

Térbeli (3D-s) transzformációk

Bár a `transform` (átalakít) néhány értéke, vagyis a transzformációs függvények lehetővé teszik az elemeknek a háromdimenziós koordináta-rendszerben történő megváltoztatását, maga az elem nem 3D-s tárgy, hanem csak a 2D-s síkon létezik és nincsen mélysége. A harmadik dimenzió alkalmazásával egy olyan pontot hozunk létre, amely a pozitív (azaz a nézőhöz közelebbi) Z-koordináták X és Y értékeit felnagyítja, a negatív (távolabbi) Z-koordináták X és Y értékeit lekicsinyíti, és ezzel előállítja a mélység látszatát. *(A térbeli transzformációkat egyes böngészők a -webkit- előtaggal tudják csak értelmezni, egyes böngészők pedig még azzal sem!)*

Az eltolás síkbeli megvalósítása térbeli kiterjesztéssel

kódolás	leírás	példa
translateZ(ertek)	eltolás vízszintesen a Z tengely mentén, ahol a pozitív szám a néző felé történő közelítést, a negatív szám pedig a nézőtől történő távolítást jelenti	<pre>#div_tZ { transform: translateZ(50px); }</pre>
translate3d(x,y,z)	a térbeli eltolás összevont alakja , amelyben csak az x és y érték lehet %-ban megadott szám, a z értéke nem	<pre>#div_t3D { transform: translate3d(10px,-10px,15px); }</pre>

Az átméretezés síkbeli megvalósítása térbeli kiterjesztéssel

kódolás	leírás	példa
scaleZ(ertek)	a felhasználó irányába történő átméretezés	<pre>#div_nagyitZ { transform: scaleZ(2); }</pre>
scale3d(x,y,z)	a térbeli átméretezés összevont alakja	<pre>#div_meretez3D { transform: scale3d(1.2,0.7,2.5); }</pre>

Az elforgatás síkbeli megvalósítása térbeli kiterjesztéssel

(demo: [https://developer.mozilla.org/en-US/docs/Web/CSS/transform-function/rotate3d\(\)](https://developer.mozilla.org/en-US/docs/Web/CSS/transform-function/rotate3d()))

kódolás	leírás	példa
rotateX(szogertek)	<i>X tengely körüli elforgatás a megadott szöggel</i>	<pre>#forgat_X { transform: rotateX(45deg); }</pre>
rotateY(szogertek)	<i>Y tengely körüli elforgatás a megadott szöggel</i>	<pre>#forgat_Y { transform: rotateY(30deg); }</pre>
rotateZ(szogertek)	<i>Z tengely körüli elforgatás a megadott szöggel</i> (megegyezik az X-Y tengelyek síkjában történő elforgatással, azaz a rotate(deg) függvénnyel)	<pre>#forgat_Z { transform: rotateZ(20deg); }</pre>
rotate3d(x,y,z,deg)	<i>térbeli elforgatás összevont alakban</i> , ahol az első három szám 0 és 1 közötti érték az elforgatás tengelyeit határozza meg*, az utolsó érték pedig az elforgatás szöge (deg)	<pre>#forgat_3D { transform: rotate3d(1,1,1,45deg); }</pre>

Az elferdítéseknek nincsen térbeli kiegészítése!

Összevont 3D-s műveletek megadása: `matrix3d()`, amely egy 4x4-es transzformációs mátrixot alkot, a paraméterül kapott (vesszővel elválasztott) tizenhat értékkel.

* A 3D térben a forgásoknak három szabadságfoka van, amelyek együttesen egyetlen forgástengelyt írnak le. A forgástengelyt egy [x, y, z] vektor határozza meg, és az origó adja át (a transzformáció-origó tulajdonság által meghatározott módon). Ha a vektor a specifikáció szerint nincs normalizálva (vagyis ha három koordinátájának négyzetének összege nem 1), akkor a felhasználói ügynök azt belül normalizálja. Egy nem normalizálható vektor, például a nullvektor, [0, 0, 0], a forgatás figyelmen kívül hagyását eredményezi, anélkül, hogy érvénytelenítené a teljes CSS tulajdonságot.

Egyéb műveletek

A térbeli jellegből adódóan két további, a síkbeli műveleteknél nem értelmezett új tulajdonság egészíti ki a transzformációkat: a hátlap láthatósága és a transzformáció stílusa.

Hátlap láthatósága: mivel a 3D-műveletekkel egy elem részben vagy teljes mértékben megfordítható, a hátoldala is láthatóvá válik. A `backface-visibility` tulajdonság alapértelmezetten láthatóra hagyja a hátoldalt (`visible`), ha az egy transzformáció során előtérbe kerülne; ha viszont nem akarjuk láthatóvá tenni a hátoldalt, akkor el kell rejtetni (`hidden`).

Transzformáció stílusa: amennyiben egy elemen térbeli transzformációt hajtunk végre, akkor az elem gyerekeinek viselkedését a rajtuk végrehajtandó esetleges további transzformációk szempontjából is befolyásolhatjuk, azaz a `transform-style` tulajdonsággal azt határozhatjuk meg, hogy a a transzformált elem gyermekei hogyan viselkedjenek a transzformáció során. Ha azt szeretnénk, hogy az elem összes gyermeke bekerüljön a transzformált elemünk 2D-s síkjába, akkor a `flat` értéket kell beállítani, ha viszont a gyermekelemmel meg szeretnénk őriztetni annak 3D-s perspektíváját, akkor a `preserve-3d` beállítást kell használni