# Számítások

## Bevezetés

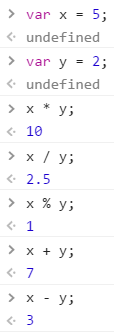
Ebben a leckében megtanuljuk legfontosabb matematikai műveleteket és függvényeket. Utána alkalmazzuk ezeket egy henger felszínét és térfogatát kiszámító programban.

## Műveletek számokkal

A számokkal végezhető műveleteket az alábbi táblázat mutatja:

|  |  |
| --- | --- |
| Műveleti jel | Művelet |
| \* | szorzás |
| / | osztás |
| % | osztás maradéka |
| + | összeadás |
| - | kivonás |

Próbáld ki a műveleteket a böngésző konzoljában:



A műveleti sorrend a matematikában megszokott:

1. Először a zárójelekben lévő kifejezéseket értékeli ki.
2. Utána következnek a szorzások és az osztások.
3. Végül pedig az összeadásokat és a kivonásokat végzi el.

Az azonos szinten lévő műveleteket balról jobbra haladva értékeli ki.

Mi lesz a következő kifejezés értéke: (5 + 5) / 2 + (6 \* 4 / 4)  
Először számold ki, azután ellenőrizd a konzolon!

Az értékadást össze lehet vonni a műveletekkel. Például a következő utasítások ugyanazt jelentik:  
x = x + 1;  
x += 1;

Vajon mit jelentenek a következők?  
x -= 2;  
x \*= 3;  
x /= 5;  
x %= 4;

Az inkrementáló és a dekrementáló oprátorokat is lehet használni a JavaScriptben:   
x++; // veszi x értékét, majd megnöveli 1-egyel  
x--; // veszi x értékét, majd csökkenti 1-egyel

Próbáld ki ezeket a műveleteket is a konzolon!

Vigyázat, ezeket csak változókra lehet alkalmazni, számokra nem! Például ennek semmi értelme: 5++

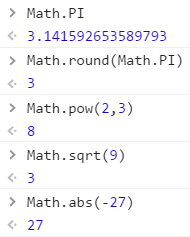
## Matematikai függvények

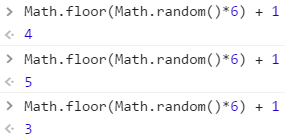
A JavaScriptben a matematikai függvények és állandók az előre definiált Math objektum metódusaiként állnak rendelkezésünkre.

A következő táblázat mutat néhány fontosabb metódust:

|  |  |
| --- | --- |
| Math.PI | 3.141592653589793 (ez nem metódus, hanem állandó) |
| Math.round() | Kerekítés egészre. |
| Math.floor() | Lefelé kerekítés |
| Math.ceil() | Felfelé kerekítés |
| Math.pow(x,y) | Hatványozás (x az y-odikon). JS6-tól: x \*\* y |
| Math.sqrt(x) | Négyzetgyök. |
| Math.abs(x) | Abszolút érték. |
| Math.random() | Véletlen szám 0 és 1 között. 0 lehet, 1 nem. 1 és 6 közötti véletlen szám: előállítása  Math.floor(Math.random()\*6) + 1; |

Próbáld ki a metódusokat a konzolon:

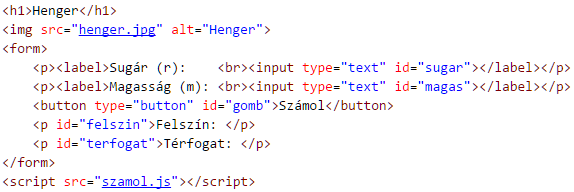




## Előkészületek

1. Nyisd meg a *henger* 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!

## Űrlap készítése

1. Egészítsd ki a HTML kódot egy formmal és a rajta lévő elemekkel:  
     
   A button elemnél muszáj megadni a típust, mert másképp nem működik helyesen!
2. Próbáld ki az oldalt! Írj be számokat a sugárhoz és a magassághoz! A script még nem működik.

## Script elkészítése

Az adatok beolvasását, a kerület kiszámítását és az eredmények kiírását egy szamol() nevű függvénnyel fogjuk elvégeztetni. A szükséges képletek:  
A = 2r(r+m)  
V = r2m

Utána ezt a függvényt hozzárendeljük a gombra kattintás eseményhez.

1. Készítsd el a szamol() függvényt a *szamol.js* fájlban:  
   
2. Rendeld hozzá a függvényt a gombra kattintáshoz (ezt a függvény után írd):  
   
3. Írj be számokat, majd próbáld ki a gomb működését!

## Önálló feladat 1.

Készíts weblapot a Pithagorasz-tétel szemléltetésére! Egy űrlapon lehessen bevinni egy derékszögű háromszög két befogóját (a és b), majd egy gomb megnyomásakor számítsa ki és írja ki az átfogó értékét (c)!

## Önálló feladat 2.

Töltsd fel egy online tárolóhelyre ezt a feladatsort és az órán készített fájlokat! Otthon is nézd át őket!