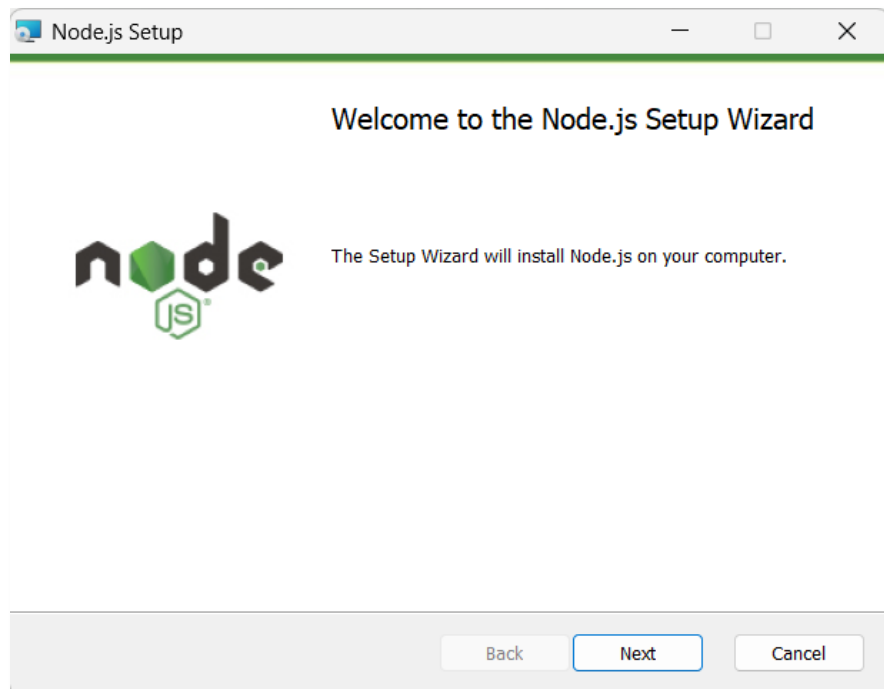
[Download Node.js \(LTS\)](#) 

Downloads Node.js **v22.16.0**¹ with long-term support.
Node.js can also be installed via [version managers](#).

Want new features sooner? Get **Node.js v24.1.0** ¹ instead.

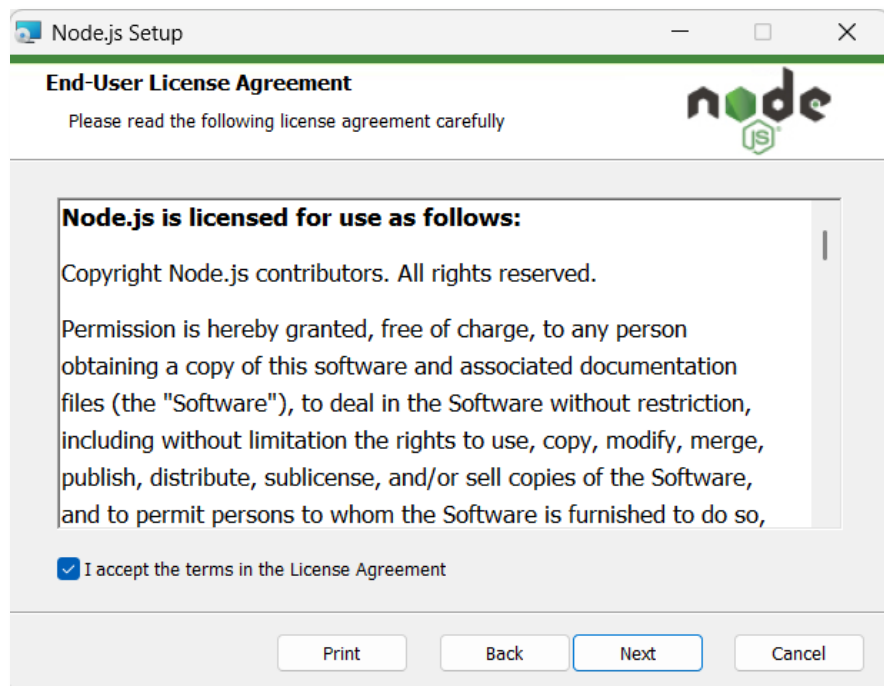
1. telepítőablak - A szoftver üdvözlő képernyője

Üdvözlő minket, és leírja, hogy ez a node.js szoftver telepítője, kattintsunk a **next** gombra!



