



Уеб разработка

CSS Layout

или как да управляваме размера и разположението на HTML елементите

Съдържание

- Width & height
- display
- Margin & Padding
- CSS Box Model
- Position

```
.ninja  
{  
  color: #000;  
  visibility: hidden;  
}
```



width
height

width & height

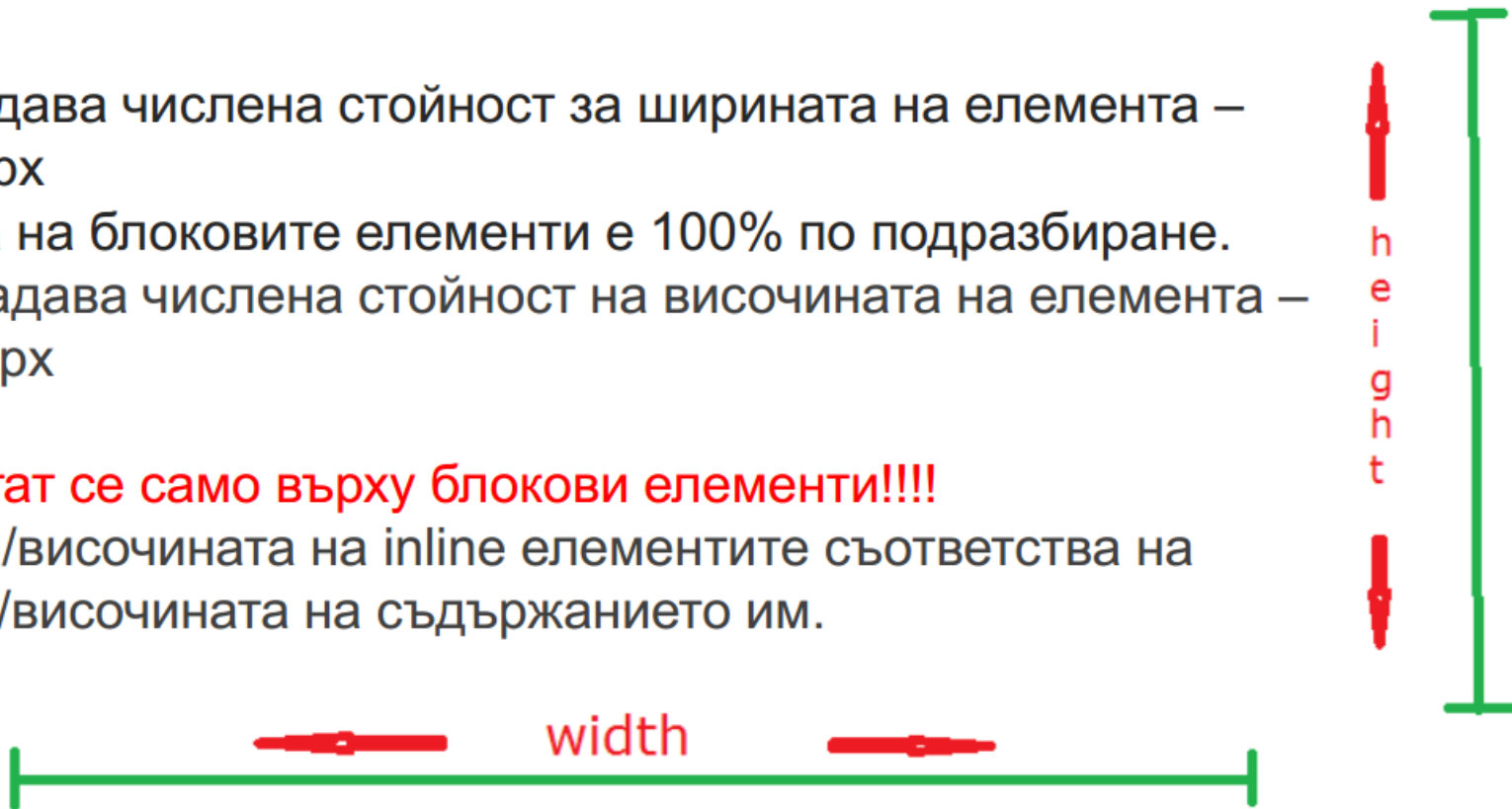
width – задава числена стойност за ширината на елемента –
напр. 200px

Ширината на блоковите елементи е 100% по подразбиране.

height – задава числена стойност на височината на елемента –
Напр. 250px

!!!!Прилагат се само върху блокови елементи!!!!

