



Recapitulare Web

Cum ar trebui gandit un website

Organizare

- Cream un folder cu denumirea website-ului;
- In el vom avea un fisier `index.html` de care legam un fisier `main.css`;
- Vom crea un folder `Images` in care vom adauga imaginile folosite in website (cand le adaugam in html folosim `Images/denumire.jpg`);

Codul de baza in HTML

<!DOCTYPE html>

<head>

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

Legatura cu CSS

<meta charset="UTF-8">

Tipul de character encoding folosit.

<meta name="viewport" content="width=device-width, initial-scale=1.0 ">

Sa nu aiba zoom cand intra pe website si pe mobil sa nu il scaleze proportional.

<title>Titlul Paginii</title>

Titlul paginii ce apare in Tab-ul de la browser

</head>

<body>

</body>

</html>

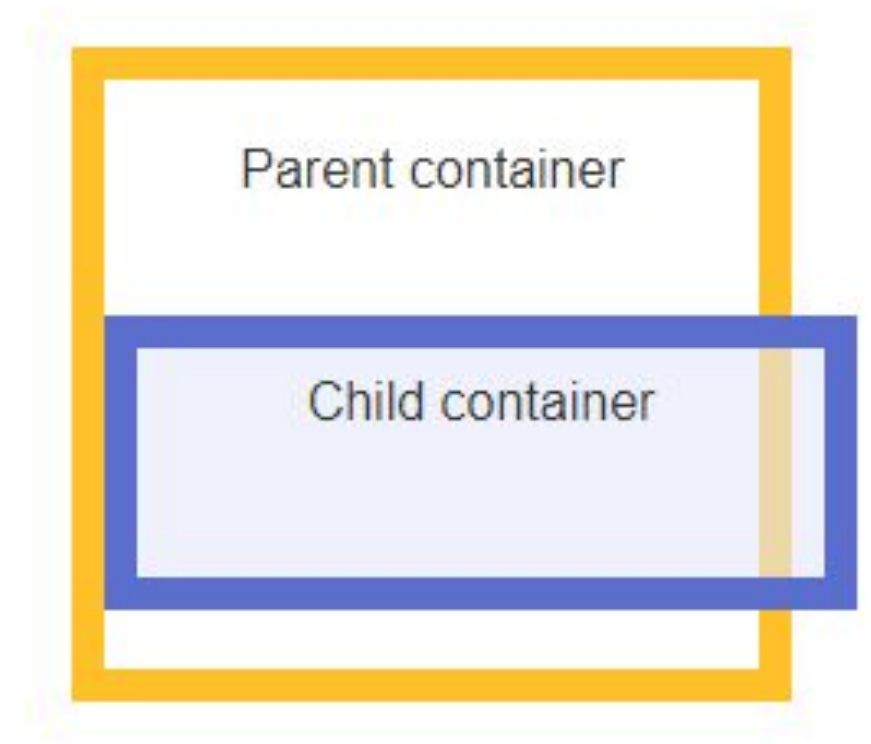
Codul de baza in CSS

```
*{  
margin: 0;  
padding: 0;  
box-sizing: border-box;  
}
```

Box sizing pentru a nu avea div-uri ce ies din alte div-uri dupa ce primesc padding;

Margin si Padding pentru a ne asigura ca nu ne apar spatii nedorite intre anumite elemente;

Ne asiguram ca niciun browser nu ne va adauga border.