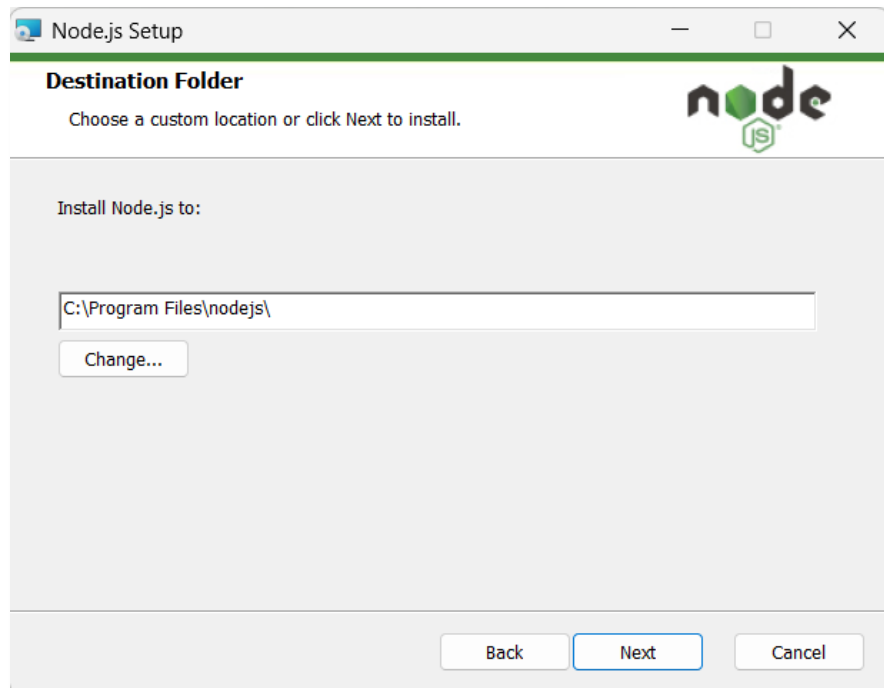
2. telepítőablak – Licenz szerződés

Licenszekkel kapcsolatos információk elfogadása, aki szeretné elfogadhatja, de mivel ígyis-úgyis szükségünk van rá, csak pipáljuk ki a checkbox-ot aluk, és kattinthatunk is a next gombra ismét.



