



ПРЕДСТАВЯНЕ И ВИДИМОСТ С CSS



Модул 2: CSS

СЪДЪРЖАНИЕ

Представяне на елемент.....	3
Свойство display.....	3
Видимост на елемент.....	7
Свойство visibility.....	7
Подреждане на елементи.....	10
Свойство z-index.....	10

Когато завършите тази лекция:

- *ще се запознаете вариантите за показване на елементите;*
- *ще се запознаете начините за видимост на елементите;*
- *ще се запознаете със свойството за подреждане на елементите.*

Представяне на елемент

Свойство `display`

Свойство `display` определя как да се представи даден елемент (фиг.1).

Стойност: `none` | `inline` | `block` | `inline-block` | `table` | `flex` | `grid`

Стойност по подразбиране: `inline`

none - премахва елемента и той не се представя

inline - представя елемента на една линия с останалите

block - представя елемента както блок и той заема цялата ширина на родителския

inline-block - представя елемента като блок на една линия с останалите

table - представя елемента като таблица

flex - представя елемента като flex контейнер

grid - представя елемента като grid контейнер

ПРИМЕР: Представяне на елементите (фиг.1)

HTML

`<p>` Пример за свойство **display** със `` стойност **none** ``. Елементът изчезва. `</p>`

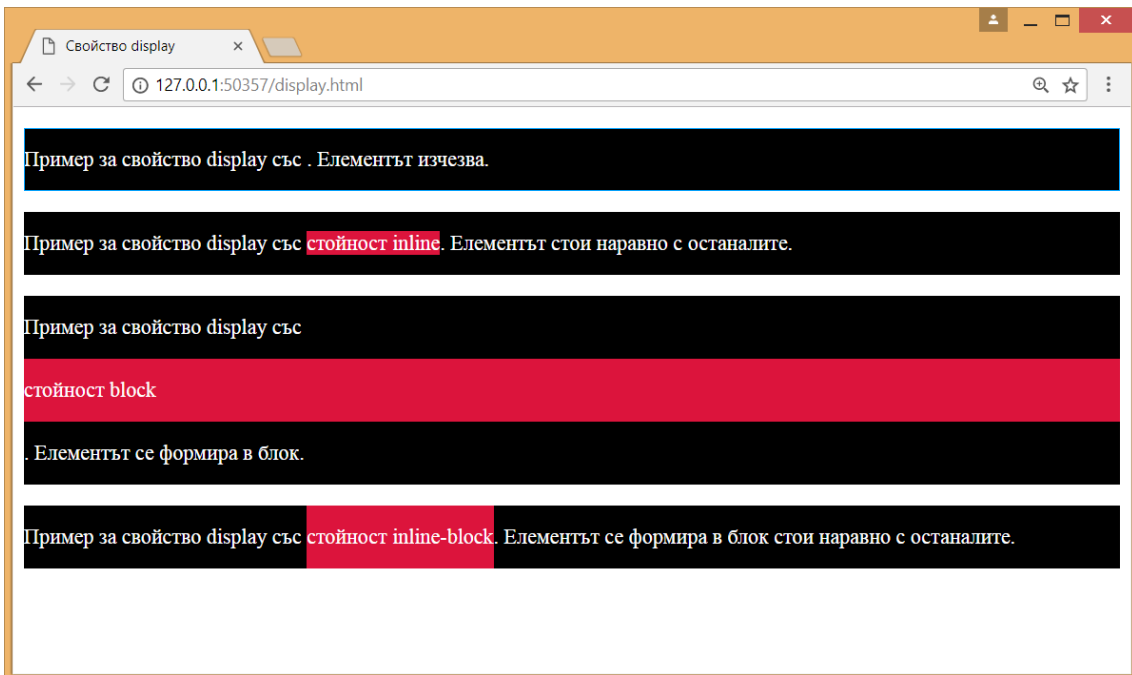
`<p>` Пример за свойство **display** със `` стойност **inline** ``. Елементът стои наравно с останалите. `</p>`

