**КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ**

# Переходы (transition)

Чтобы переход имел место, элемент должен получить изменение состояния и для каждого состояния должны быть определены разные стили.

Самый простой способ задать стили для различных состояний — это воспользоваться псевдоклассами :hover, :focus, :active и :target.

Есть четыре свойства, связанных с переходами в целом, это ***transition-property, transition-duration, transition-timing-function и transition-delay***. Не все они обязательны для создания перехода, наиболее популярны первые три.

В приведённом ниже примере блок изменит цвет фона в течение одной секунды в линейном виде.

|  |  |
| --- | --- |
| html | css |
|  |  |

*Смотри пример в папке z0\_transition.*

# Трансформация (transform)

Вместе с CSS3 появились новые способы позиционирования и изменения элементов. В настоящее время основные техники вёрстки могут быть пересмотрены с альтернативными методами для размера, положения и изменения элементов. Все эти новые техники стали возможными благодаря свойству ***transform***.

Свойство transform поставляется в двух разных наборах — двумерных и трёхмерных. В каждом из них есть свои индивидуальные свойства и значения.

Использование значения ***scale*** в свойстве transform позволяет изменить визуальный размер элемента. Значение масштаба по умолчанию равно 1, поэтому любое значение от .99 до .01 визуально уменьшает элемент, в то время как любое значение больше или равное 1.01 визуально увеличивает элемент.

|  |  |
| --- | --- |
| html | css |
|  |  |

*Смотри пример в папке z0\_transform.*

# Анимация (animate)

Переходы выполняют большую работу по созданию визуальных взаимодействий из одного состояния в другое и идеально подходят для этих видов одиночных изменений состояния. Тем не менее, когда требуется больший контроль, для переходов необходимо несколько состояний. С этого момента переходы уходят, а в дело вступает [анимация.](http://coding.smashingmagazine.com/2011/09/14/the-guide-to-css-animation-principles-and-examples/)

## **Ключевые кадры анимации**

Чтобы установить несколько точек, в которых элемент должен совершать переход, используется правило ***@keyframes***. Это правило включает имя анимации, любое число контрольных точек, а также свойства, предназначенные для анимации.

@keyframes slide {

0% { left: 0; top: 0; } 50% { left: 244px; top: 100px;

} 100% { left: 488px; top: 0;