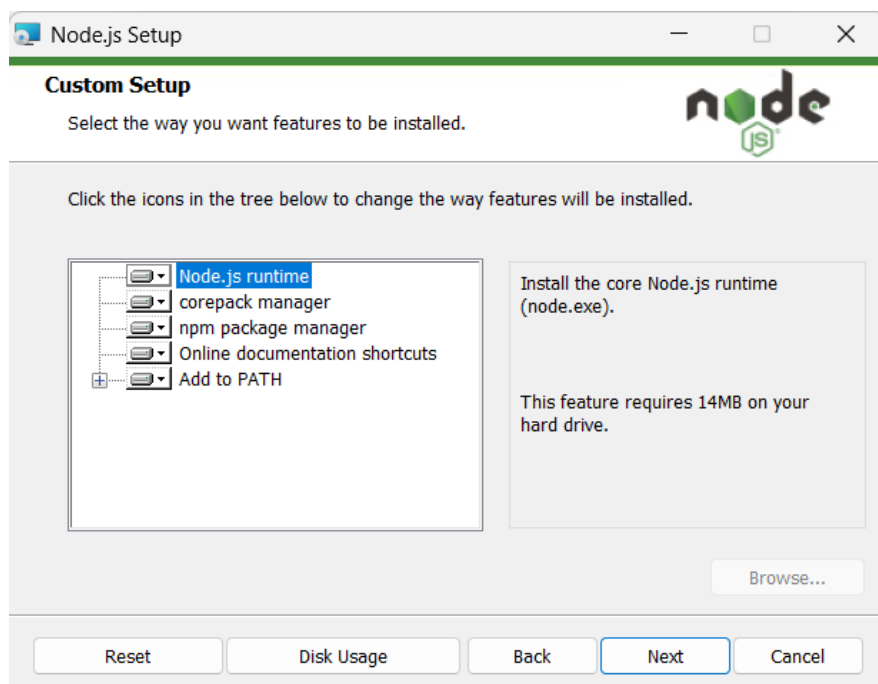
3. telepítőablak – Telepítés helye

Itt megadhatjuk mely helyre szeretnénk telepíteni a szoftver összetevőit, miután kiválasztottuk, mehet is a kattintás a next gombra, de természetesen maradhat az alapértelmezett útvonal is.



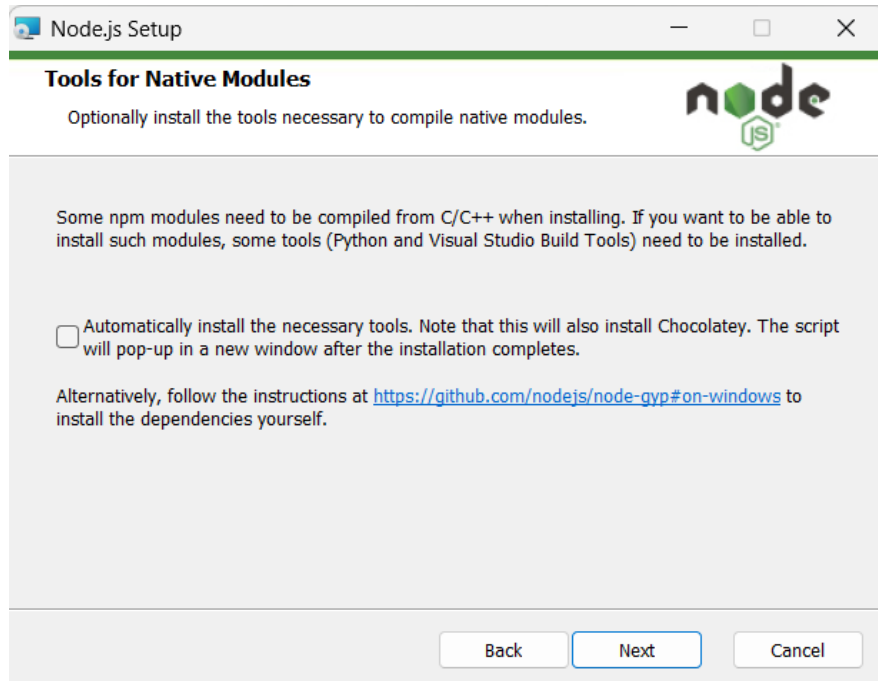
4. telepítőablak – Összetevők

Itt az összes összetevő be van kapcsolva alapértelmezetten, melyből az összes telepítése ajánlott, így ezt ezúttal nem igazán érdemes bolygatni. (Jelenleg a legfontosabb az **npm package manager** telepítése, a **TypeScripthez**)