`<p>` Пример за свойство **display** със `` стойност **block** ``. Елементът се формира в блок. `</p>`

`<p>` Пример за свойство **display** със `` стойност **inline-block** ``. Елементът се формира в блок стои наравно с останалите. `</p>`

CSS

```
p{
  background-color: black;
  color: white;
  line-height: 3em;
}
span{ background-color: crimson; }
span#none{ display: none; }
span#inline{ display: inline; }
span#block{ display: block; }
span#inlineblock{ display: inline-block; }
```



Фиг. 1

Видимост на елемент

Свойство `visibility`

Свойство `visibility` определя дали елемента да се вижда.

Стойност: `visible` | `hidden` | `inherit`

Стойност по подразбиране: `inherit`

Задаването на стойност `hidden` на свойство `visibility` се различава от стойността `none` при свойство `display` по това, че скритите (`hidden`) елементи продължават да заемат пространството си в структурата на уеб страницата.

ПРИМЕР: Видимост на елементите (фиг.2)

HTML

```
<p> Пример за свойство visibility със <span id="visible"> стойност visible </span>. Елементът е видим. </p>
```


`<p>` Пример за свойство `visibility` със `` стойност `hidden` ``. Елементът не е видим. `</p>`

`<p>` Пример за свойство `visibility` със `` стойност `none` ``. Елементът изчезва. `</p>`

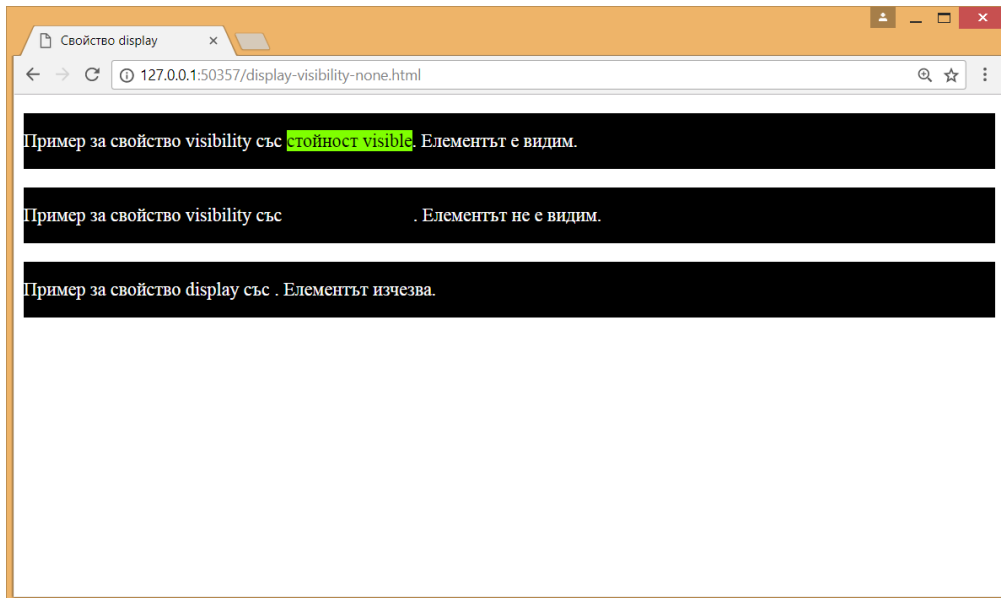
CSS

```
p{
  background-color: black;
  color: white;
  line-height: 3em;
}
span{
  background-color: chartreuse;
  color: black;
}
```

```
span#visible{ visibility: visible; }
```

```
span#hidden{ visibility: hidden; }
```

```
span#none{ display: none; }
```



Фиг. 2

Подреждане на елементи

Свойство z-index

Свойство **z-index** определя подредбата на елементи насложени един върху друг по оста z.