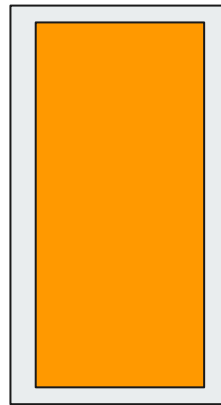


Codul de baza in CSS

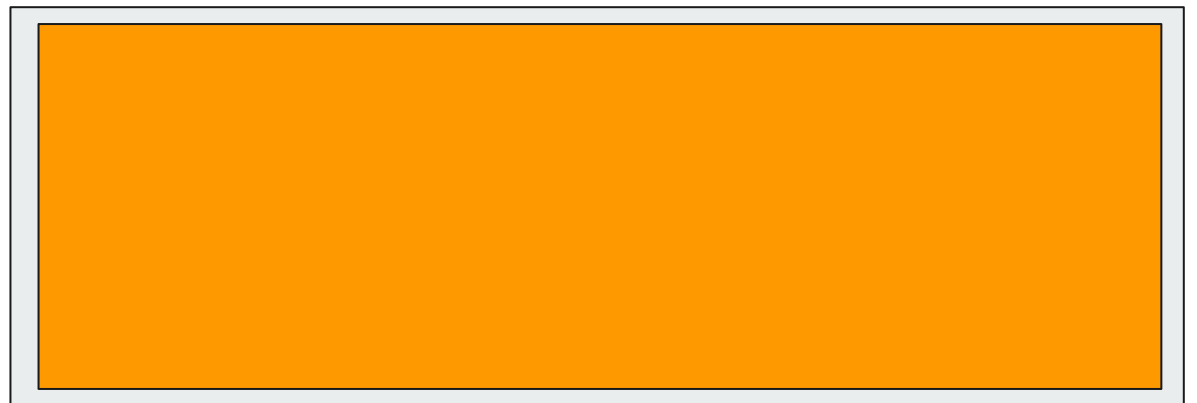
```
html, body{  
width:100%;  
height:100%;  
}
```

Pentru a ne permite sa marim cat vrem ceea ce introducem in interior.

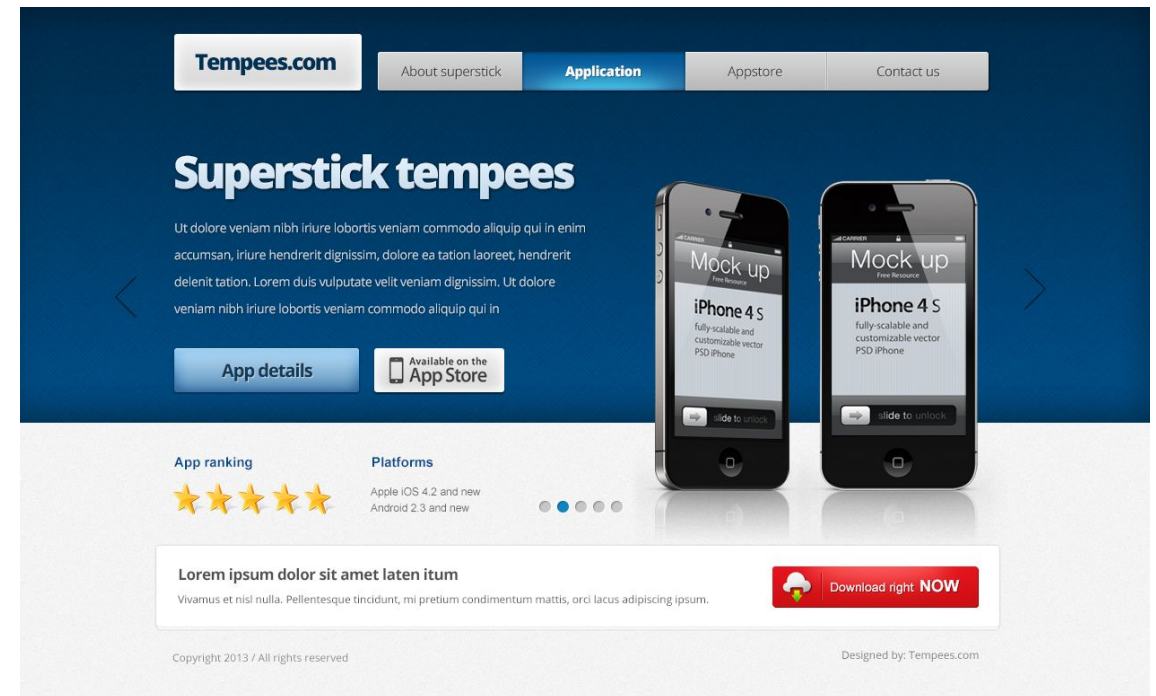
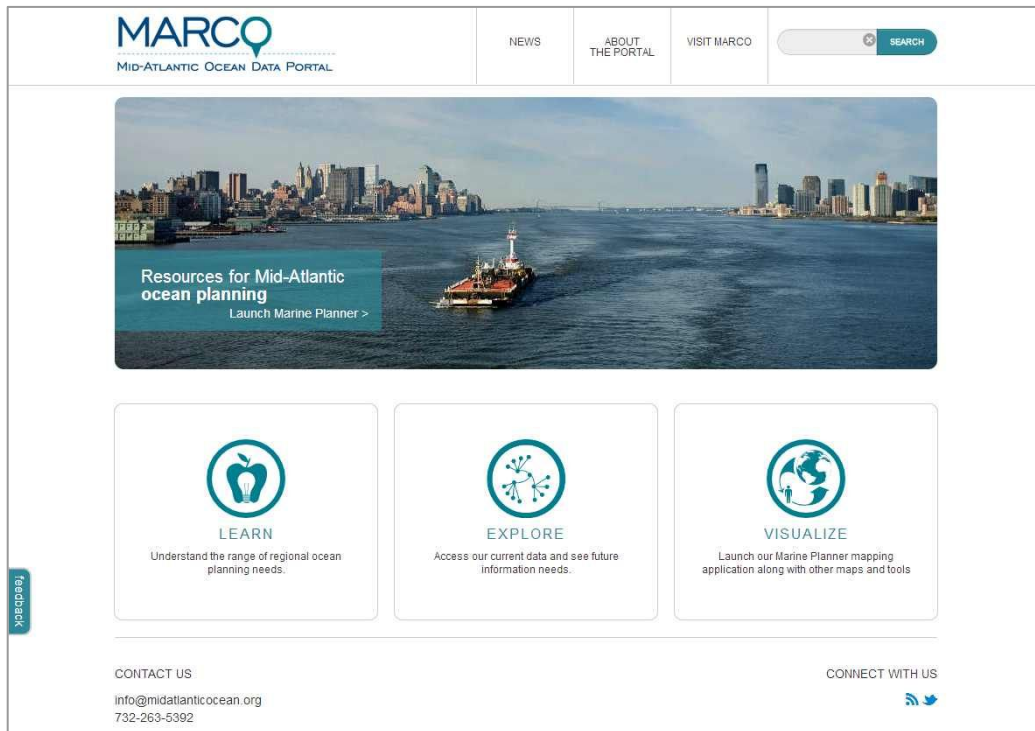
(in general cand vrem ca un obiect sa se intinda pe intreaga dimensiune a device-ului are nevoie ca parintele lui sa faca asta mai intai);