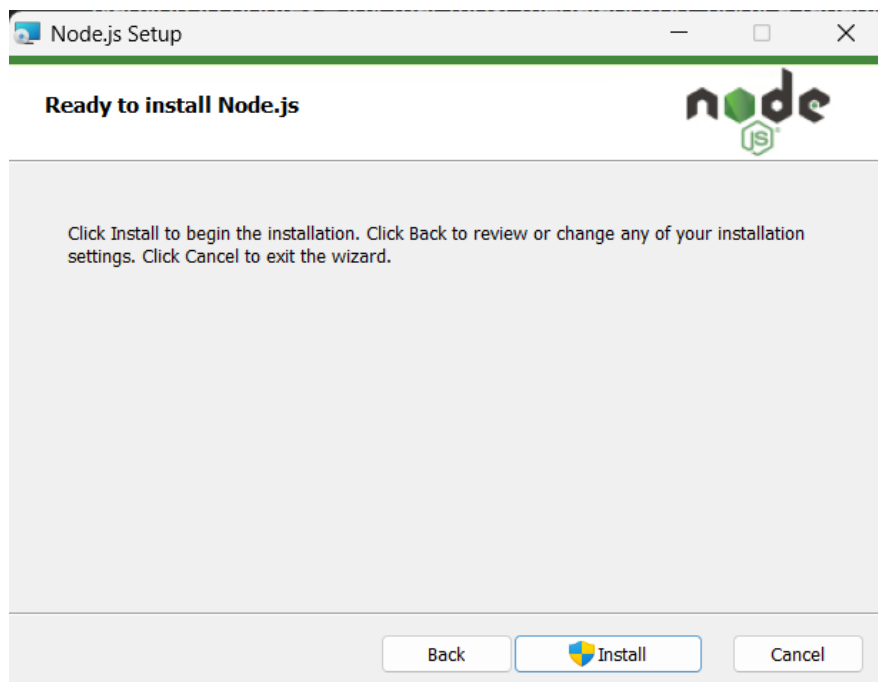
5. telepítőablak – Kiegészítő modulok

Ez az opció natív C/C++ modulokhoz szükséges fejlesztői eszközöket telepít. **Ha csak TypeScriptet, webes JavaScriptet vagy front-end fejlesztést tanulsz, hagyd üresen a jelölőnégyzetet!** Csak akkor pipáld be, ha később például adatbázis illesztőmodulokat vagy gépi tanulási könyvtárakat akarsz fordítani Node.js-ben.



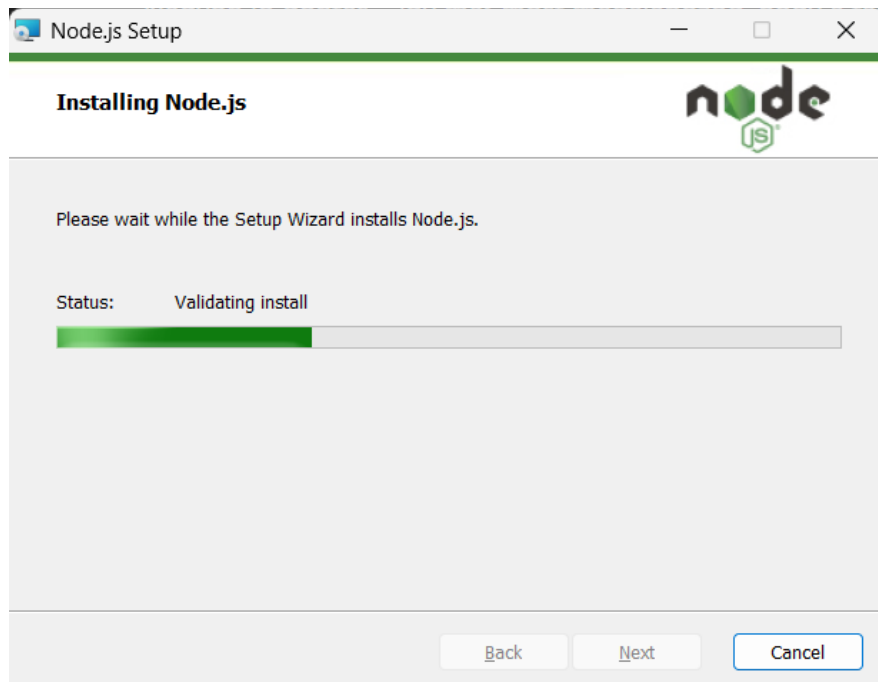
6. telepítőablak – Telepítésre kész

A következő ablak, csak informál, készen állunk a telepítésre, most még visszaléphetünk, ha valamit módosítani akarunk. **.Install** gombra kattintva elindul a telepítés!



