



# SAHIFA TRANSFORMATSIYASI. ANIMATSIYALAR

## Reja:

1. Transformatsiya
2. O'tishlar
3. Animatsiyalar

# Transformasiya

**CSSdagi transformatsiyalar** - bu elementni ma'lum bir shakldan boshqa shaklga o'tish imkon beradigan funktsiyalar to'plamidir:

- **translate:** elementni uchta o'q x, y va z (koordinatalar o'qi) bo'ylab harakatlantiradi;
- **rotate:** elementni markaz nuqtasi atrofida aylantiradi ;
- **scale:** elementning o'lchamini o'zgartiradi;
- **skew:** elementni qiyshaytiradi.

# Transformatsiya xususiyatlari

Transformatsiya qilish uchun uchta xususiyat mavjud:

01

`transform` qaysi funktsiya qo'llanilishini aniqlaydi (translate, rotate, scale va hokazo);

02

`transform-origin`  
transformatsiyaning boshlang'ich nuqtasini o'zgartirishga imkon beradi (background-position kabi ishlaydi);

03

`transform-style` 3D o'lchamda moslashtirish uchun ishlatiladi.

# translate () funksiyasi

translate() funktsiyasi elementni tekislikda (x va y o'qlari bo'ylab) harakatlantirishga imkon beradi. Bu funksiya ikki xil parametr:

- **bitta** parametr: elementni x o'qi bo'ylab harakatlantirish;
- **ikkita** parametr: **x** o'qi uchun birinchi qiymat, ikkinchisi **y** o'qi uchun kiritiladi.

Bu left va top qiymatlar bilan nisbiy joylashishni aniqlashga o'xshaydi.

Keling translate funksiyasi yordamida yopiq animatsiyamizni left/top joylashish bo'yicha harakatga keltiramiz :

```
HTML ▼
1 <p>Salom</p>
2 |
```

Esda tuting: **transform** bu xususiyat, **translate()** esa qiymat.

translateX () va translateY () qiymatlaridan ham navbati bilan foydalanish mumkin.

CSS ▼

```
1 p {
2   animation: translating 4s linear 0s infinite;
3   background: turquoise;
4   color: white;
5   height: 60px;
6   line-height: 60px;
7   margin: 0;
8   position: absolute;
9   text-align: center;
10  width: 60px;
11 }
12 @keyframes translating {
13   0% { transform: translate(0, 0); }
14   25% { transform: translate(240px, 0); }
15   50% { transform: translate(240px, 140px); }
16   75% { transform: translate(0, 140px); }
17   100% { transform: translate(0, 0); }
18 }
```

# rotate( )

rotate() funksiyasi elementni nuqta atrofida aylantirishga imkon beradi. Standart bo'yicha aylanish elementning atrofida bo'ladi. Buni aylanma stolda o'ynaydigan fonograf yozuvlaridek tushunish mumkin.

rotate() faqat bitta parametrni oladi, ya'ni bu burchak qiymati (graduslar (deg), radianlar (rad) yoki aylanishlar tezligi (turn)) bilan belgilanadi.

```
HTML ▼  
1 <p>Noshirlik</p>  
2  
3  
4
```

```
CSS ▼  
1 p {  
2   animation: rotating 4s linear 0s infinite;  
3   background: slateblue;  
4   border-radius: 60px;  
5   color: white;  
6   height: 120px;  
7   line-height: 120px;  
8   margin: 0;  
9   text-align: center;  
10  width: 120px;  
11 }  
12  
13 @keyframes rotating {  
14   0% {  
15     transform: rotate(0deg);  
16   }  
17   100% {  
18     transform: rotate(360deg);  
19   }  
20 }
```

# Scale ()

Scale() funksiyasi element o'lchamini o'zgartirishga imkon beradi. Bu elementni kattalashtirishi yoki kichraytirishi mumkin. Funktsiya quyidagilarni qabul qiladi:

- **bitta** parameter yordamida: elementning o'lchamini balandligi va kengligi bo'yicha teng ravishda o'zgartirish;
- **ikkita** parameter yordamida: birinchi qiymat elementni gorizontal, ikkinchisi vertikal o'lchamini o'zgartiradi.