Daca parintele nu are width, atunci width-ul lui este dat de ceea ce contine el (daca nu contine nimic nu exista si nu il vom vedea, chiar daca ii aplicam un background-color).



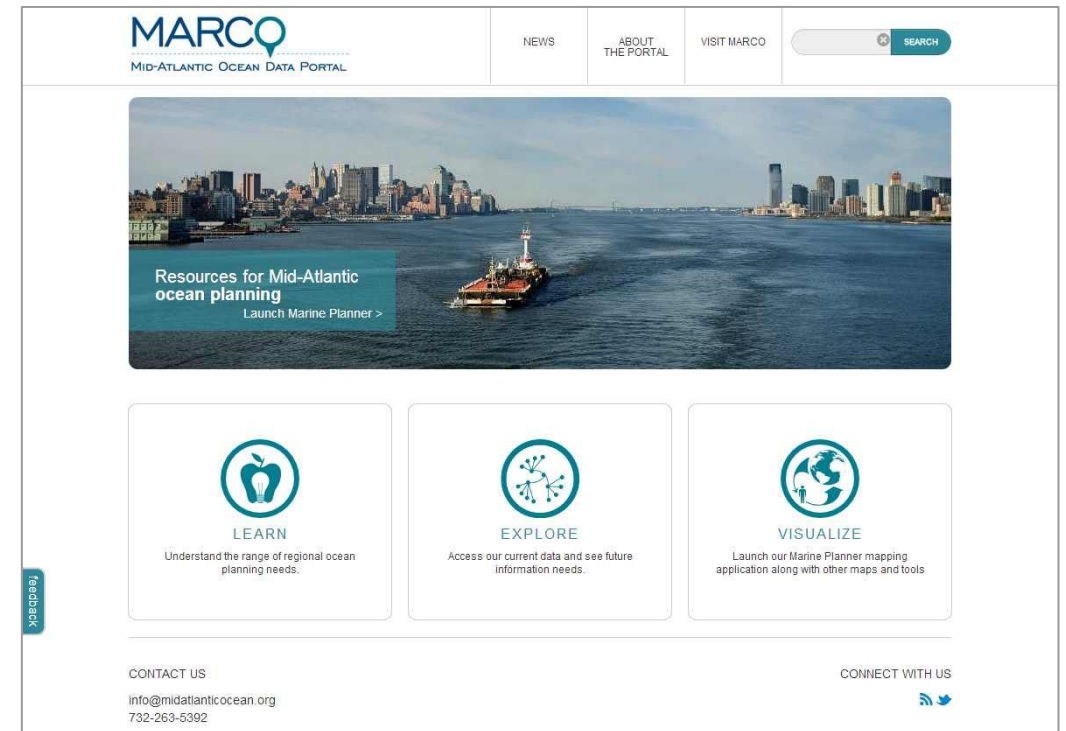
Avem background pe html sau pe sectiuni ?