}

}

*Смотри пример в папке z0\_keyframe.*

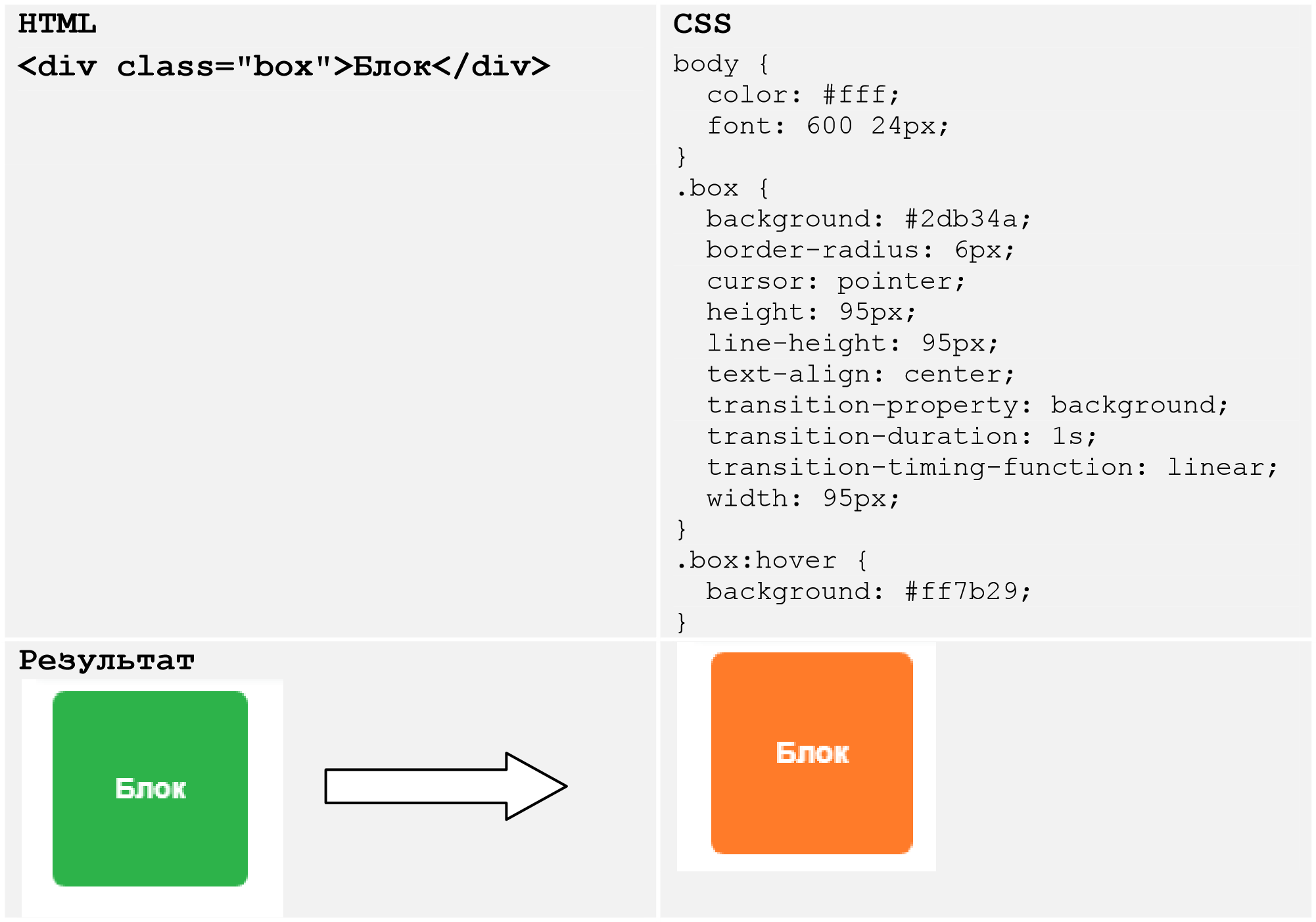
Анимация выше называется ***slide***, состояния начинаются сразу же после открытия правила ***@keyframes***. Разные контрольные точки ключевых кадров задаются помощью процентов, начиная с 0% и работая до 100%, с промежуточной точкой на 50%. При желании вместо 0% и 100% могут быть использованы ключевые слова from и to. Кроме 50% также могут быть указаны дополнительные контрольные точки. Свойства элементов для анимации перечислены внутри каждой контрольной точки, left и top в приведённом выше примере.

Важно отметить, как и с переходами, могут быть анимированы только отдельные свойства. Подумайте, как вы могли бы переместить элемент сверху вниз, например. Попытка анимировать от top: 0 до bottom: 0 не будет работать, потому что анимация может применять только переход в пределах одного свойства, а не от одного свойства к другому. В этом случае элемент необходимо анимировать от top: 0 до top: 100%.

**ВЫПОЛНЕНИЕ ЗАДАНИЯ ПО ОБРАЗЦУ:**

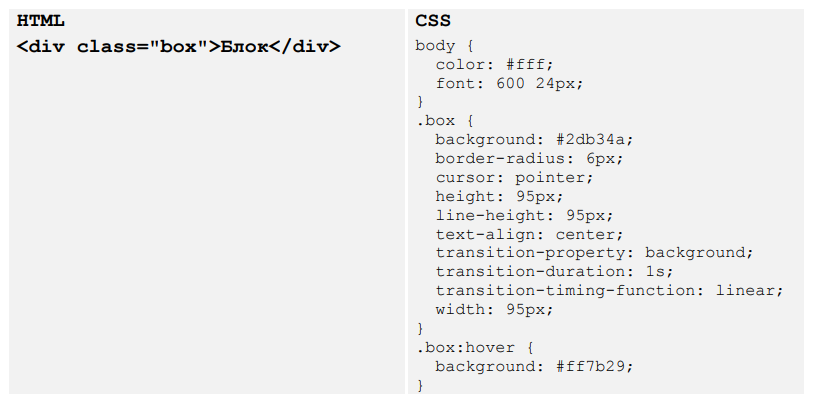
### **Задание 1** (Переходы)

В примере блок должен изменить цвет фона в течение одной секунды в линейном виде



**Задание 2** (Переходы)

В примере блок меняет свою форму:



**Результат**



**Задание 3** (Трансформация)

Вращение блоков:

