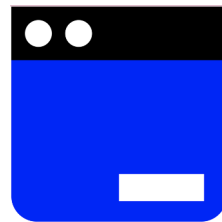




**Code  
Academy**



# CSS animacijos ir perėjimai

HTML & CSS

16 paskaita



# Paskaitos eiga



CSS animacijos



CSS animacijų  
manipuliavimas



CSS perėjimai ir jų  
manipuliavimas

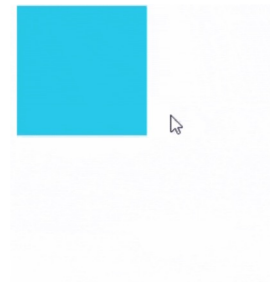
# CSS Perèjimai





# CSS perėjimai

CSS perėjimai leidžia sklandžiai keisti savybės vertes per tam tikrą laiką.



# Kaip panaudoti perėjimus?





## Perėjimo efektas

Norėdami sukurti perėjimo efektą, turite nurodyti 2 dalykus:

1. CSS savybę, kuriai norite pridėti efektą
2. Efekto trukmę

```
div {  
  width: 100px;  
  height: 100px;  
  background: red;  
  transition: width 2s;  
}
```

**Pastaba:** Jei trukmė nenurodyta, perėjimas prasmės neturi, nes numatytoji reikšmė yra 0.

## Kodas

```
<!DOCTYPE html>
<html>
<head>
  <style>
    div {
      width: 100px;
      height: 100px;
      background: violet;
      transition: width 2s;
    }
    div:hover {
      width: 300px;
    }
  </style>
</head>
<body>
  <h1>Perėjimo savybė</h1>
  <p>Norėdami pamatyti perėjimo efektą, užveskite
  pelės žymeklį ant toliau esančio div elemento:</p>
  <div></div>
</body>
</html>
```

## Rezultatas



### Perėjimo savybė

Norėdami pamatyti perėjimo efektą, užveskite pelės žymeklį ant toliau esančio div elemento:



Atkreipkite dėmesį, kad žymekliui *išėjus* iš elemento, jis palaipsniui pasikeičia į pradinį stilių.



## Pakeiskime kelias savybės vertes

Toliau pateiktame pavyzdyje pridedamas perėjimo efektas pločio ir aukščio savybėms, kurių trukmė yra 2 sekundės pločiui ir 4 sekundės aukščiui:

žr. kitoje skaidrėje.



## Kodas

```
<!DOCTYPE html>
<html>
<head>
  <style>
    div {
      width: 100px;
      height: 100px;
      background: violet;
      transition: width 2s, height 4s;
    }
    div:hover {
      width: 300px;
      height: 300px;
    }
  </style>
</head>
<body>
  <h1>Perėjimo savybė</h1>
  <p>Norėdami pamatyti perėjimo efektą, užveskite
  pelės žymeklį ant toliau esančio div elemento:</p>
  <div></div>
</body>
</html>
```

## Rezultatas



### Perėjimo savybė

Norėdami pamatyti perėjimo efektą, užveskite pelės žymeklį ant toliau esančio div elemento:





# Nurodykite perėjimo greičio kreivę

`transition-timing-function` nurodo perėjimo efekto greičio kreivę.

Funkcija gali turėti šias reikšmes:

`ease` – lėta pradžia, tada greitai ir lėta pabaiga (default)

`linear` – perėjimo efektas *linijiniu* greičiu nuo pradžios iki pabaigos

`ease-in` – perėjimo efektas su lėta pradžia

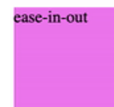
`ease-out` – perėjimo efektas su lėta pabaiga

`ease-in-out` – nurodo perėjimo efektą su lėta pradžia ir pabaiga

`cubic-bezier(n,n,n,n)` – leidžia apibrėžti savo reikšmes

## transition-timing-function savybė

Užveskite pelės žymeklį ant toliau pateiktų div elementų, kad pamatytumėte skirtingas greičio kreives:



Kaip panaudoti perėjimus?



## Perėjimo uždelsimas

`transition-delay` nurodo  
perėjimo efekto uždelsimą  
(sekundėmis).

## Kodas

```
<!DOCTYPE html>
<html>
<head>
  <style>
    div {
      width: 100px;
      height: 100px;
      background: violet;
      transition: width 3s;
      transition-delay: 1s;
    }
    div:hover {
      width: 300px;
    }
  </style>
</head>
<body>
  <h1>transition-delay savybė</h1>
  <p>Norėdami pamatyti perėjimo efektą, užveskite
  pelės žymeklį ant toliau esančio div elemento:</p>
  <div></div>
  <p><b>Pastaba:</b> Perėjimo efektas turi 1 sekundės
  uždelimą, todėl luktelkite:</p>
</body>
</html>
```

## Rezultatas



### transition-delay savybė

Norėdami pamatyti perėjimo efektą, užveskite pelės žymeklį ant toliau esančio div elemento:



**Pastaba:** Perėjimo efektas turi 1 sekundės uždelimą, todėl luktelkite:)

# CSS Animacija





# CSS animacija

CSS animacija leidžia animuoti perėjimus iš vienos CSS stiliaus į kitą.