Daca am background pe html

Daca am o singura culoare/imagine in background atunci voi pune pe html acea culoare/imagine (background-size: cover, acopera intreaga suprafata scaland proportional imaginea).

```
html, body{  
    width: 100%;  
    height: 100%;  
    background-color: #e9e9e9;  
    background-image: url(bg-general.jpg);  
    background: center no-repeat fixed;  
    background-size: cover;  
}
```



Daca am background pe sectiuni

O sa impartim website-ul nostru in cate sectiuni avem (div-uri) carora le vom da cate o clasa. In CSS setam width 100% si culoarea/imaginea acelei sectiuni (height-ul si-l va lua in functie de ce contine acea sectiune).

```
html, body {width: 100%; height: 100%;}
```

```
.sectiune1{
```

```
width: 100%;
```

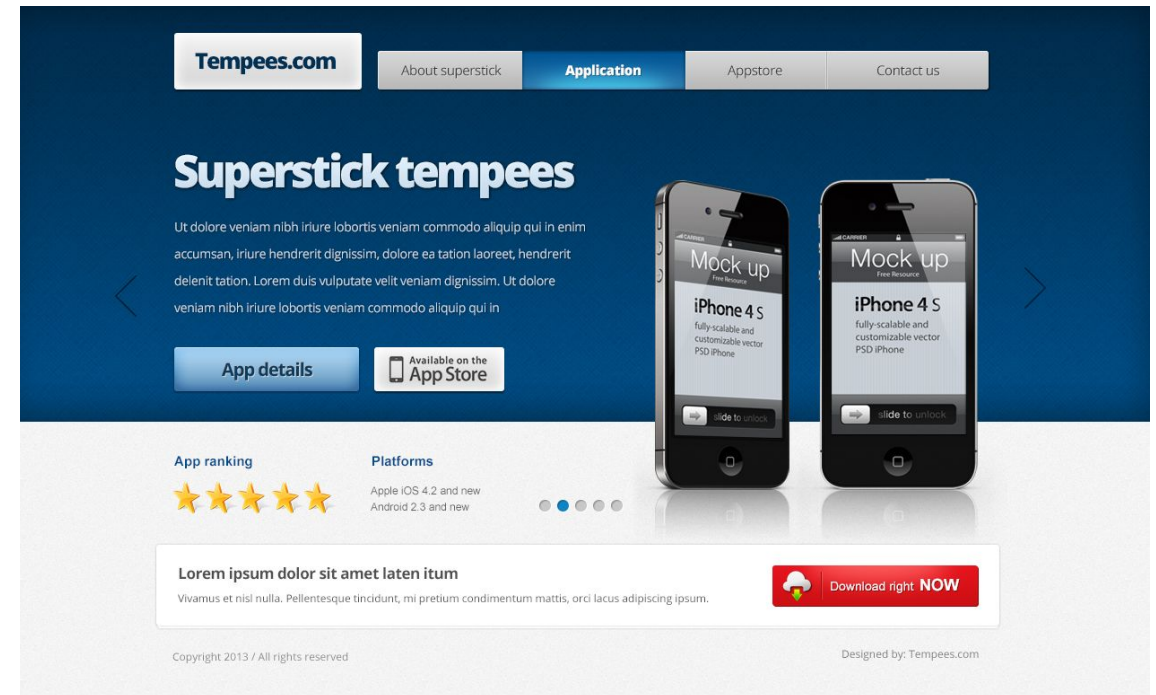
```
background-color: red;
```

```
background-image: url(bg-general.jpg);
```

```
background: center no-repeat fixed;
```

```
background-size: cover;
```

```
}
```



Cum ar arata html-ul nostru?