****

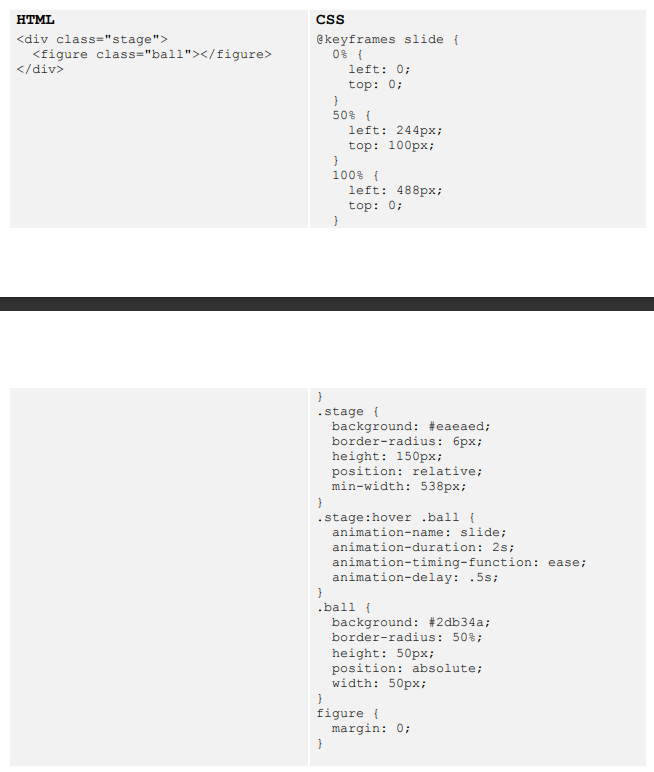
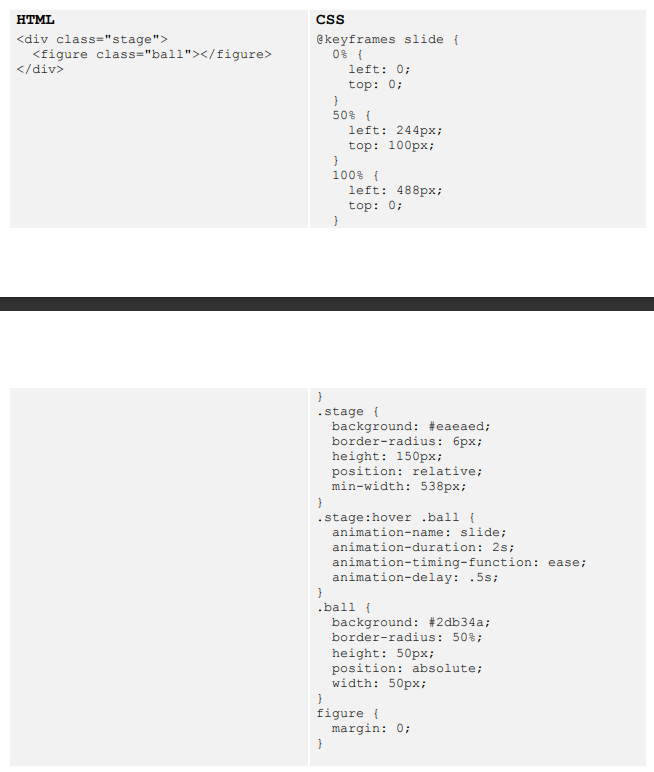
**Задание 4** (Трансформация)