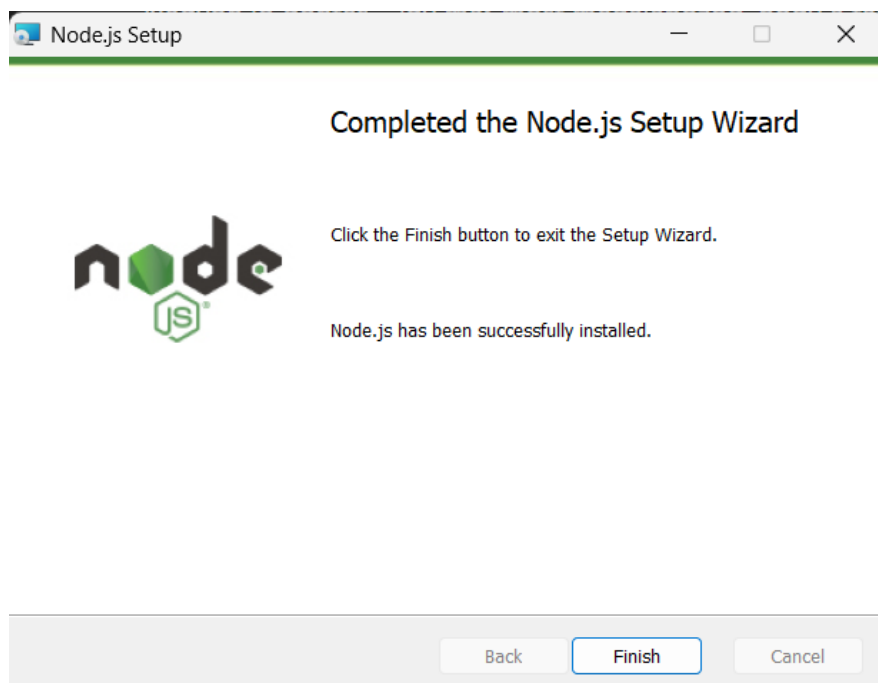
7. telepítőablak – Telepítés állapot

A következő ablakon csak simán várunk kell, az aktuális telepítés állapotát láthatjuk.



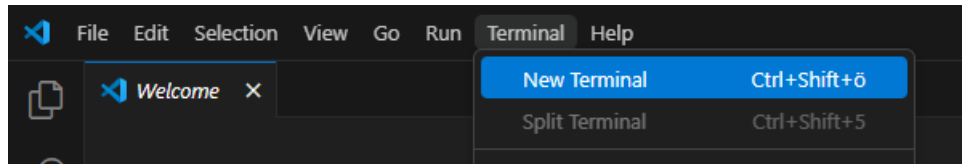
8. telepítőablak – Sikeres telepítés visszajelzése

Amennyiben a telepítésünk sikeres volt, az alábbi ablakot fogod látni! Kattints a **Finish**-re.



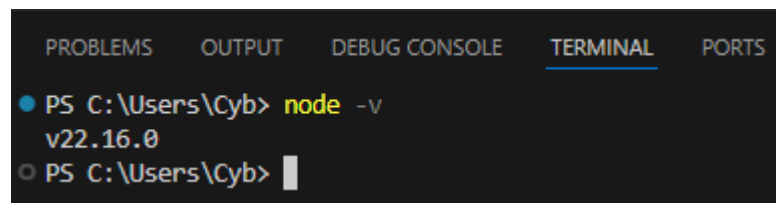
9. Telepítés ellenőrzése – Terminál ablak indítása

Indíts el egy új terminál ablakot a VSCode-on belül, a képen látható módon.



10. Telepítés ellenőrzése – node -v parancs futtatása

Telepítés sikerességének vissza ellenőrzésére használjuk terminálban a **node -v** parancsot, ami megjeleníti a számítógépen lévő aktuális node verzióját. A terminál alapértelmezetten a VSCode ablak alsó részében jelenik meg oda kell beírnunk a szöveget.



TypeScript letöltése

Miután ellenőriztük, hogy fent van a node verzió, megnézhetjük az aktuális **npm package manager** verziót is az **npm -v** paranccsal, szintén parancssorban.

```
PS C:\Users\Cyb> npm -v  
10.9.2
```

Telepítő parancs

A typescript telepítéséhez az: **npm install -g typescript**

```
PS C:\Users\Cyb> npm install -g typescript  
added 1 package in 1s
```

```
PS C:\Users\Cyb> npm install -g typescript  
changed 1 package in 3s
```

Aki most először telepíti annak az **added package** szöveg kell megjelenjen, akinek már volt fent korábban verzió (mint nekem is a képen) a **changed package** szöveg fog megjelenni.