<body>

<div class="sectiune1">

content

</div>

<div class="sectiune2">

content

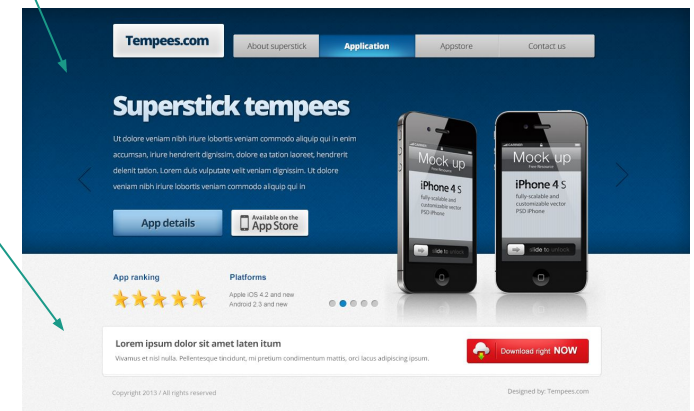
</div>

.....

</body>

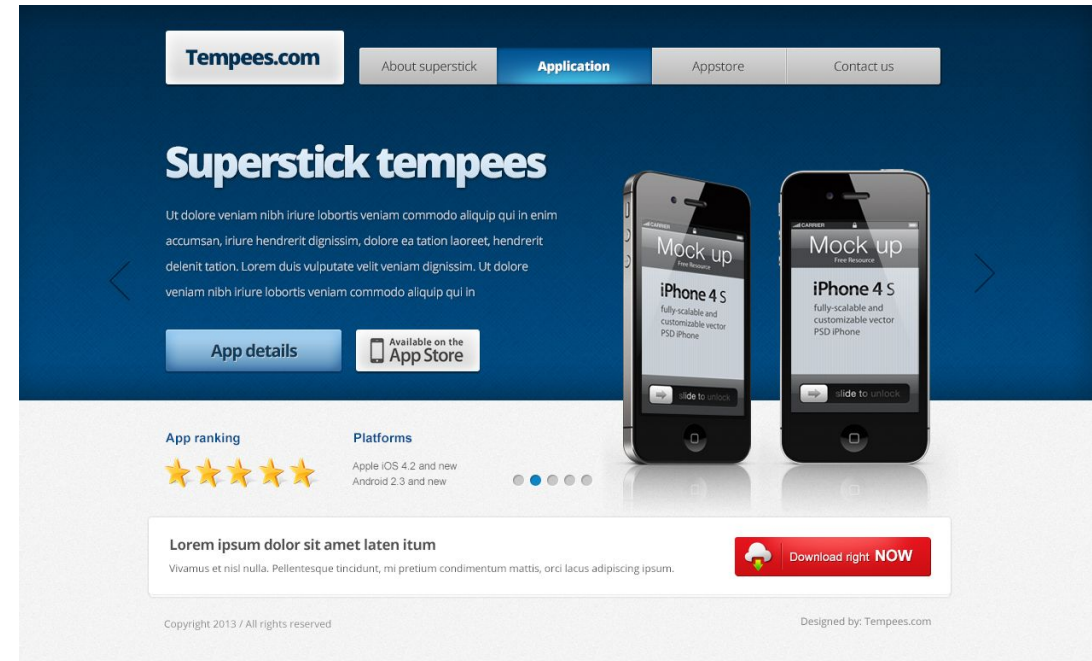
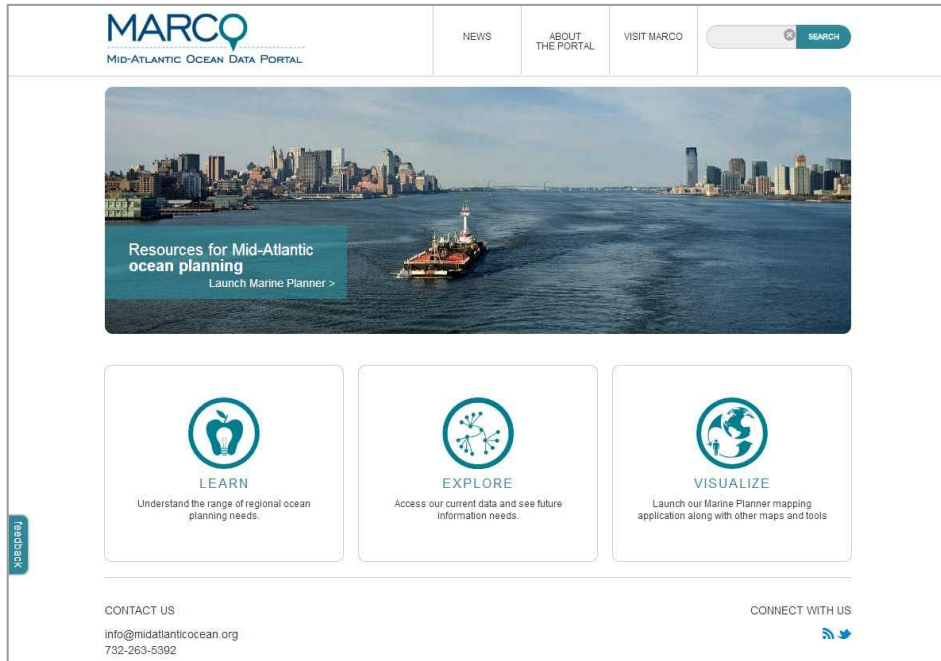
Background prima
sectiune