Переворачивающаяся карточка

****

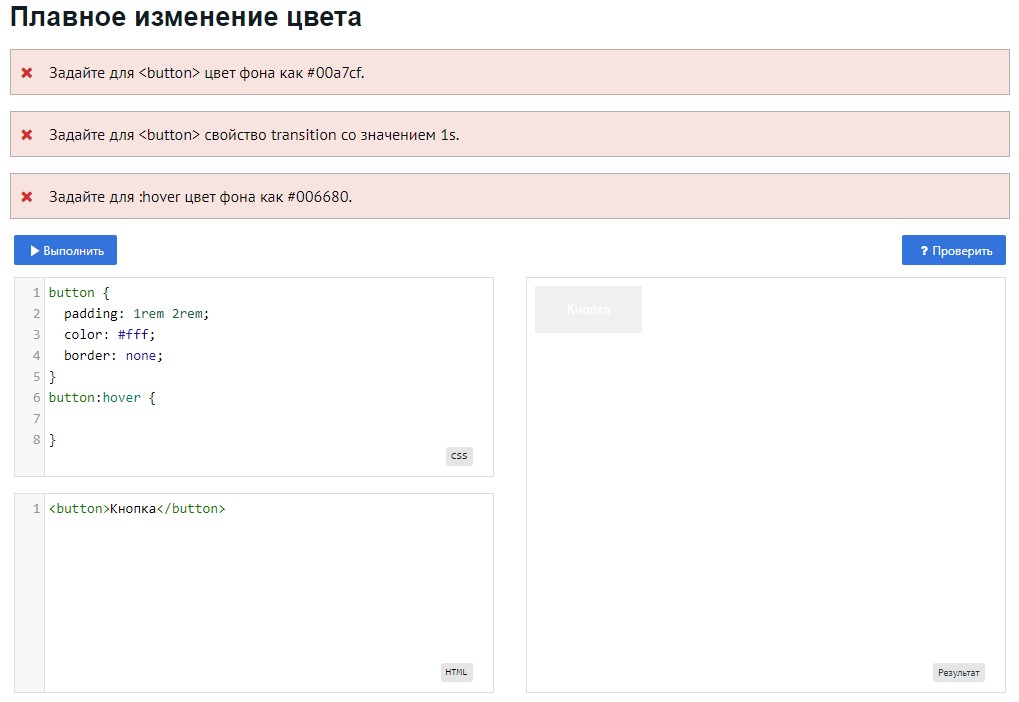
**Задание 5** (Анимация)

При наведении курсора на шарик происходит анимация.

# САМОСТОЯТЕЛЬНОЕ ВЫПОЛНЕНИЕ

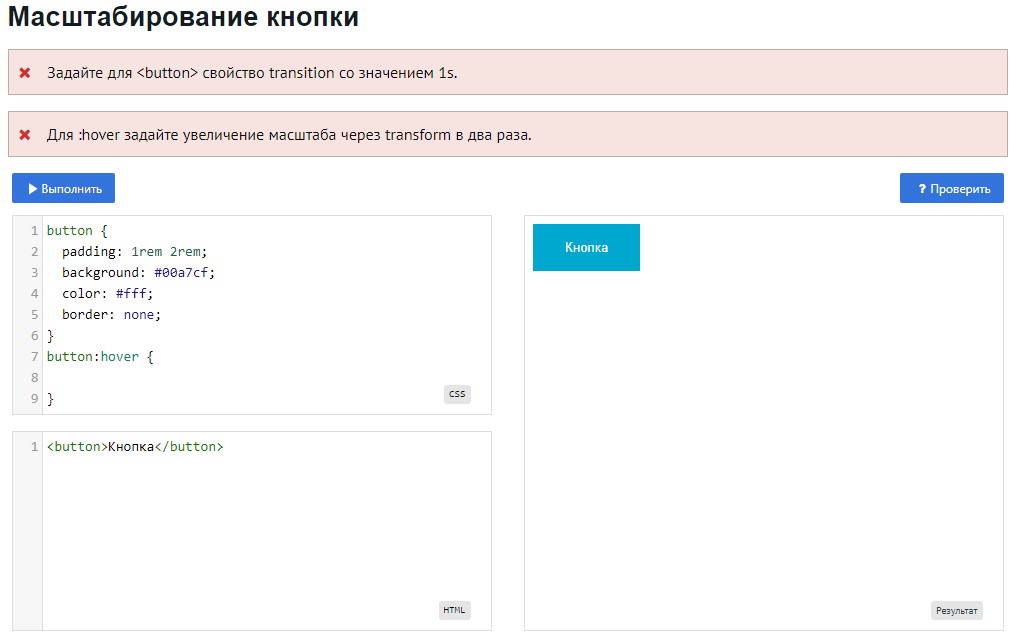
## **ЗАДАНИЕ 1** (Переходы)



Проверить правильность выполнения можно в онлайн редакторе

[(https://webref.ru/practice/2657)](https://webref.ru/practice/2657)

## **ЗАДАНИЕ 2** (Трансформация)



Проверить правильность выполнения можно в онлайн редакторе

[(https://webref.ru/practice/2658)](https://webref.ru/practice/2657)

## **ЗАДАНИЕ 3** (Анимация)

Переместить объект с помощью правила @keyframes.