Mumkin bo'lgan qiymatlar oralig'i:

- **1**: obyekt asl hajmini saqlab qoladi;
- **2**: obyekt ikki baravar kattalashadi;
- **0,5**: element yarim barobar kichrayadi;
- **0**: element yo'qoladi (chunki uning balandligi va kengligi nolga teng);
- **-1**: element teskari aks etadi.

translate() bo'lgani kabi, scale()da gorizontal ravishda vertikal ravishda o'lchamlarini o'zgartirish uchun x va y: scaleX () va scaleY () qiymatlarni ishlatish mumkin.

# skew ( )

skew ( ) funksiyasi elementlarni yon tomonlarini chiziqlar bo'ylab siljitish orqali qiyshaytirib ko'rsatishga imkon beradi .

Ushbu konvertatsiya qilish funksiyasi kamdan kam qo'llaniladi, chunki uning natijalari jozibador bo'lmaydi. Biroq, qanday ishlashini ko'rib chiqaylik.

skew ( ) funksiyasi scale( ) singari quyidagilarni qabul qiladi:

- **bitta parametr:** element gorizontal ravishda qiyshaytirish;
- **ikkita parametr:** birinchi qiymat elementni gorizontal ravishda, ikkinchisi vertikal ravishda qiyshaytiradi.

skew ( ) faqat gradus(deg) burchak qiymatlarini qabul qiladi.

# O'tishlar

CSSda **o'tish** bir tekis element boshqa holatga o'tishidir. Ya'ni bu jarayon animatsiya kabi birinchi holatdan so'ngi holatiga qadar o'zgarishli namoyishidir.

Quyidagilarni belgilab olish mumkin:

- **transition-property** – qanday xususiyat animatsiya qilinmoqda;
- **transition-duration** – qancha vaqt animatsiya davom etmoqda;
- **transition-timing-function** – oraliq holatlar qanday hisoblanmoqda;
- **transition-delay** – animatsiya qancha vatqdan so'ng boshlanmoqda.

Siz har bir xususiyatini alohida o'rnatishingiz yoki stenografiya ko'rinishidan foydalanishingiz mumkin.

O'tishlar ko'pincha **:hover** holati uchun ishlatiladi.

**:hoverda** CSS stillar bir zumda qo'llanish o'rniga, fon va matn rangi sekin o'zgaradi.

HTML ▼

```
1 <a>O'tishsiz</a>
2 <a class="with-transition">O'tishli</a>
```

CSS ▼

```
1 a {
2   background: lightgrey;
3   color: grey;
4   margin-right: 10px;
5   padding: 5px 10px;
6   transition: none;
7   cursor: pointer;
8   text-decoration: underline;
9 }
10
11 a:hover {
12   background: yellow;
13   color: red;
14 }
15
16 a.with-transition {
17   transition: 1s;
18 }
```

# transition-duration

**transition-duration** – bu o'tishni yaratish uchun zarur bo'lgan

yagona stil xususiyati. Qiymatlarni **soniyada** (1s)  
yoki **millisekundalarda** (100ms) o'rnatish mumkin.

HTML ▼

```
1 <a>O'tishsiz</a>
2 <a class="with-fast-transition">O'tish davomiyligi 0.5s</a>
3 <a class="with-slow-transition">O'tish davomiyligi 3s</a>
4 |
```

CSS ▼

```
1 a {
2   background: lightgrey;
3   color: grey;
4   margin-right: 10px;
5   padding: 5px 10px;
6   transition: none;
7   cursor: pointer;
8   text-decoration: underline;
9 }
10
11 a:hover {
12   background: yellow;
13   color: red;
14 }
15
16 a.with-fast-transition {
17   transition: 0.5s;
18 }
19
20 a.with-slow-transition {
21   transition: 3s;
22 }
23
```

# transition-timing-function

Oraliq holat vaqtini belgilash funktsiyasi **o'tish** paytidagi qiymatni hisoblaydi. Standart holatda **transition-timing-function** ease qiymatni qabul qiladi: boshida tezlashib va oxirida sekinlashadi.

```
<!--meta-->
<title></title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="shortcut icon" href="/favicon.ico">
<link rel="icon" href="/favicon.ico" type="image/x-icon">
<!--CSS-->
<link type="text/css" rel="stylesheet" href="/css/main.css">
<link rel="stylesheet" href="https://unpkg.com/normalize.css@8.0.1/normalize.css">
<link rel="stylesheet" href="/css/animation.css">
<link rel="stylesheet" href="css/transition.css">
```