Background a
doua sectiune



Pe ce dimensiune facem website-ul ?

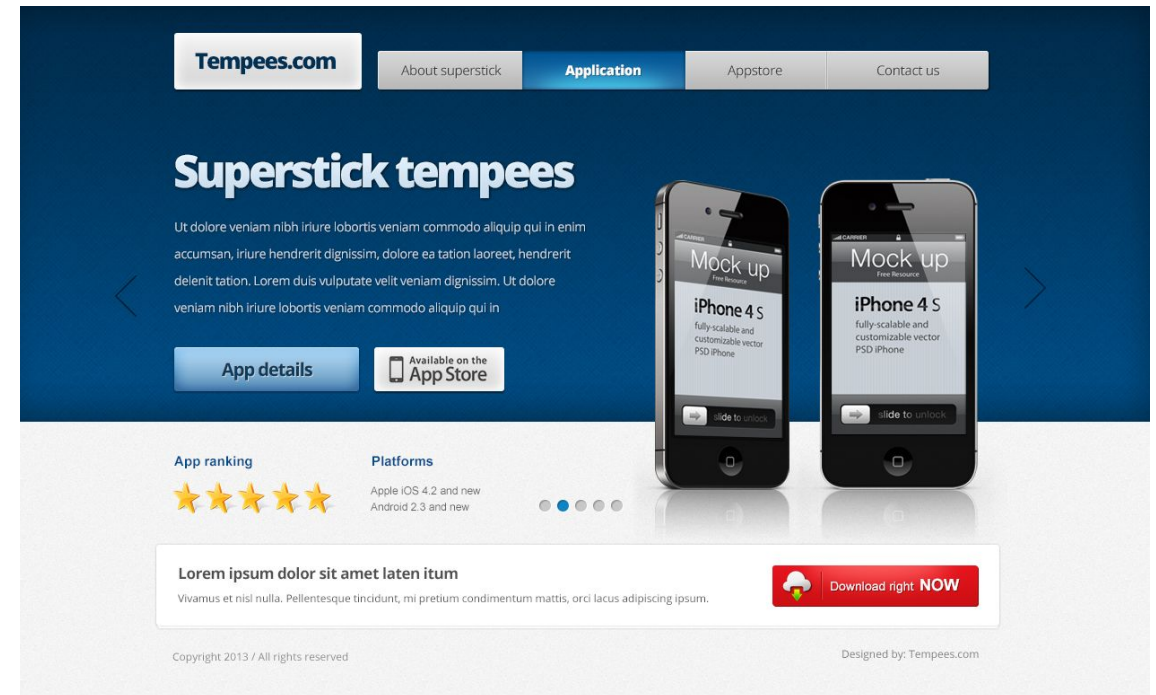
Am hotarat ca background-ul o sa se intinda pe intreaga latime a device-ului;
Apoi avem website-ul propriu zis unde in majoritate a cazurilor este 1024px
(care ar fi latimea minima a unui rezolutii in zilele noastre).



