School:		
Date:	**Teacher's name:**	
Grade:	**Number present:**	**absent:**
Topic of the lesson: Floats. Positioning elements.		
Learning objective(s) that this lesson is contributing to	This lesson will introduce you to positioning elements.	
Lesson objectives	**All learners will be able to**: **Most learners will be able to**: • This lesson will introduce you to positioning elements. • **Some learners will be able to:** Write a program	
Assessment Criteria	**Literacy -1point** **Originality -1point** **Time -1 point**	
Value links	"Consider as unhappy that day or that hour in which you have not learned anything new and added nothing to your education" Jan Amos Comenius	
Previous learning	Box model in CSS	
Cross curricular links	Mathematics, logic, informatics	

Time	Planned activities 1. Organizational moment 2. Work with cards 3. A moment of PA	Resources
Beginning 35 min	**Organizational moment** Define the meaning of these words, what they are: CSS / HTML Php SQL Javascript Python Go Java Ruby C ++ FROM Cards with the above words are distributed to children, children in the group determine what it is and where they are used, they can use mobile means or cool computers	
Middle 25 min	To begin with, we will deal with the very concept of positioning. Positioning is the definition of a specific location on the page of an element (box). Positioning is absolute, relative, fixed, and static. I'll hurry with an example, below we will understand what is written in it.	

```
<! DOCTYPE HTML PUBLIC "- // W3C // DTD HTML 4.01 Transitional // EN"
"http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<title> Positioning. </title>

</head>

<body>

<div style = "float: left; background: # c00; border-style: double; padding: 5px;
height: 1500px; width: 300px">

<div style = "position: relative; left: 10px; top: 50px; background: # c0c; border-
style: double; padding: 5px; width: 200px"> The block is positioned relative to the
edges of the parent element. </div>

</div>

<div style = "position: absolute; left: 200px; top: 350px; background: # cc0;
border-style: double; padding: 5px; width: 200px"> The block is absolutely
positioned and pulled out of the general stream, its position is set from the edges
of the browser window. As you can see, this block can overlap with other elements
of the page. </div>

<div style = "position: fixed; left: 150px; top: 150px; background: # 0cc; border-
style: double; padding: 5px; width: 200px"> And this is a fixed block, also
removed from the general stream, however, when scrolling page he does not
change his position. Earlier versions of Internet Explorer ignore this property.
</div>

</body>

</html>
```

watch an example

So, in order to position an element, the position property and one of its possible values are applied to it:

• absolute - Absolute positioning of an element.

• relative - Relative positioning of the element.

• fixed - Fixed positioning of the element.

• static - Static positioning of an element. (The item is displayed as usual.)

• inherit - Inherits the value of the parent element.

Now let's go deeper ..

Absolute positioning.

An absolutely positioned element (position: absolute) is derived from the general stream and, despite other elements and their relative positions, takes the specified place on the page from the edges / edges of the browser window. With this positioning method, one element can overlap on top of another.

In order to position an element from the edges / edges of the browser window, we need the following CSS properties:

• bottom - The distance from the bottom edge of the browser window.

• left - The distance from the left edge of the browser window.

• right - The distance from the right edge of the browser window.

• top - The distance from the top edge of the browser window.

The distance data can be specified in pixels, percent, or any other accepted CSS unit, the default value is auto.

Example:

```
<! DOCTYPE HTML PUBLIC "- // W3C // DTD HTML 4.01 Transitional // EN" "http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<title> Absolute positioning. </title>

</head>

<body>

<div style = "background: # 0f0; border: # 000000 2px solid; padding: 5px; margin: 10px; width: 300px; height: 200px;"> <h1> Block No. 1 </h1> </div>

<div style = "background: # 00f; border: # 000000 2px solid; padding: 5px; margin: 10px; width: 500px; height: 100px;"> <h1> Block No. 2 </h1> </div>

<div style = "position: absolute; left: 200px; top: 100px; background: # f00; border: # 000 2px solid; padding: 5px; width: 200px; height: 200px;"> <h1> Block number 3 < / h1> This block is absolutely positioned! <br> <br> Blocks 1 and 2 do not affect its location in any way. </div>

</body>

</html>
```

watch an example

As you can see, in the example, the third block left the general flow of elements and lives by its own rules, the rest of the page layout does not affect the location of this block.

Relative positioning.

Relative positioning (position: relative) determines the position of the element relative to the edges of the parent element and is not inferred from the general flow.

As in the case with absolute positioning, the distance from the edges / edges of the parent element is set using the properties: bottom, left, right, top.

Example:

```
<! DOCTYPE HTML PUBLIC "- // W3C // DTD HTML 4.01 Transitional // EN"
"http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<title> Relative positioning. </title>

<style type = "text / css">

h1 {

color: # 008000;

font-size: 20px

}

div.blok1 {

background: # c0e4ff;

border: # 000000 2px solid;

padding: 5px;

width: 500px;

height: 400px;

}

div.blok2 {

position: relative;
```

```
left: 150px;

background: # ffa0c5;

border: # 000 2px solid;

padding: 5px; width: 250px;

height: 200px;

}

</style>

</head>

<body>

<div class = "blok1">

<h1> Parent element - block No. 1 </h1>

<div class = "blok2">

<h1> Block No. 2 </h1>

This block is positioned relative to the left edge of the parent element.

</div>

</div>

</body>

</html>
```

watch an example

If the parent element is not explicitly set, then the report is kept from the edge / edges of the browser window.

