[Lesson 7]

Roi Yehoshua 2018

[DAN.IT EDUCATION]

What we learnt last time?

- How to apply different style to elements upon hovering a mouse
- How to apply specific CSS rules for n-th element with the same class
- How to create blockquote using only CSS.
- How to bind additional content to elements using ::before and ::after

[Our targets for today]

- How to move a block on the page from its position without influencing a flow
- How to place blocks on top of the other blocks
- How to create static block which will remain on the same place
- Different ways of assigning colors in CSS
- Ways of making a color/block transparent

[DAN.IT EDUCATION]

CSS Layout – The position Property

- The position property specifies the type of positioning method used for an element
- There are five different position values:
  - Static (default)
  - relative
  - fixed
  - absolute
  - sticky
- Elements are then positioned using the top, bottom, left, and right properties.
  These positions work differently depending on the position value

Static Position

- HTML elements are positioned static by default
- An element with static position is positioned according to the normal flow of the page
- Static positioned elements are not affected by the top, bottom, left, and right properties

```
div.static {
    position: static;
    border: 3px solid #73AD21;
}
```

position: static;

this div element has position: static;

Relative Position

- An element with position: relative; is positioned relative to its normal position
- Setting the top, right, bottom, and left properties of a relatively-positioned element will cause it to be adjusted away from its normal position
- Other content will not be adjusted to fit into any gap left by the element

```
div.relative {
    position: relative;
    left: 30px;
    border: 3px solid #73AD21;
}
```

position: relative;

this div element has position: relative;

Absolute Position

- An element with position: absolute; is positioned relative to the nearest positioned ancestor (i.e., an element whose position is anything except static)
- If it has not positioned ancestors, it uses the document body

```
div.relative {
        position: relative;
        width: 400px;
        height: 200px;
        border: 3px solid #73AD21;
}

div.absolute {
            position: absolute;
            top: 80px;
            right: 0;
            width: 200px;
            height: 100px;
            border: 3px solid #73AD21;
}
```

position: relative;

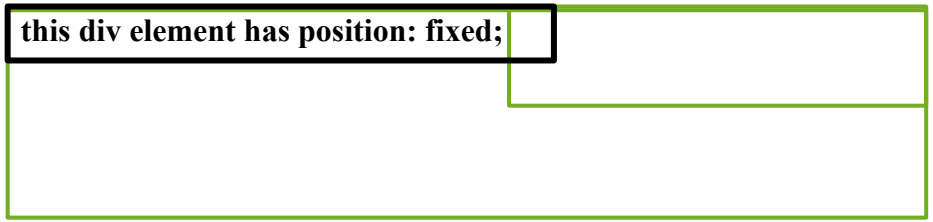
this div element has position: relative;

this div element has position: absolute;

Fixed Position

- An element with position: fixed; is positioned relative to the viewport, which means it always stays in the same place even if the page is scrolled
- A fixed element does not leave a gap in the page where it would normally have been located

```
div.fixed {
    position: fixed;
    top: 0;
    left: 0;
    width: 300px;
    border: 3px solid black;
    background-color: lightblue;
}
```



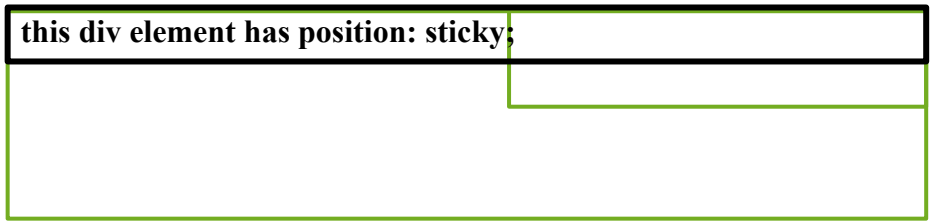
position: fixed;

An element with: position: fixed is positioned relative to the viewport, which means it always stays in the same place even if the page is scrolled

Sticky Position

→ position: sticky; is a hybrid of relative and fixed positioning. The element is treated as relative positioned until it crosses a specified threshold, at which point it is treated as fixed positioned.

```
div.sticky {
    position: sticky;
    top: 20px;
}
```



position: sticky;

