

Οριζόντια Διάταξη Στοιχείων - Δημιουργία Στηλών

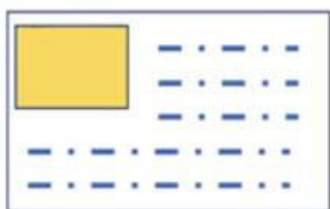
Θα δούμε τώρα τη δυνατότητα που έχουμε να διατάξουμε το περιεχόμενο μιας ιστοσελίδας με χρήση της ιδιότητας `float`. Υπάρχουν και άλλοι εναλλακτικοί τρόποι όπως είναι το `flex` και το `grid` που αντικαθιστούν τη `float`, η `float` όμως έχει κάποιες ενδιαφέρουσες ιδιότητες και είναι σχετικά απλή στη χρήση.

Η `float` μπορεί να πάρει την τιμή **none** που δεν έχει καμία επίδραση στη ροή, ή μία εκ των **left** και **right** που καθορίζουν τη θέση του στοιχείου στα αριστερά ή στα δεξιά ενός υποδοχέα (ενός κουτιού). Εάν σε διαδοχικά στοιχεία δώσουμε την ιδιότητα **float: left** τότε αυτά θα τοποθετηθούν το ένα δίπλα στο άλλο και όχι το ένα κάτω από το άλλο που είναι η προκαθορισμένη συμπεριφορά των κουτιών.

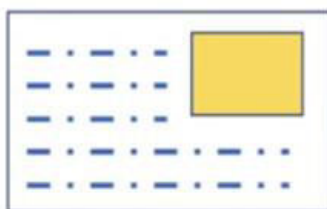
Εάν έχουμε έναν υποδοχέα και μέσα σε αυτόν έχουμε ένα στοιχείο (π.χ. μια εικόνα) που έχει την ιδιότητα **float: left**, το στοιχείο αυτό θα μπει στην αριστερή πλευρά του υποδοχέα και το κείμενο ή τα άλλα στοιχεία τα `inline elements` θα ακολουθήσουν γύρω από αυτό το στοιχείο. Κάτι που βλέπουμε συνήθως σε ένα άρθρο για παράδειγμα μιας εφημερίδας που η εικόνα είναι στη γωνία και το κείμενο ρέει γύρω από την εικόνα.

float: left ή right το κείμενο και τα υπόλοιπα `inline` στοιχεία περιστοιχίζουν το στοιχείο στο οποίο έχει επιβληθεί η ιδιότητα.

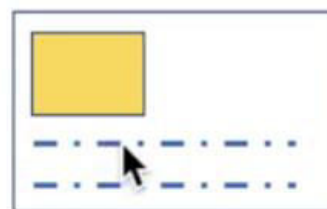
float: none το κείμενο ακολουθεί.



float: left

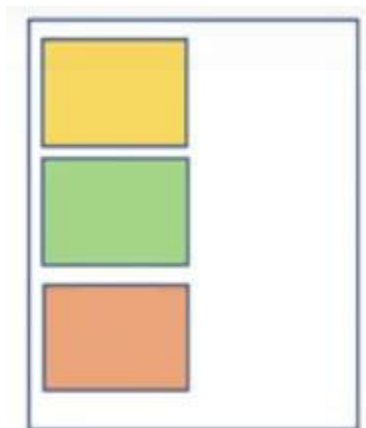


float: right



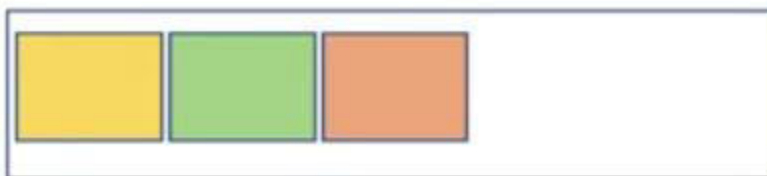
float: none

Χωρίς την ιδιότητα float τα στοιχεία τύπου block στοιβάζονται το ένα κάτω από το άλλο ακόμη και αν το μέγεθός τους επιτρέπει διάταξη δίπλα-δίπλα.



