

 notes

F 1.8

# Website development



## Beginner level

Materials prepared by the department  
of methodological development department



 notes

front-end

beginner level

F 1.8

# Creating the My Games Hub Website

## ● Transform

**transform** is a CSS property that allows to translate, skew or scale an element. Can take the following values:

- **translate (x-offset, y-offset)** moves the element, separate values can be used to move the element along the vertical axis or horizontal axis (**translateX()**, **translateY()**);
- **scale()** changes the scale of the element. For example, **scale(0.5)** will decrease the element twofold. If you need to change the scale along x-axis or along y-axis, the values **scaleX()** and **scaleY()** can be used
- **rotate()** turns the element clockwise through a given number of degrees.  
For example, **rotate(45deg)** will turn the element through 45 degrees. To make the element turn anti-clockwise, a negative number should be specified **rotate(-90deg)**.

## ● Transform

- **scew()** skews the element sides along x-axis and y-axis. Takes the skew angle along the x and y axes as a value - **scew(10deg, 25deg)**. Also separate values for x (**scewX()**) and y (**scewY()**) can be specified.

To apply several transformations to the same element, you need to specify them all in one property:\

```
transform: scale(2) scew(15deg) rotate(30deg) translate(100px 50px);
```

## ● Pseudo-classes

Apart from the usual classes that can be created in HTML, there are also **pseudo-classes**. They are created in **CSS** and are used to work with elements when they are in a certain state. For example, to change the styles of the element when the user hovers over it with the cursor.

**:hover** is a **pseudo-class** that allows to change the styles of the element when the user hovers over it with the cursor. When the cursor is no longer over the element, its original styles will be restored

```
a:hover {  
    color: green; /* the text of the link will become green when the user hovers over it*/  
}
```

## ● Delay

**Transition** is a property that allows to change the animation settings, namely, its duration and the delay. Transition is a shorthand property for 4 properties:

**transition-delay** (a delay before the animation starts, specified in seconds (s) or in milliseconds (ms))

**transition-duration** (the duration of transition)

**transition-property** (the property to which the transition will be applied – for example, transform or margin)

**transition-timing-function** (animation smoothness)

All of these values (or some of them) can be specified in one transition property:

**transition: 2s; /\* animation time is 2 seconds \*/**

**transition: 5s 1s transform; /\* the transition will start with 1 second delay and will last 5 seconds\*/**

## ● Animation and keyframes

To create an animation that can be activated in any part of the code, the **@keyframes** keyword can be used (it's usually written at the end of the CSS file), the animation name is added after it.

```
@keyframes pulsar {  
}
```

In curly brackets the so-called animation **key frames** must be specified (at least the first one and the last one). They can be specified as **from** and **to**, but it is more convenient to use percent. The **beginning** of the animation is specified as **0%** and the **end** as **100%**.

## ● Animation and keyframes

The example of animation during which the element will “pulse” - take turn becoming transparent and opaque.

```
@keyframes pulsar {  
    0% {  
        opacity: 1;  
    }  
    100% {  
        opacity: 0;  
    }  
}
```

## ● Other animation properties

**animation-delay** - a delay before the animation starts (in seconds or milliseconds)

**animation-timing-function** - the animation smoothness. Takes the following values:

- **linear** - animates at even speed;
- **easy** - the animation starts slowly, speeds up and then it slows down again;
- **easy-in-out** - the same as easy, but speeds up faster;
- **easy-in** - the animation starts slowly and then speeds up;
- **easy-out** - the animation starts quickly and then slows down;
- **steps** -the animation transitions are abrupt, going step-by-step, the number of steps is specified in parentheses - steps(10, end);

## ● Other animation properties

- **step-start** - applies the final values of the styles right away;
- **step-end** - the styles have their initial values during the whole animation time and then transition to the final values.

## ● Filters

**filter** is a CSS property that applies various graphical effects to the element. Can take the following values (more than one value can be specified):

- **blur** can be specified in % or other units;
- **brightness** can be specified in % or as a decimal number. 0 is the color black, 100% or 1 is the original state;
- **contrast** - 0% - the color black, 100% is the original state;
- **drop-shadow** - the element shadow, takes x-offset and y-offset, blur radius and the color;
- **grayscale** - 0% is the original state, 100% converts all the colors to grayscale;
- **hue-rotate** - takes a hue rotation angle and changes the color;
- **invert** - inverts the colors, takes values from 0% (the original state) to 100% (fully inverted colors);

## ● Filters

- **opacity** - 0% is fully transparent element, 100% is fully opaque element;
- **saturate** - the color saturation 100% is the original state, 0% is a black-and-white image;
- **sepia** - converts the image to sepia, 0% is the original state, 100% - sepia.