



VILNIUSCODINGSCHOOL

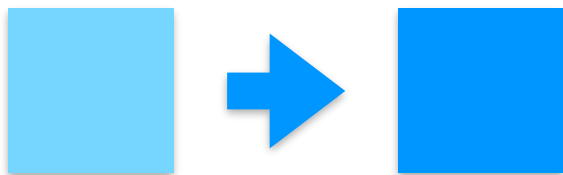
CSS Animacija

Transition

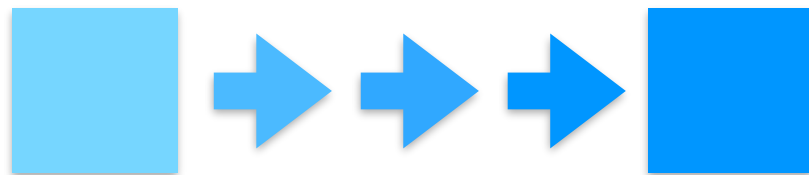


Transition perkeičia savybę(-es) ir trunka tam tikrą laiko tarpą.

Iškart



Transition



Transition - spalvos



Spalvas galime pakeisti su CSS

Transition pridedame prie startinės spalvos tokiu formatu:

transition: <savybė> <trukmė>;

Pirkti

```
.btn {  
    background-color: #84d2f9;  
}
```

```
.btn {  
    background-color: #84d2f9;  
    transition: background-color 0.8s;  
}
```

Pirkti

```
.btn:hover {  
    background-color: #0089ce;  
}
```

Transition - spalvos



Galime keisti kelias savybes tuo pačiu metu:

```
.btn {  
    background-color:#84d2f9;  
    color:#fff;  
    transition: background-color 0.8s, color 0.8s;  
}  
  
.btn:hover {  
    background-color:#0089ce;  
    color:#e3e3e3;  
}
```

Galime naudoti raktinį žodį **all**, jei norime pritaikyti transition visoms savybėms:

```
.btn {  
    background-color:#84d2f9;  
    color:#fff;  
    transition: all 0.8s;  
}  
  
.btn:hover {  
    background-color:#0089ce;  
    color:#e3e3e3;  
}
```

Transition formatas



Seka nesvarbi, tik trukmės skaičius privalo būti prieš delsimo skaičių.

transition: <saveybė> <trukmė> <funkcija> <delsimas>;

Default reikšmės:

<saveybė> - all

<funkcija> - ease

<delsimas> - 0

```
.btn {  
  background-color:#84d2f9;  
  color:#fff;  
  transition: all 0.8s ease 0;  
}
```



Galime default reikšmių ir
nerašyti (nebent norėtume jas
pakeisti)

```
.btn {  
  background-color:#84d2f9;  
  color:#fff;  
  transition: 0.8s;  
}
```

Transition prefiksai



Kai rašome transition, rekomenduotina rašyti su naršyklių prefiksais:

```
.btn {  
    background-color:#84d2f9;  
    color:#fff;  
    -webkit-transition: 0.8s; /* Safari, Chrome */  
    -moz-transition: 0.8s; /* Mozilla Firefox */  
    -ms-transition: 0.8s; /* Internet Explorer */  
    -o-transition: 0.8s; /* Opera */  
    transition: 0.8s;  
}
```

Transition - pozicija



Transition taip pat naudojamas paslėpto turinio įkėlimui. Padarysime mygtuką, kuriame pasikeičia tekstas, kai užvedame ant jo su pele.

PIRKITE ŽAIDIMĄ

Skriskite, bėkite ir šokinėte kartu su Stabsiu ir jo draugais šiame nuotaikingame žaidime.

Spauskite mygtuką "Pirkti" ir įsigykite mūsų žaidimą dabar!

Žaidimą "Stabsis ir jo draugai" šiuo metu galite įsigyti tik internetu.

Pirkti 5.99€



PIRKITE ŽAIDIMĄ

Skriskite, bėkite ir šokinėte kartu su Stabsiu ir jo draugais šiame nuotaikingame žaidime.

Spauskite mygtuką "Pirkti" ir įsigykite mūsų žaidimą dabar!