Animacijas sudaro 2 komponentai:

1. **Stilius**, apibūdinantis CSS animaciją
2. **Pagrindinių kadro rinkinys**,  
(angl. keyframes) nurodantis  
animacijos stiliaus pradžios ir pabaigos būsenas

# @keyframes

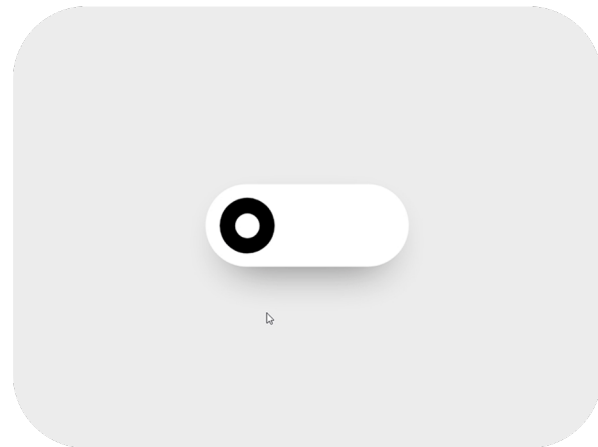




## @keyframes taisyklė

Kai nurodote CSS stilius `@keyframes` taisyklėje, tam tikru metu animacija palaipsniui pasikeis iš esamo stiliaus į naują.

Kad animacija veiktų, turite susieti animaciją su elementu.





## Kodas

```
/* Animacijos kodas */  
@keyframes example {  
  from {background-color: red;}  
  to {background-color: yellow;}  
}  
  
/* Elementas ant kurio naudojame animaciją */  
div {  
  width: 100px;  
  height: 100px;  
  background-color: red;  
  animation-name: example;  
  animation-duration: 4s;  
}
```

## Rezultatas



## CSS Animacija



**Pastaba:** Kai animacija pasibaigs, grįš į pradinį stilių.



# animation-duration

**Pastaba:** `animation-duration` ypatybė apibrėžia, kiek laiko turi trukti animacija. Jei `animation-duration` nenurodyta, animacija nebus rodoma, nes numatytoji reikšmė yra 0 sekundžių.

Prieš tai pateiktame pavyzdyje nurodėme, kada stilius pasikeis, naudodami raktinius žodžius `from` ir `to` (tai reiškia 0 % (pradžią) ir 100 % (užbaigtą)).

Galima naudoti ir procentus. Naudodami procentus galite pridėti tiek stiliaus pakeitimų, kiek norite.



# animation-delay

`animation-delay` ypatybė nurodo kiek animacija dels, kol pradės veikti, nuo jos animacijos pradžios.

## Kodas

```
/* Animacijos kodas */
@keyframes example {
  0%   {background-color:#77A33E;left:0px;top:0px;}
  25%  {background-color:#D1FF94;left:200px;top:0px;}
  50%  {background-color:#BAF073;left:200px;top:200px;}
  75%  {background-color:#8F2EA3;left:0px;top:200px;}
  100% {background-color:#DA73F0;left:0px;top:0px;}
}

/* Elementas ant kurio naudojame animaciją */
div {
  width: 100px;
  height: 100px;
  background-color:#77A33E;
  position: relative;
  animation-name: example;
  animation-duration: 4s;
  animation-delay: 2s;
}
```

## Rezultatas



### CSS Animacija

animation-delay ypatybė nurodo delsimą nuo animacijos pradžios. Toliau pateiktame pavyzdyje animacija pradeda po 2 sekundžių:





# animation-timing-function

`animation-timing-function`  
ypatybė nurodo animacijos greičio  
kreivę.



## animation-timing-function ypatybės reikšmės

`ease` – animacija su lėta pradžia, tada įsibėgėja ir baigiasi suletėdama (default)

`linear` – juda tuo pačiu greičiu nuo pradžios iki pabaigos, linijiškai

`ease-in` – lėta pradžia

`ease-out` – lėta pabaiga

`ease-in-out` – nurodo animaciją su lėta pradžia ir pabaiga, bet greitesniu viduriu

`cubic-bezier(n,n,n,n)` – leidžia apibrėžti savo reikšmes



# Klausimai?



# Iki kito karto!