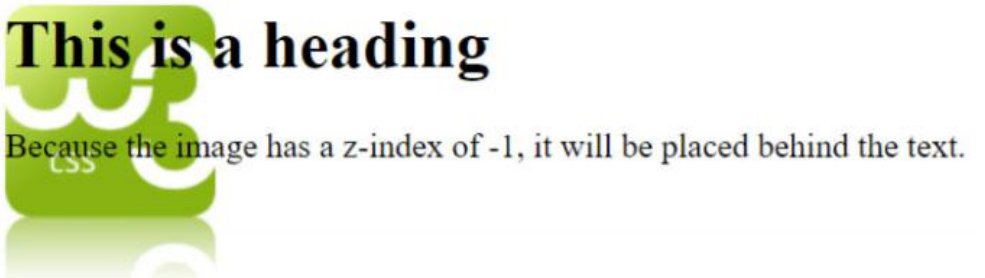
An element with: position: sticky will be placed as a regular block on the page, until the page is scrolled, so this block reaches specified threshold. After that it will be treated as positioned fixed, and would not influence the regular flow any more

Overlapping Elements

- When elements are positioned, they can overlap other elements
- The z-index property specifies the stack order of an element (which element should be placed in front of, or behind, the others)
- An element can have a positive or negative stack order:

```
img {
        position: absolute;
        top: 0;
        left: 0;
        width: 100px;
        z-index: -1;

}
```



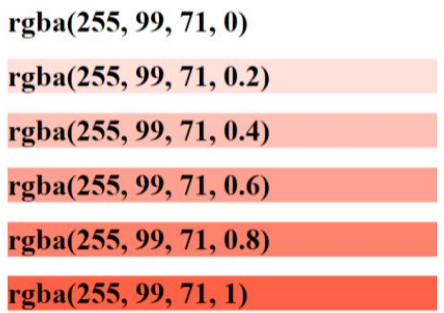
Colors in CSS

- A color can be specified by:
  - a valid color name - like "red" (140 colors)
  - a HEX value - like "#ff0000"
  - an RGB value - like "rgb(255,0,0)"
  - an HSL value - like “hsl(0, 100%, 50%)”

RGBA Colors

- RGBA color values are an extension of RGB color values with an alpha channel, which
- An RGBA color value is specified with: rgba(red, green, blue, alpha)
- The alpha parameter is a number between 0.0 (fully transparent) and 1.0 (not transparent at all)

```
<h1 style="background-color:rgba(255, 99, 71, 0);”>rgba(255, 99, 71, 0)</h1>
<h1 style="background-color:rgba(255, 99, 71, 0.2);”>rgba(255, 99, 71, 0.2)</h1>
<h1 style="background-color:rgba(255, 99, 71, 0.4);”>rgba(255, 99, 71, 0.4)</h1>
<h1 style="background-color:rgba(255, 99, 71, 0.6);”>rgba(255, 99, 71, 0.6)</h1>
<h1 style="background-color:rgba(255, 99, 71, 0.8);”>rgba(255, 99, 71, 0.8)</h1>
<h1 style="background-color:rgba(255, 99, 71, 1);”>rgba(255, 99, 71, 1)</h1>
```



[DAN.IT EDUCATION]

Other ways of adding transparency

- RGBA - rgba(255, 0, 0, 0.5)
- HSLA - hsla(0, 100%, 50%, 0.5)
- HEXA - #ff000080
- “opacity” property:
  - makes all block content transparent
  - a number between 0.0 (fully transparent) and 1.0 (not transparent at all)

[Control questions]

1. What positioning shall you use to create a footer attached to bottom of the page?
2. Will absolutely positioned push away other absolutely positioned elements?
3. What are the two use cases for “position: relative”?
4. Will absolutely positioned be always shown on top of other elements?
5. What CSS property defines which of the layers will be shown on top of the other layers?
6. Which ways of assigning a color in CSS do you know?
7. What are the ways of making block’s background transparent?
8. What is the difference between “opacity” and “background-color” with transparency?

[DAN.IT EDUCATION]

Materials

Core materials:

https://html5book.ru/css-colors/ - How to work with colors in CSS

http://dreamhelg.ru/2011/02/css-position-in-10-steps/

https://html5book.ru/css-position/

https://html5book.ru/css-position/

Additional materials:

http://htmlbook.ru/css/value/color - CSS colors

https://csscolor.ru/hex - CSS colors palette

https://html5book.ru/css-position/

https://developer.mozilla.org/ru/docs/Web/CSS/position

http://shpargalkablog.ru/2017/04/position-sticky.html

Materials

Video materials:
https://www.youtube.com/watch?v=nteG5Rwt-eA

https://www.youtube.com/watch?v=NQVIKd-bnw4