Fixed positioning.

Fixed positioning (position: fixed) is similar to absolute, the element is derived from the general stream, its coordinates are calculated from the edges of the browser window, but when scrolling the page, the element does not change its position.

Example:

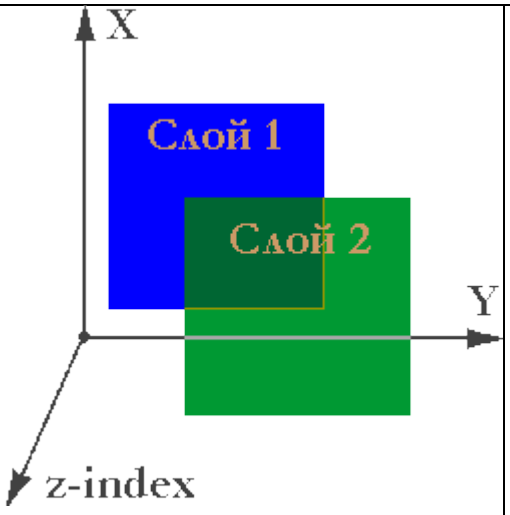
File style.css

```
h1 {
color: # 800;
text-align: center;
font-size: 30px
}
h2 {
color: # 088;
text-align: center;
font-size: 18px
}
div.blok1 {
background-image: url (fon.gif);
border: # 000 2px solid;
padding: 5px;
width: 800px;
height: 4000px
}
div.blok2 {
position: fixed;
left: 250px;
top: 300px;
border: # 080 6px double;
padding: 5px;
width: 300px
}
```

Index.html file

```
<! DOCTYPE HTML PUBLIC "- // W3C // DTD HTML 4.01 Transitional // EN"
"http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<title> Fixed positioning. </title>

<link rel = "stylesheet" href = "style.css" type = "text / css">

</head>

<body>

<div class = "blok1">

<h1> What a page </h1>

</div>

<div class = "blok2">

<h2> Fixed block with intrusive ads </h2>

<h1> Buy an elephant !! </h1>

A fixed block, derived from the general flow of elements, when scrolling the page,
it does not change its position. Earlier versions of Internet Explorer ignore this
property.

</div>

</body>

</html>
```

watch an example



P.S. Earlier versions of Internet Explorer ignore this property and the element is displayed on the page as if it were not positioned at all.

z-index

As already mentioned, the positioned elements can be superimposed one on top of the other, thereby simulating a certain three-dimensionality of the page, where each subsequent element superimposed on top of each other is a layer.

The z-index property allows the webmaster to control the positions of these layers in the depth of the screen (along the Z axis), in other words, allows the browser to indicate which elements should be displayed in the foreground and which in the background.

Values of the z-index property:

• auto — Elements overlap in the order that they appear in the HTML code. (default).

• integer - The higher this value, the higher the position of the element in relation to those elements whose value is lower.

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<title>z-index</title>

</head>

<body>

<div align="center" style="position: absolute; z-index:5; width: 350px; height:
100px; top: 120px; left: 0px; color: #0000ff; font-size:100px">z-index</div>
```

```
<div style="position: absolute; z-index:3; width: 150px; height: 150px; top: 0px; left: 100px; background-color: #ff00ff"> </div>

<div style="position: absolute; z-index:4; width: 150px; height: 150px; top: 100px; left: 0px; background-color: #ff0000"> </div>

<div style="position: absolute; z-index:2; width: 150px; height: 150px; top: 100px; left: 200px; background-color: #ffff00"> </div>

<div style="position: absolute; z-index:1; width: 150px; height: 150px; top: 200px; left: 100px; background-color: #00ff00"> </div>

</body>

</html>
```

watch an example

The numerical value of z-index may be negative, but not all browsers correctly interpret negative values.

It should also be noted that with the z-index value equal, in the foreground is the element that goes lower than the rest in the HTML code.

The same rule applies when z-index is equal to auto, or if this property had not been applied to certain elements at all.

Well, what's the point, you ask, generally use z-index if you can just arrange the elements in the HTML code in the right order?

Let me show you an example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<title>z-index</title>

<style type="text/css">

body {background-color: #c5ffa0}

a {

position: absolute;

z-index: auto;

top: 100px;
```

	border: #000000 1px solid;
	}
	a:hover {
	position: absolute;
	z-index:1;
	top: 80px;
	border: #800000 1px solid;
	}
	</style>
	</head>
	<body>
	 Проведите курсором по картам
	<img src="karta1.gif">
	<img src="karta2.gif">
	<img src="karta3.gif">
	<img src="karta4.gif">
	<img src="karta5.gif">
	<img src="karta6.gif">
	 (a href="#")
	 CSS.
	</body>
	</html>
	
	watch the example
	
	As you can see, the z-index property is irreplaceable where there is a certain dynamic.	

End **5 min**	Homework: watch the video at https://www.youtube.com/watch?v=KmTK8kub_gw 1 hour 37 min	

Differentiation – how do you plan to give more support? How do you plan to challenge the more able learners?	**Assessment – how are you planning to check learners' learning?**	**Health and Safety**
Watch videos and learn more strengthen knowledge about JS	Working in a group each participant is rated the highest score 3 And at the end of the lesson, a self-assessment card is issued	A moment of physical activity in the middle of the lesson