Verzió és telepítés sikerességének ellenőrzése

A typescript verzió visszaellenőrzéséhez, használhatjuk a: **tsc -v** parancsot.

```
PS C:\Users\Cyb> tsc -v  
Version 5.8.3
```

TypeScript használata

A TypeScriptek használatához **.ts** kiterjesztésű fájlt kell létrehoznunk, ebbe kerül majd a TypeScript kód. Viszont ezeket le kell fordítani valahova, ehhez kell egy JavaScript file is, ide történik a fordítás.

Alapvető filestruktúra

Használatához létre kell hozni az alábbi filestruktúrát:

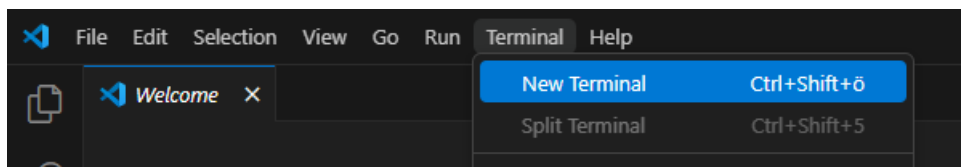
index.html – Alap HTML oldal, ide meg kell hívni a **mintakod.js** állomány tartalmát **script tag**-ek közé

mintakod.ts – Ez fogja tartalmazni a TypeScript kódot

mintakod.js – Ide kerül a lefordított TS kód a fordítása parancs kiadása után! (Automatikusan is legenerálódik, fordításkor)

Terminal

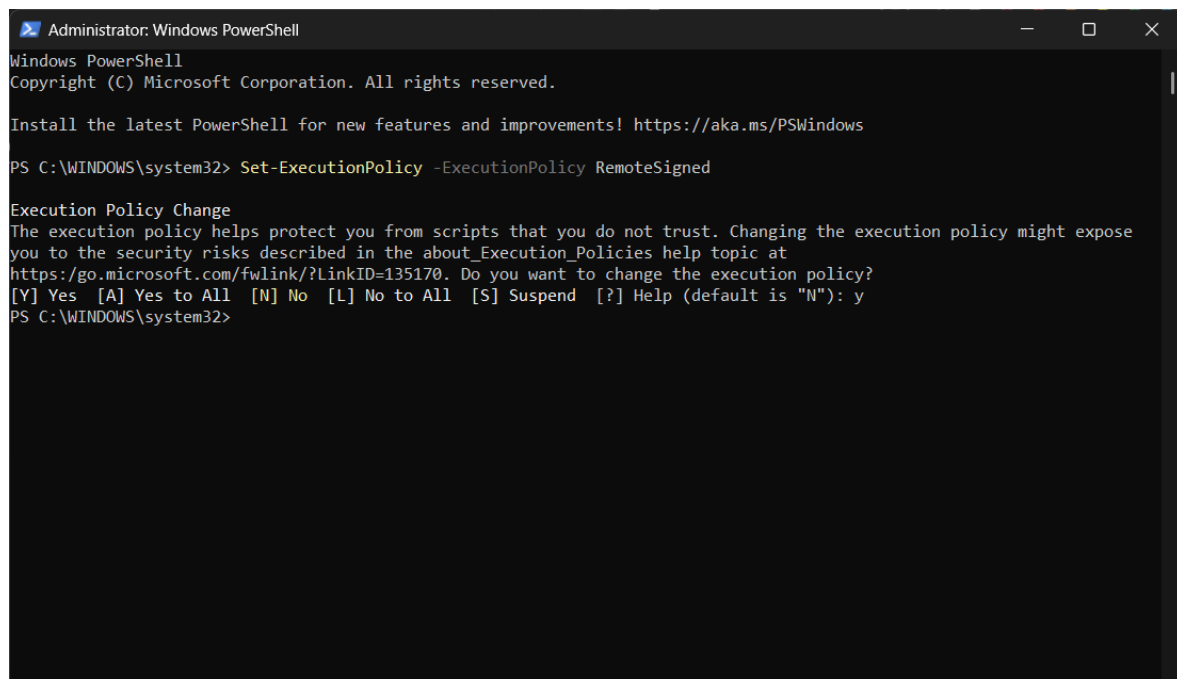
A kódok futtatásához szükség lesz a Visual Studio Coda-ban beépített terminálra melyet a **terminal fülön** (utolsó előtt) a **new terminal** menüponttal érhetünk el, mint korábban a jegyzetben olvasható volt.



