

Трансформації

Трансформації дозволяють:

- пересувати елементи
- змінювати масштаб
- повертати елементи

За умовчанням трансформація відбувається щодо центру елемента,
але цю поведінку можна змінити

Трансформований елемент - це елемент із будь-яким встановленим значенням властивості *transform*.

Види:

- `transform: translate()`
- `transform: rotate()`
- `transform: scale()`
- `transform: slew()`

CSS-переходи

CSS-переходи дозволяють анімувати зміну значення властивості з часом, керуючи швидкістю цього переходу.

Зміна відбувається при настанні певної події, наприклад *hover*

Перехід має два стани – *початкове* та *кінцеве* значення властивості

Властивості:

- `transition-property`
- `transition-duration`
- `transition-timing-function`
- `transition-delay`

CSS-анімація

CSS-анімація надає веб-сторінкам динамічність.

На відміну від переходів:

- анімації не потрібен обов'язковий ініціатор (подія)
- її можна повторити нескінченну кількість разів
- між початковим та кінцевим станами може бути будь-яка кількість проміжних станів

Оголошується директивою *@keyframes*

Властивості:

- `animation-name`
- `animation-duration`
- `animation-timing-function`
- `animation-delay`
- `animation-iteration-count`

HOW BROWSER RENDERING WORKS

Javascript

Style

Layout

Paint

Composite

- **Javascript**

Often JS kicks things off by triggering animation changes (often CSS too)

- **Style Calculations**

The browser figures out which styles apply to which elements, based off of your selectors

- **Layout**

Browser calculates how much space each element takes up and where each element should go

- **Painting**

The browser fills in the pixels on the screen, painting all the text, colors, borders, images, etc. This is often done across multiple layers.

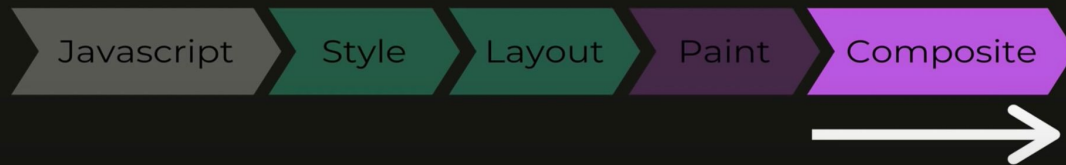
- **Compositing**

The browser takes the multiple layers it painted and combines them into one to draw on the screen. It makes sure overlapping elements are displayed correctly.

1. **JavaScript** - скрипт або подія, наприклад хOVER або фокус, запускають анімацію стилів елемента
2. **Style** - відбувається обчислення нових стилів елементів, розраховується специфічність, конфлікти, каскадування тощо
3. **Layout** - розрахунок геометрії елементів. При анімації властивості, що впливає на геометрію, наприклад margin, браузер обчислює нову позицію цілої групи елементів
4. **Painting** - промальовування шарів, тому що елементи вкладені один в одного або спозиціоновані
5. **Composite** - компонування всіх підготовлених шарів та малювання фінальної картинки на екрані

Animating Opacity/Transform

Skips layout and painting!





Дмитро Рябовол

- Markup розробник в Studytube
- Викладач HTML/CSS технологій
- 5+ років досвіду
- Блогер, web-фрілансер

Instagram: [dima_riabovol](#)

LinkedIn: [Dmytro Riabovol](#)

Web-site: [dimariabovol.pro](#)

Успіхів!