

CSS Анимация

WEB
COURSE
ORT DNIPRO

ORTDNIPRO.ORG/WEB

1. Анимация средствами CSS

Два подхода к анимации в CSS

Анимация переходов

плавный переход между различными значениями стилевых свойств, базируется на свойстве **transition**.

Анимация «по сценарию»

заранее определенная последовательность действий, которые проигрываются во время выполнения анимации, базируется на свойстве **animation**.

2. Анимация «по сценарию»

```

16 @keyframes my-script{
17     0%{
18         background: yellow;
19         transform: rotate(0deg) ;
20     }
21
22     70% {
23         background: blue;
24         transform: rotate(45deg) ;
25     }
26
27     90%{
28         background: orange;
29         transform: rotate(-15deg) ;
30     }
31
32     100%{
33         background: yellow;
34         transform: rotate(0deg) ;
35     }
36 }

```

Анимация «по сценарию»

Анимация «по сценарию» состоит из двух шагов. **Первый шаг:** создается «сценарий» при помощи директивы **@keyframes** задаются «опорные кадры» т.е. состояния через которые наша анимация должна проходить. Кадры нумеруются в **процентах** (для анимации между двумя состояниями возможно использовать **from - to**), за счёт этого не только определяется последовательность кадров но и рассчитываются время необходимое на проигрывание перехода между кадрами. У сценария должно быть указано имя.

Анимация «по сценарию»

Анимация «по сценарию» состоит из двух шагов. **Второй шаг** включает проигрывание анимации для тега. Для этого с помощью с помощью соответствующих свойств указать: название сценария (**animation-name**), продолжительность анимации (**animation-duration**), функция распределения времени (**animation-timing-function**), задержка начала анимации (**animation-delay**), количество повторений анимации (**animation-iteration-count** – от **1** до бесконечности **infinite**), направление анимации (**animation-direction**), и какие стили применять по завершению анимации (**animation-fill-mode**). Или же можно использовать общее свойство **animation** позволяющее одной строкой задать все параметры анимации.

```
3  div{
4      animation-name: my-script;
5      animation-duration: 3s;
6      animation-timing-function: linear;
7      animation-delay: 1s;
8      animation-iteration-count: infinite;
9      animation-direction: alternate;
10     animation-fill-mode: forwards;
11
12     /* Или же */
13
14     animation: my-script 3s linear 1s infinite alternate forwards;
15 }
```

3. Немного практики

Анимация по «сценарию»



Состоит из двух шагов:

Первый шаг: создается «сценарий» при помощи директивы **@keyframes** задаются «опорные кадры» т.е. состояния через которые наша анимация должна проходить.

Второй шаг: при помощи свойства **animation** «сценарий» крепиться к какому-либо элементу.

Воспользуйтесь разметкой в репозитории занятия: [***./src/helicopter-template***](#)

4. Анимация в **Font Awesome**

Анимация в Font Awesome



```
<div class="fa-3x">
  <i class="fas fa-spinner fa-spin"></i>
  <i class="fas fa-circle-notch fa-spin"></i>
  <i class="fas fa-sync fa-spin"></i>
  <i class="fas fa-cog fa-spin"></i>
  <i class="fas fa-spinner fa-pulse"></i>
  <i class="fas fa-stroopwafel fa-spin"></i>
</div>
```

Библиотека **Font Awesome** содержит ряд эффектов которые могут быть полезным, в первую очередь для анимация для загрузчиков.

<https://fontawesome.com/docs/web/style/animate>

Воспользуйтесь разметкой в репозитории занятия: ***./src/fa-template***

Необязательное))

Домашнее задание

Домашнее задание



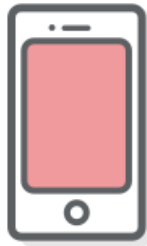
У вас есть изображение дороги и автомобиля. Создайте «CSS-мультфильм» о том как автомобиль едет по дороге. *П.С. можно добавить еще автомобилей...*

П.С. Задайте картинку размер 1000x1000 px и не замораживайтесь вопросом масштабирования.

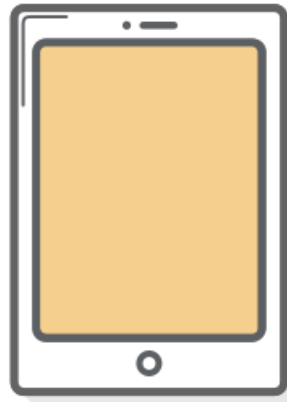
Воспользуйтесь изображениями в репозитории занятия: [./src/homework](#)

На следующем занятии

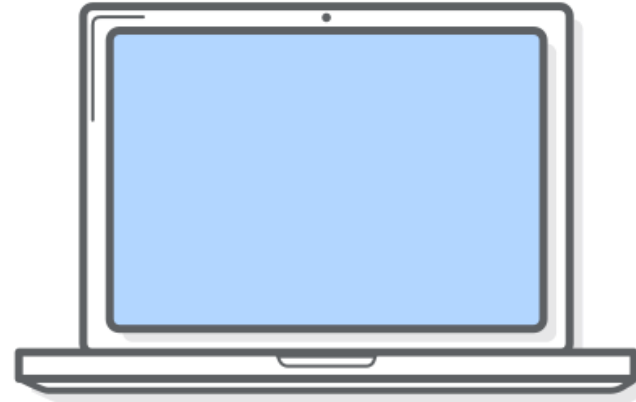
Адаптивность



MOBILE



TABLET



DESKTOP

Построение адаптивной вёрстки