Simán nem futtathatjuk a TypeScriptet, előtte a Windows PowerShell ablakban, amit **adminisztrátorként (Futtatás rendszergazdaként)** indítunk, ki kell adnunk a következő parancsot, mely amiatt van

Set-ExecutionPolicy -ExecutionPolicy RemoteSigned

A parancs kiadása után az Y-t kell megnyomni az engedélyezéshez



TypeScript file fordítása

Miután mindent telepítettünk és aktiváltunk, a terminálban el kell menni az aktuális mappánkba, ehhez a `cd` és `cd..` parancsot érdemes használni.

cd mappaElérésiÚt az adott mappába lép be.

cd.. paranccsal visszalépünk az adott mappából

Miután ott vagyunk csak kiadjuk az alábbi parancsot:

```
PS W:\tsminta_projekt> tsc mintakod.ts
```

, és a kódot lefordítja egy JS fájlba! Amit, ha az nem létezett volna, létre is hozza magától!

Amennyiben szeretnéd, hogy ne kelljen folyton kiadni a parancsot add ki azt az alábbi módon:

```
PS W:\tsminta_projekt> tsc mintakod.ts -w
```

A **-w** a watch szóból ered, így figyelni folyamatosan a változásokat és **automatikusan fordítja a typescript kódot**.

```
[12:32:01] Starting compilation in watch mode...  
[12:32:02] Found 0 errors. Watching for file changes.
```

Ebből a „figyelési funkcióból” kilépni a **ctrl+c** billentyűkombinációval tudsz

JQuery használata TypeScript segítségével

Lehetőségünk van a JQuery modult importálni a kódunkhoz.

Ehhez 2 dolgot kell tennünk, elsőként importálni a projekthez a JQuery könyvtárat az alábbi paranccsal:

```
PS W:\tsminta_projekt> npm i @types/jquery --save-dev  
added 2 packages in 2s
```

Ennek segítségével a JQuery **típusosan** használható lesz a TS segítségével. A `@types/jquery` csak a **TypeScript számára adja meg, hogyan kell értelmezni a \$ változót** és a JQuery függvényeket – **maga a jQuery működéséhez a JavaScript könyvtárat is külön be kell öltetnünk**, például CDN-nel.

```
▼ TSMINTA_PROJEKT  
  > node_modules  
    <> index.html  
    JS mintakod.js  
    TS mintakod.ts  
    {} package-lock.json  
    {} package.json
```

Majd hivatkozni rá az index fájl head részében, lehetőleg a mi fájlunk előtt, hogy nehogya felülírja az általunk megírt kódokat!

```
<script  
src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>
```

További kiegészítő információkat az alábbi oldalakon találhatsz még:

<https://www.typescriptlang.org/>

Ez a TypeScript hivatalos weboldala, ahol megtalálható a teljes dokumentáció, nyelvi referencia, bevezetők és eszközök. Ideális kiindulópont a TypeScript tanuláshoz, hiszen folyamatosan frissül, és tartalmazza az összes funkció részletes magyarázatát, példakódokkal és verzióinformációkkal. A „Docs”, „Handbook” és „What's New” részek különösen hasznosak.

<https://www.w3schools.com/typescript/index.php>

A W3Schools TypeScript-oldala egy egyszerű, vizuálisan áttekinthető bevezetőt nyújt a nyelv alapjaiba. Interaktív példákkal, könnyen érthető nyelvezettel mutatja be a típusokat, változókat, függvényeket, osztályokat és más alapvető fogalmakat. Ideális teljesen kezdőknek, akik most ismerkednek a típusos programozással.

<https://www.typescriptlang.org/play/>

Ez a TypeScript online „Playground” felülete, ahol a felhasználó azonnal kipróbálhatja a TypeScript kódokat böngészőből, telepítés nélkül. A kódírás közben valós időben látható a lefordított JavaScript-kód is, így könnyen megérthető, mit csinál a fordító. Oktatásban, demózáshoz vagy gyors teszteléshez ideális.