Стойност: **auto** | положително или отрицателно цяло число | **inherit**

Стойност по подразбиране: **auto**

ПРИМЕР: Подреждане на елементите по ос z (фиг.3)

HTML

```
<div class="behind"> <p> z-index: -1 </p> </div>
```

```
<div class="level0"> <p> z-index: 0 </p> </div>
```

```
<div class="level1"> <p> z-index: 1 </p> </div>
```

```
<div class="level2"> <p> z-index: 2 </p> </div>
```

CSS

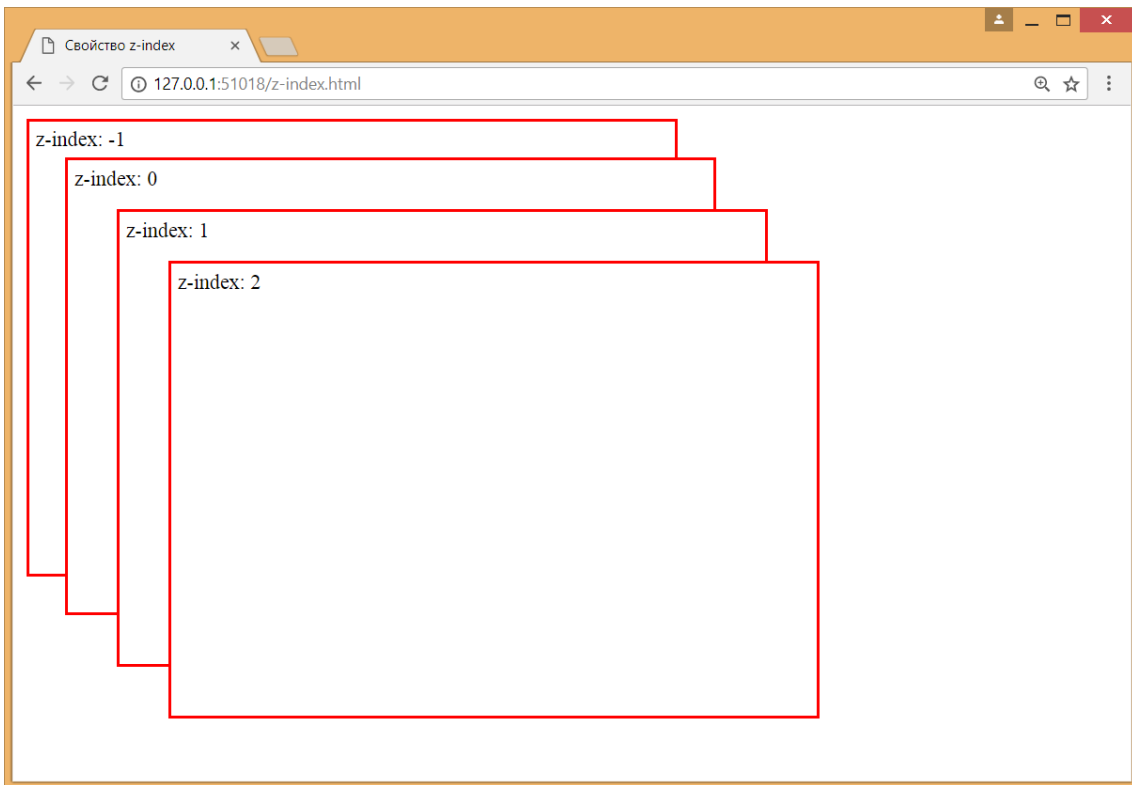
```
div{  
    position: absolute;  
    width: 500px;  
    height: 350px;  
    background-color: white;  
    border: 2px solid red;  
}  
div.behind{  
    top: 10px;  
    left: 10px;  
    z-index: -1;  
}  
div.level10{  
    top: 40px;
```

```
    left: 40px;
    z-index: 0;
}


div.level1{
    top: 80px;
    left: 80px;
    z-index: 1;
}


div.level2{
    top: 120px;
    left: 120px;
    z-index: 2;
}


```



Фиг.3