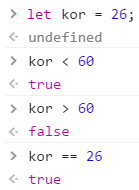
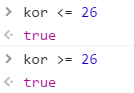
# Elágazások

## Bevezetés

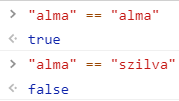
Ebben a leckében megtanuljuk, hogy készíthetünk elágazásokat a JavaScriptben. Ezután megnézzük, hogyan tudjuk elenőrizni a scriptekben beolvasott adatokat.

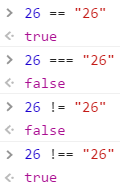
## A boolean típus

Egy boolean típusú változó kétféle logikai értéket tartalmazhat: true (igaz), vagy false (hamis). Például let hazas = false;

A logikai értéket leggyakrabban két érték összehasonlításával állítjuk elő. Próbáld ki a következőket a böngésző konzolján:  
  
  
  
Mint látható, a JavaScriptben is a szokásos műveleti jelekkel lehet összehasonlítani az értékeket.

A JavaScriptben (a Java-val ellentétben) a szövegeket is össze lehet hasonlítani a relációs jelekkel.



Van még egy különlegesség is. Próbáld ki a következőket:  


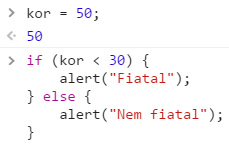
Két egyenlőségjel használata esetén a JavaScript a számot automatikusan stringgé alakítja az összehasonlítás előtt, ezért lesz az első sor igaz.

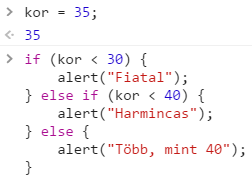
Három egyenlőségjelnél nem végez automatikus típusátalakítást, ezért nem egyenlő a két érték. Ilyenkor a típusnak is meg kellene egyeznie.

Ugyanezek a nem egyenlő vizsgálatnál is igazak.

## Az if utasítás

Az if utasítással feltételhez köthetjük más utasítások végrehajtását. Próbáld ki a következőt:  
  
Mivel az if utáni feltétel igaz, végrehajtotta az alert utasítást.  
Mi történt volna, ha nem igaz a zárójelben lévő feltétel? Próbáld ki!

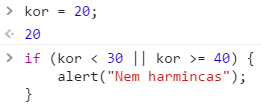
Az if utasításhoz meg lehet adni else ágat is, amelyet akkor hajt végre a program, ha nem igaz a feltétel. Próbáld ki:  
  


Többirányú elágazást is lehet készíteni else if ágak segítségével:  


## Logikai műveletek

Több feltételt össze lehet kapcsolni a logikai műveletek segítségével.

Az ÉS művelet ( && ) eredménye akkor igaz, mindkét feltétel igaz. Például:  


A VAGY művelet ( || ) eredménye akkor igaz, ha legalább az egyik feltétel igaz. Például:  


A JavaScript a logikai műveleteket az összehasonlítások után végzi el. Több logikai művelet esetén először az ÉS műveleteket hajtja végre, utána a VAGY műveleteket.

Ha az egyik feltétel alapján az eredmény már biztosan igaz vagy hamis, akkor a többi összehasonlítást már nem végzi el, mert feleslegesek.

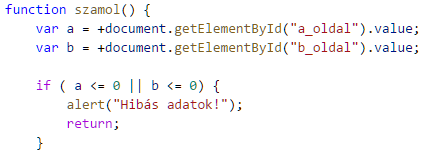
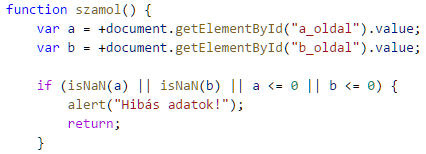
Ha például az előző if utasításban a kor kisebb, mint 30, akkor a 40-nel már nem hasonlítja össze, mert az eredmény már biztosan true.

## Előkészületek

1. Nyisd meg a *tegla* mappát, majd az abban lévő fájlokat a Visual Studio Code-ban!
2. Nyisd meg az oldalt a böngészőben is!
3. Nézd át az eddig elkészített kódokat!

## Adatbevitel ellenőrzése

Az űrlap beviteli mezőiből vagy a prompt() függvénnyel beolvasott értékeket célszerű felhasználás előtt ellenőrizni. Például egy negatív oldalhossznak vagy egy szám helyett beírt szövegnek nincs értelme.

1. Egészítsd ki a szamol() függvényt úgy, hogy csak pozitív számokat fogadjon el az a és a b oldal értékeire:  
     
   Ha semmit nem írsz be, azt is nullának veszi a számmá való átalakítás miatt. Próbáld ki!
2. Ez a vizsgálat még nem kezeli azt az esetet, amikor érvénytelen számot írsz be. (Például betű van benne.) Ilyenkor azt kell vizsgálni, hogy az érték egyenlő-e az NaN értékkel (NaN = Not a Number = nem szám). Ezt viszont nem lehet összehasonlítással, hanem az isNaN() függvényt kell használni. Egészítsd ki a kódot így:  
     
   Próbáld ki érvénytelen számokkal!  
   Figyeld meg, hogy először azt vizsgáltuk, hogy szám-e, és csak utána hasonlítottuk össze a 0-val! Ha nem szám, akkor az összehasonlítást már próbálja elvégezni.

## Önálló feladat

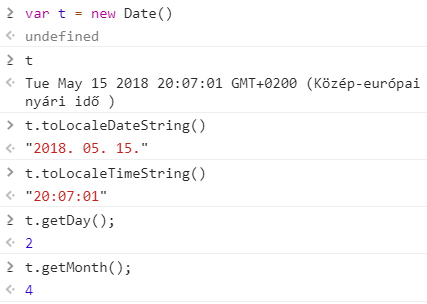
Nyisd meg a *henger* mappát, amely az egyik korábbi feladat megoldását tartalmazza! Ellenőrizd a beolvasott adatokat úgy, mint a téglalapnál!

## Dátum és idő lekérdezése

A JavaScriptben van egy Date nevű osztály, amelyből a new utasítással tudunk létrehozni egy új példányt:  


Próbáld ki a konzolon!

A t objektum a létrehozáskor aktuális dátumot és időt tartalmazza:  


Ebből különböző metódusokkal le tudjuk kérdezni a dátum és az idő egyes részeit:  


A napoknál a 0-s elem a vasárnap, a hónapoknál a január.

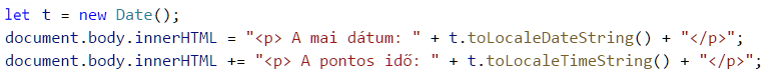
Próbáld ki a konzolon!

Készíts egy új mappát *naptar* néven, és nyisd meg a Visual Studio Code-ban!

Hozz létre egy új HTML fájlt *naptar.html* néven, az alap HTML tartalommal!

Készíts egy program.js fájlt, és kapcsold a HTML fájlhoz!

Írd a program.js fájlba a következőket, majd nézd meg az eredményt a böngészőben:



Figyeld meg, hogyan hivatkoztunk a body elemre!

## A switch utasítás

A JavaScriptben is használható a switch utasítás, ugyanabban a formában, mint a Java-ban.

Egészítsd ki a programot úgy, hogy a nap nevét is írja ki:



Próbáld ki! Figyeld meg, hogy a vasárnap a 0-dik nap a héten!

## Önálló feladat

Írasd ki az évszakot is az oldalra!