Ширината/височината на inline елементите съответства на ширината/височината на съдържанието им.



display

```
.fear {  
  display: none;  
}
```

display

Как да превърнем блоковите в inline елементи и обратно ...

display: block – елементът застава на нов ред, както

и този след него и може да променяме размерите му

incididunt ut labore et dolore magna aliqua. Ut
enim ad minim veniam, quis nostrud
exercitation ullamco laboris nisi ut aliquip ex
Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed do eiusmod tempor
incididunt ut labore et dolore magna aliqua. Ut
enim ad minim veniam, quis nostrud
exercitation ullamco laboris nisi ut aliquip ex
ea commodo consequat. Duis aute irure dolor
in reprehenderit in voluptate velit esse cillum
dolore eu fugiat nulla pariatur. Excepteur sint
occaecat cupidatat non proident, sunt in culpa
qui officia deserunt mollit anim id est
laborum. ea commodo consequat. Duis aute
irure dolor in reprehenderit in voluptate velit
esse cillum dolore eu fugiat nulla pariatur.
Excepteur sint occaecat cupidatat non
proident, sunt in culpa qui officia deserunt



```
div {  
  width: 300px;  
  height: 450px;  
  background-color: #501DD1;  
  color: #D5F738  
}  
span {  
  background-color: #FB9504;  
  color: #125EF3;  
  display: block;  
  width: 200px;  
  height: 200px;  
}
```



adipiscing elit, sed do eiusmod tempor
incididunt ut labore et dolore magna aliqua. Ut
enim ad minim veniam, quis nostrud
exercitation ullamco laboris
nisi ut aliquip ex

Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed do eiusmod tempor
incididunt ut labore et dolore magna aliqua. Ut
enim ad minim veniam, quis nostrud
exercitation ullamco laboris nisi ut aliquip ex
ea commodo consequat. Duis aute irure dolor
in reprehenderit in voluptate velit esse cillum
dolore eu fugiat nulla pariatur. Excepteur sint
occaecat cupidatat non proident, sunt in culpa
qui officia deserunt mollit anim id est
laborum.
ea commodo consequat. Duis
aute irure dolor

display

display: inline – елементът застава на един и същи ред с този преди него /ако е inline-елемент/

Напр. превръщане на li от блокови в inline елементи.

- [Link 1](#)
 - [Link 2](#)
 - [Link 3](#)
 - [Link 4](#)
 - [Link 5](#)
- преди и след стилизиране
- [Link 1](#) [Link 2](#) [Link 3](#) [Link 4](#) [Link 5](#)

```
li {  
  display: inline;  
  background-color: #C375B2;  
}
```

! Няма да може да контролираме височината и широчината на елементите!

display: inline-block – подреждат се като inline елементи, но може да управляваме размерите им.

margin
padding



margin & padding

- дефинират разстоянието около елемента, за който са зададени

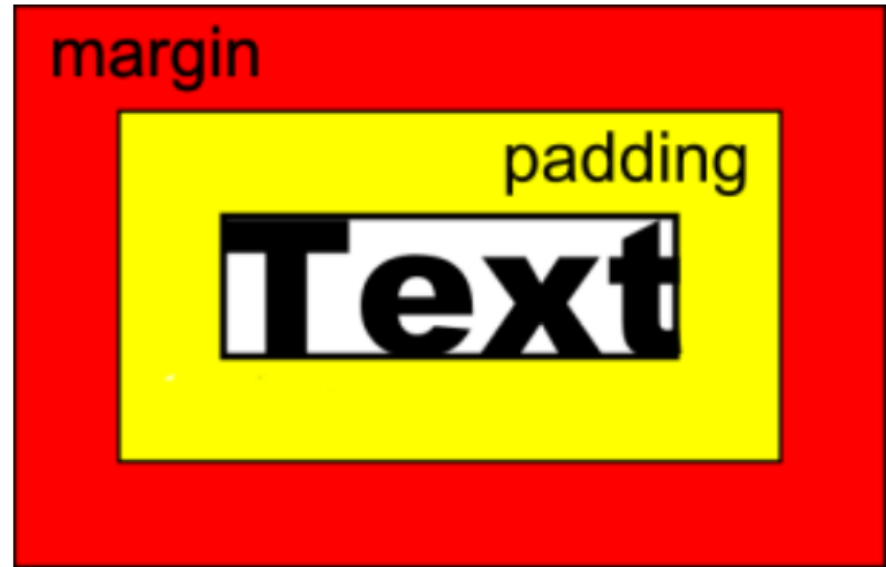
- Ще ги задаваме с px – 10px, 30px ...