Žaidimą "Stabsis ir jo draugai" šiuo metu galite įsigyti tik internetu.

Nedelsk!

Transition - pozicija



```
.top {
  top:0;
  opacity: 1;
}

.bottom {
  top:20px;
  opacity: 0;
}

.btn:hover .top {
  top:-20px;
  opacity: 0;
}

.btn:hover .bottom{
  top:0;
  opacity: 1;
}
```

```
<a href="#" class="btn btn-pirkti">
  <span class="top content">Pirkti 5.99€</span>
  <span class="bottom content">Nedelsk!</span>
</a>
```

Tekstas “Pirkti 5.99 €” yra mygtuke, o tekstas “Nedelsk” po mygtuko. Tuomet, hover “Pirkti 5.99 €” pakeliame 20 px virš mygtuko, o “Nedelsk” pakeliame 20 px į mygtuko vidų. Su opacity paslepiame tai, kas už mygtuko ribų. Pridedame transition, kad perėjimas būtų gražus.

(failai transition.html ir style.css)

```
.btn {
  position:relative;
}

.content {
  position: absolute;
  transition: 0.3s;
  overflow: hidden;
}
```


Transition - visibility



Kai paspaudžiame mygtuką, norėtume, kad iššoktų tekstas.

Pradinius parametrus nustatome `opacity:0;` ir `visibility:hidden;`

Kai mygtukas paspaudžiamas, pridedame naują CSS klasę `.open`, kuri pakeičia `opacity:1` ir `visibility:visible;`

`opacity` ir `visibility` naudojame kartu, nes `opacity` gali paslėpti turinį, bet tas elementas vis dar užima tą patį plotį ir aukštį. `visibility` padaro elementą transparent mygtuko paspaudimams.

`display` neturi tarpinio varianto - pradinis yra `none` ir tada turime `block`, todėl negalime daryti `transition`. Tuo tarpu `opacity` turi tarpinį variantą - pradinis yra `0` ir tada turime `1`, o tarp jų galime pasirinkti tarpinį variantą, pvz. `0.5`.

(failai: `visibility.html` ir `visibility.scss`)

Transform



transform leidžia mums keisti elementus - pasukti, padidinti, pastumti, sumažinti.

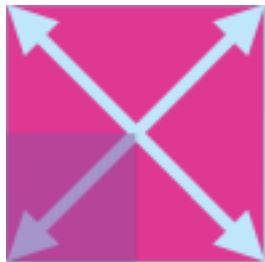
Pasukti

`transform: rotate(1turn);` arba `transform: rotate(360deg);`
`transition: transform 2s;`

(failai: transform.html ir transform.css)

Padidinti

`transform: scale(2);`



`transform: scale(4, 2);`



Taip pat galime rašyti x ir y atskirai: `transform: scaleX(4);` ir `transform: scaleY(2);`

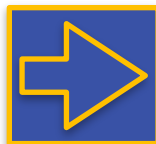
Transform



transform leidžia mums keisti elementus - pasukti, padidinti, pastumti, sumažinti.

Pastumti

`transform: translateX(3px);`



pastumia elementą 3px į dešinę

`transform: translateY(-3px);`



pastumia elementą 3px į viršų

`transform: translate(3px);`



pastumia elementą 3px į dešinę ir apačią

(failai transform.html ir transform.css)

Keyframes



keyframes - tai sąrašas viso to, kas turėtų įvykti animacijos metu. Tarkim, jei elementas turi startinę spalvą ir po transition pakeičia ją kita spalva, negalime nurodyti trečios spalvos per vidurį tos animacijos vien su transition. Keyframes leidžia mums nurodyti tarpinius parametrus.

Keyframe animacija turi dvi dalis:

1. Sukuriame animaciją:

```
@keyframes vespa {  
  0%    {left:100%;}  
  36%   {left:59%;}  
  40%   {left:60%;}  
  100%  {left:-10%;}  
}
```

2. Priskiriame animaciją elementui:

```
#vespa{  
  width:100px;  
  animation:vespa 5s infinite ease-out;  
  animation-fill-mode:forwards;  
  position:fixed;  
  bottom:0;  
  right:-10%;  
}
```

(failai *keyframe.html* ir *keyframe.css*)