Website-ul propriu zis

Continutul nostru va fi in div-uri in interiorul fiecarei sectiuni pentru a putea aplica pe ele o clasa ce are setata dimensiunea si pozitia centrata. Ex.:

```
.container {  
display: table;  
margin: 0 auto;  
width: 1024px;  
}
```



Cum ar arata html-ul nostru?

```
<body>
```

```
  <div class="sectiune1">
```

```
    <div class="container">
```

```
      content
```

```
    </div>
```

```
  </div>
```

```
  .....
```

```
</body>
```



Cum aranjam continutul website-ului ?

1. Display
2. Float
3. Position

Display

Cele mai importante tipuri de display sunt urmatoarele:

- **Display: inline** - apare pe acelasi rand cu obiectul precedent, nu putem aplica width si height, daca ii dam padding si background-color ne apare dar celelalte elemente nu tin cont de el ci de dimensiunea continutului acelui element;
- **Display: block** - apare pe randul urmator, putem aplica width si height, tine cont daca dam padding unui element, by default fara nicio dimensiune preia intreaga dimensiune html-ului sau a containerului in care se afla;
- **Display: inline-block** - se comporta ca un inline, apare pe acelasi rand insa putem folosi width, height si/sau padding cu background-color;
- **Display: table** - putem sa ii dam width si height, putem sa il centram in pagina cu margin: 0 auto, by default daca nu are dimensiune, background-ul se va pune doar pe dimensiunea continutului sau;

Float

Incepem sa folosim float atunci cand vrem sa folosim 2 sau mai multe coloane in designul nostru (ceva in stanga sau dreapta a ceva);

- **None** - care este si default, il folosim cand avem un website pe o singura coloana ca layout tot continutul website-ului vine unul sub altul;
- **Left** - il folosim pe un obiect daca vrem ca urmatorul/urmatoarele obiecte sa vina in dreapta lui ;
- **Right** - il folosim pe un obiect daca vrem ca urmatorul/urmatoarele obiecte sa vina in stanga lui ;

Atentie! Cand folosim float pe ceva afecteaza toate obiectele ce ii urmeaza, unde vrem sa oprim float-ul ii dam proprietatea **clear:left, right sau both** ;

Position

Folosim position atunci cand vrem sa mutam un obiect in scena noastra intr-o pozitie nenaturala, care sa nu influenteze celelalte obiecte din website (cand ne dam seama ca este imposibil de pozitionat cu display si/sau float);

1. **Position:static** - pozitia default cu care lucram in cazul in care nu ii precizam nimic;
2. **Position:relative** - nu schimba pozitia initiala a obiectului insa dupa ce ii dam position relative putem sa il mutam folosind attributele: top, bottom, left, right;
3. **Position: absolute** - scoate un element din website, nu mai afecteaza site-ul in niciun fel si nu mai este afectat de celelalte obiecte, este pozitionat fata de html sau fata de un container din care face parte care are position:relative;
4. **Position:fixed** - scoate un element din website si il pune deasupra celorlalte cu o pozitie fixa indiferent in ce loc ne-am uita in pagina);

Practice

Clear:left
Float:left

Float:left

Float Left

Display: Table

Display: inline-block

Position:relative
top:100px

Superstick tempees

Ut dolore veniam nibh iriure lobortis veniam commodo aliquip qui in enim accumsan, iriure hendrerit dignissim, dolore ea tation laoreet, hendrerit delenit tation. Lorem dui vulputate velit veniam dignissim. Ut dolore veniam nibh iriure lobortis veniam commodo aliquip qui in

App details

Available on the App Store

App ranking

★★★★★

Platforms

Apple iOS 4.2 and new
Android 2.3 and new

Download right NOW

Copyright 2013 / All rights reserved

Designed by: Tempees.com

Superstick tempees

Ut dolore veniam nibh iriure lobortis veniam commodo aliquip qui in enim accumsan, iriure hendrerit dignissim, dolore ea tation laoreet, hendrerit delenit tation. Lorem dui vulputate velit veniam dignissim. Ut dolore veniam nibh iriure lobortis veniam commodo aliquip qui in

[App details](#)

Available on the
App Store



App ranking



Platforms

Apple iOS 4.2 and new
Android 2.3 and new

Lorem ipsum dolor sit amet laten itum

Vivamus et nisl nulla. Pellentesque tincidunt, mi pretium condimentum mattis, orci lacus adipiscing ipsum.

[Download right NOW](#)