- Разлика

- padding - при задаване на стойност >0 се появява

разстояние между съдържанието и външната граница на елемента/**!демо!**

- margin – при задаване на стойност >0 се появява разстояние между външната граница на елемента и съседните му елементи.



margin & padding

- Могат да бъдат задавани /margin аналогично и padding/

1 . за всяка една от страните на елемента поотделно –

`margin: 1px 3px 5px 7px;` по посока на часовниковата стрелка – горе, дясно, долу, ляво

Или, когато трябва да посочим само някое от четирите разстояния -

`margin-top:1px; margin-right:3px;margin-bottom:5px; margin-left:7px;`

2. две по две - `margin: 5px 10px` –

Горната и долната по 5px, дясната и лявата по 10px

3. всички заедно - `margin: 5px`

margin & padding - особености

- <https://hacks.mozilla.org/2015/03/understanding-inline-box-model/>
- ! Когато вертикалните margin-и на два елемента се допират, като резултат ще се появи само по-голямата стойност. Няма да се сумират!

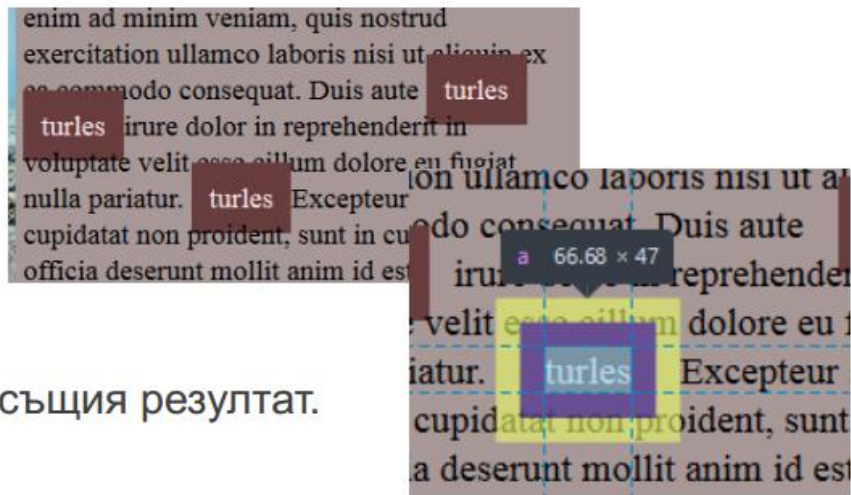
- inline елементите и padding

! padding вляво и вдясно избутва

текста, но отгоре и отдолу се

припокрива със съдържанието

на съседните редове. margin дава същия резултат.



