06

Responsive Designing

WHAT YOU WILL LEARN

* 2D Transforms
* 3D Transforms
* Transitions
* Animation

2D Transforms

* Transforms allow you to move, rotate, scale, and skew elements.
* With the CSS transform property you can use the following 2D transformation methods:
* The translate() Method - The translate() method moves an element from its current position (according to the parameters given for the X-axis and the Y-axis).
  + - Example - div {  
        transform: translate(50px, 100px);  
       }
* The rotate() Method - The rotate() method rotates an element clockwise or counter-clockwise according to a given degree.
  + - Example - div {  
        transform: rotate(20deg);  
      }
* The scale() Method - The scale() method increases or decreases the size of an element (according to the parameters given for the width and height).
  + - Example - div {  
        transform: scale(2, 3);  
      }
* The skew() Method - The skew() method skews an element along the X and Y-axis by the given angles.
  + - Example - div {  
        transform: skew(20deg, 10deg);  
      }
* The matrix() Method - The matrix() method combines all the 2D transform methods into one.
  + - Example - div {  
        transform: matrix(1, -0.3, 0, 1, 0, 0);  
      }

3D Transforms

* rotateX() - The rotateX() method rotates an element around its X-axis at a given degree:
  + - Example - #myDiv:hover {  
        transform: rotateX(150deg);  
      }
* rotateY() - The rotateY() method rotates an element around its Y-axis at a given degree:
  + - Example - #myDiv:hover {  
        transform: rotateY(150deg);  
      }
* rotateZ() - The rotateZ() method rotates an element around its Z-axis at a given degree:
  + - Example - #myDiv:hover {  
        transform: rotateZ(150deg);  
      }

Transitions

CSS transitions allows you to change property values smoothly, over a given duration.

* Transition - The transition effect will start when the specified CSS property (width) changes value.

Example - div {

width: 100px;

height: 100px;

background: red;

transition: width 2s;

}

div:hover {

width: 300px;

}

* transition-delay - specifies a delay (in seconds) for the transition effect
  + - Example - div {  
        transition-delay: 2s;  
      }
* transition-duration - Specifies how many seconds or milliseconds a transition effect takes to complete
  + - Example - div {  
        transition-duration: 5s;  
      }
* transition-property - Specifies the name of the CSS property the transition effect is for
  + - Example - div {  
        transition-property: width;  
      }  
        
      div:hover {  
        width: 300px;  
      }

Animation

CSS allows animation of HTML elements without using JavaScript or Flash!

* @keyframes
* animation-name
* animation-duration
* animation-delay
* animation-iteration-count
* animation-direction
* animation-timing-function
* animation-fill-mode
* animation