HTML ▼

```
1 <main>
2 <p><strong>ease</strong>: sekin, tez, sekin</p>
3 <div class="ease"></div>
4 <p><strong>linear</strong>: bir xil tezlik</p>
5 <div class="linear"></div>
6 <p><strong>ease-in</strong>: sekin, tez</p>
7 <div class="ease-in"></div>
8 <p><strong>ease-out</strong>: tez, sekin</p>
9 <div class="ease-out"></div>
10 <p><strong>ease-in-out</strong>: ease kabi, juda tez va juda sekin</p>
11 <div class="ease-in-out"></div>
12 </main>
13
```

CSS ▼

```
1  ▾ div {
2    left: 0;
3    height: 20px;
4    width: 20px;
5    position: relative;
6    transition: 1s;
7  }
8  ▾ main:hover div {
9    left: 300px;
10 }
11 ▾ .ease {
12   transition-timing-function: ease;
13 }
14 ▾ .linear {
15   transition-timing-function: linear;
16 }
17 ▾ .ease-in {
18   transition-timing-function: ease-in;
19 }
20 ▾ .ease-out {
21   transition-timing-function: ease-out;
22 }
23 ▾ .ease-in-out {
24   transition-timing-function: ease-in-out;
25 }
26 ▾ div:nth-child(2) {
27   background: crimson;
28 }
29 ▾ div:nth-child(4) {
30   background: midnightblue;
31 }
32 ▾ div:nth-child(6) {
33   background: mediumseagreen;
34 }
35 ▾ div:nth-child(8) {
36   background: orangered;
37 }
38 ▾ div:nth-child(10) {
39   background: darkviolet;
40 }
41
```

ease: sekin, tez, sekin



linear: bir xil tezlik



ease-in: sekin, tez



ease-out: tez, sekin



ease-in-out: ease kabi, juda tez va juda sekin



# Animatsiyalar

Animatsiya xususiyatlari:

- **animation-name** : animatsiya nomi ;
- **animation-duration**: animatsiya qancha davom etadi;
- **animation-timing-function** : oraliq holatlar qanday hisoblanganligi;
- **animation-delay**: animatsiya biror vaqt o'tgach boshlanadi;
- **animation-iteration-count**: animatsiya necha marta takrorlanishi kerak;
- **animation-direction** : harakat teskari yo'nalishda ketishi kerakmi yoki yo'qmi;
- **animation-fill-mode**: animatsiya boshlanishidan oldin va tugagandan so'ng qanday stillar qo'llaniladi.



# @keyframes

HTML elementlarini harakatlantirishdan oldin, asosiy kadrlar yordamida animatsiya yozishingiz kerak . Umuman olganda, asosiy kadrlar animatsiyadagi har bir oraliqning qadamidir . Ular foizlar yordamida aniqlanadi:

- 0% - animatsiyaning birinchi bosqichi;
- 50% animatsiyaning yarim qadamidir;
- 100% bu oxirgi qadam.

Har bir asosiy kadr - bu CSS qoidasi , ya'ni odatdagidek CSS xususiyatlarini yozishingiz mumkin.

Animatsiyani aniqlash uchun @keyframes kalit so'zini va uning nomi bilan yoziladi:

```
@keyframes around {  
  0% { left: 0; top: 0; }  
  25% { left: 240px; top: 0; }  
  50% { left: 240px; top: 140px; }  
  75% { left: 0; top: 140px; }  
  100% { left: 0; top: 0; }  
}  
p { animation: around 4s linear infinite; }
```

# animation-name

Animatsiya nomi kamida ikki marta ishlatiladi :

- @keyframes;
- animation-name.

```
@keyframes whatever {  
  /* ... */  
}  
  
.selector { animation-name: whatever; }
```

CSS sinf nomlari singari, animatsiya nomlari ham quyidagilarni o'z ichiga olishi mumkin:

- harflar (a-z);
- raqamlar (0-9);
- pastki chiziq (\_);
- defis (-).

Nom raqam bilan yoki ikkita defis bilan boshlana olmaydi.

# animation-duration



O'tish davomiyligi singari , animatsiya davomiyligi **soniyalarda** (1 soniya) yoki **millisekundalarda** (200 soniya) o'rnatilishi mumkin.

```
.selector { animation-duration: 0.5s; }
```



**Uyga vazifa:** Animatsiyaning qolgan xususiyatlarini mustaqil o'rganiladi.

**E'tiboringiz  
uchun  
rahmat!**