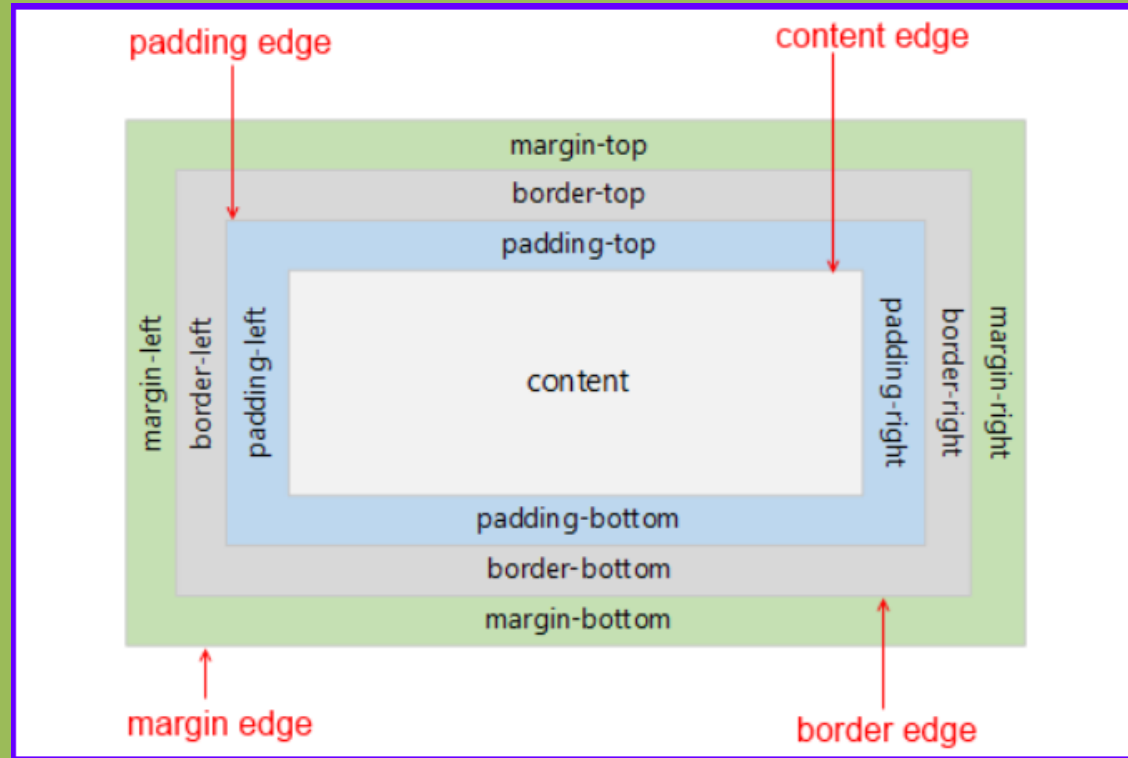
margin & padding

- ! Нулираме зададените по подразбиране margin&padding на

ВСИЧКИ, използвани от нас елементи в кода!

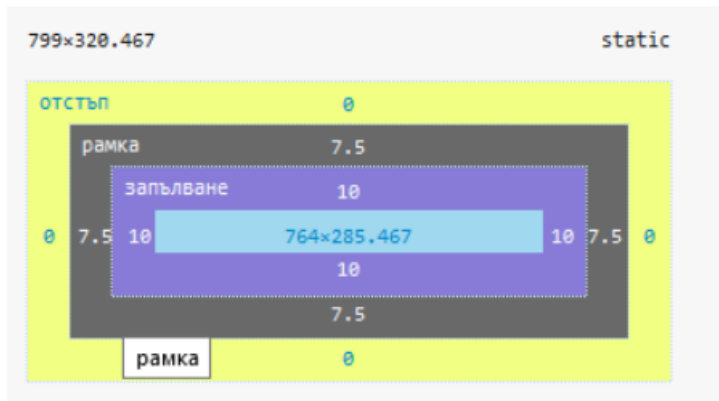
```
html, body, div, ul, li, a, img, p {  
    padding: 0;  
    margin: 0;  
}
```

box-model



box-model

- При измерването на елементите трябва да имаме предвид, че големината на елемента е сума
- Ширина – right-border+left-border+right-padding+left-padding
- Височина - top-border+bottom-border+top-padding+bottom-padding



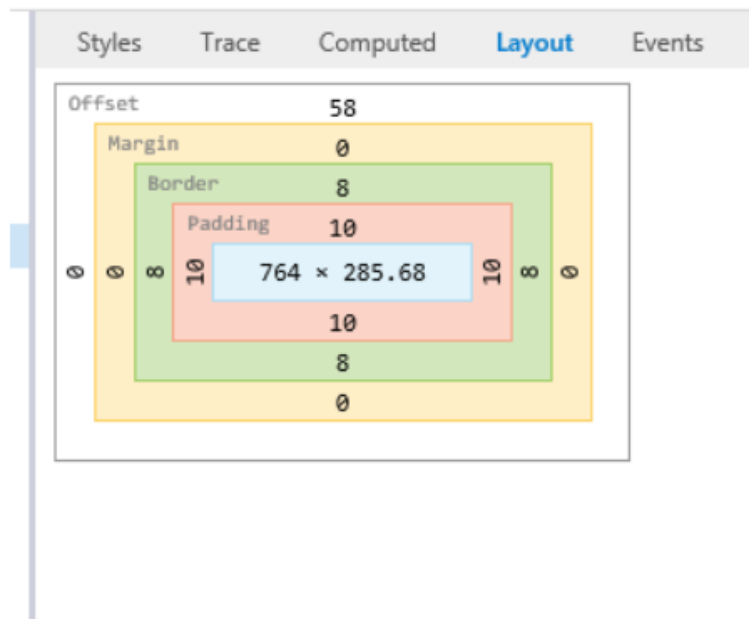
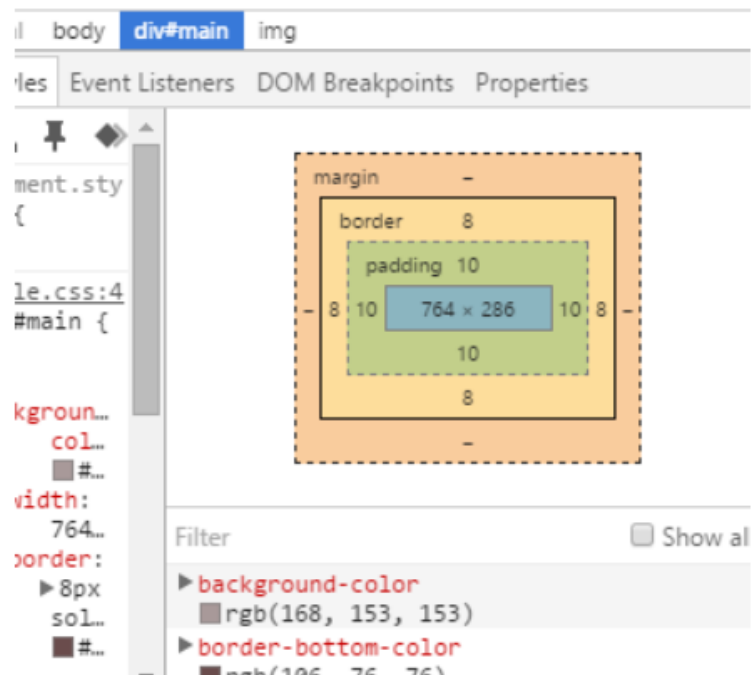
```
div#main {  
  background-color: #A89999;  
  width: 764px;  
  border: 8px solid #6A4C4C;  
  padding: 10px;  
}
```

– не разчитайте с точност до 1px размерите, за да не

се чупят страниците!

box-model

Същия код в Chrome и IE



position

position

`position` – определя позицията на елемента в поредицата от HTML елементи, разположени на страницата

Приема една от следните стойности

- `static` (по подразбиране)
- `relative` – в сравнение с това къде елемента би се появил при позиция `static`
- `absolute` – спрямо първия родителски елемент, имащ позиция различна от `static`
- `fixed` – спрямо прозореца на браузъра, но игнорира скролирането.

position

top, left, bottom, right:

конкретизираме отстоянието на
absolute/fixed/relative

позиционирани елементи като
число

fixed positioned div

static position

absolute positioned div

relative position

bare in mind the padding when calculating the
offset!!!!

```
#static {  
  background-color: #FC9E9E;  
  width: 300px;  
  height: 200px;  
  padding: 30px;  
}  
  
#relative {  
  position: relative;  
  background-color: #8C4B4B;  
  width: 300px;  
  height: 200px;  
  padding: 30px;  
  left: 360px;  
  top: -260px;  
}  
  
#relative p {  
  color: #FFF605;  
}  
  
#abs {  
  position: absolute;  
  top: 0;  
  left: 0;  
  background-color: #98F407;  
}  
  
#fixed {  
  position: fixed;  
  top: 0;  
  left: 0;  
  background-color: #0FF77C;  
}
```