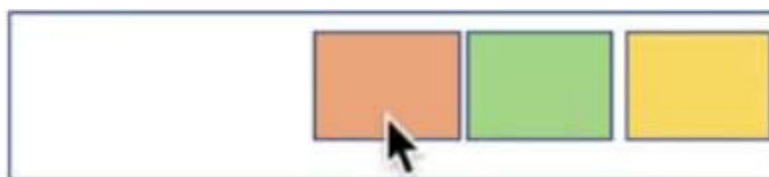
Η ιδιότητα float μας επιτρέπει να επιτύχουμε οριζόντια διάταξη των στοιχείων. Έτσι εάν έχω τα τρία στοιχεία που φαίνονται πιο πάνω τα οποία χωράνε να μπούνε το ένα δίπλα στο άλλο, τότε βάζοντας και στα τρία την ιδιότητα **float:left** τα στοιχεία μπαίνουν το ένα δίπλα στο άλλο ακολουθώντας την διάταξη που φαίνεται πιο κάτω:

float: left

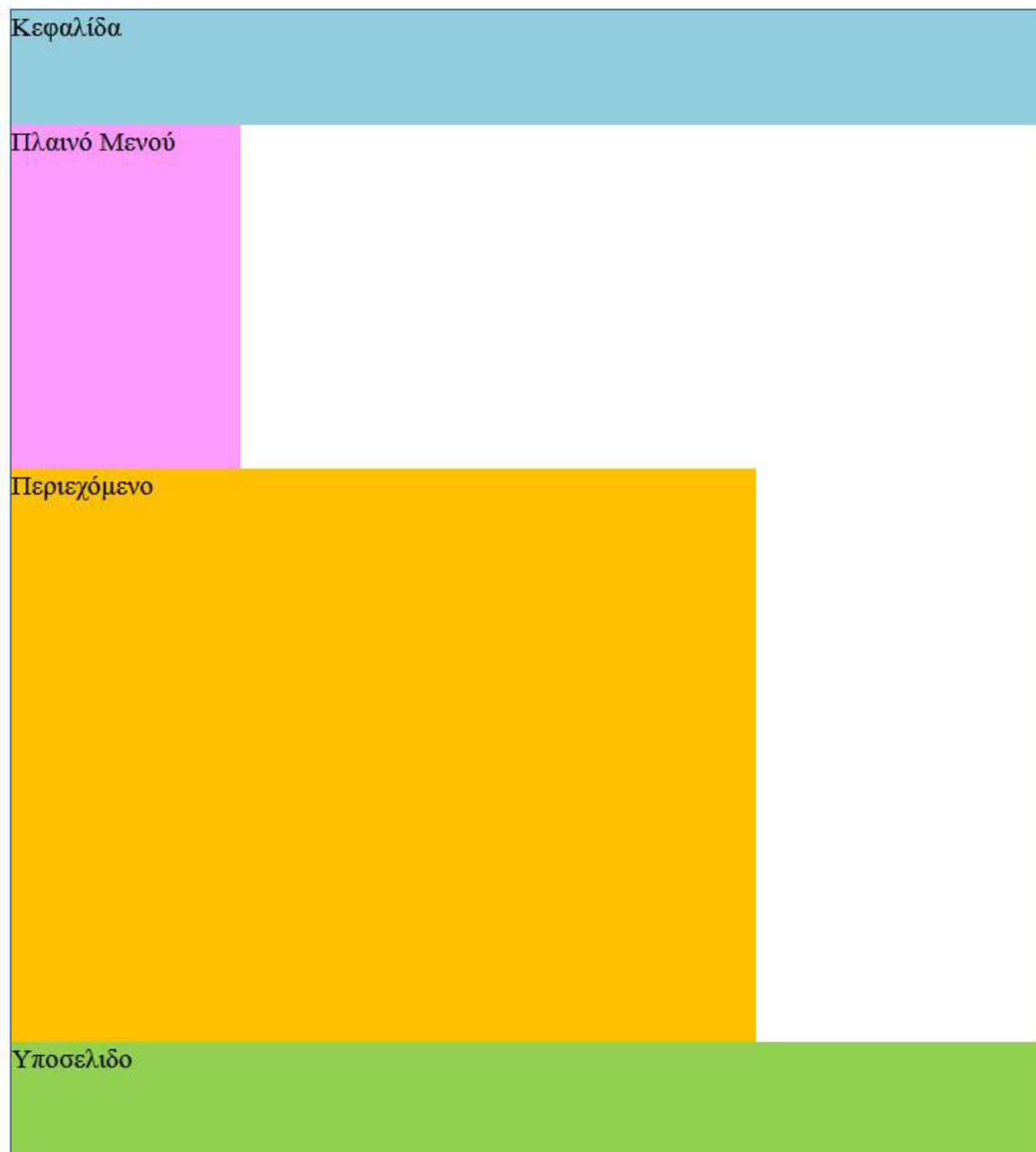


Αντίθετα αν στα ίδια στοιχείαβάλω με τη σειρά στο κίτρινο, το πράσινο και το πορτοκαλί την ιδιότητα **float:right** τότε βλέπω ότι το κίτρινο θα πάει στη δεξιά άκρη, το πράσινο δίπλα του και το πορτοκαλί δίπλα. Έτσι τα βλέπω στοιβαγμένα στη δεξιά πλευρά του υποδοχέα.

float: right



Ας δούμε ένα παράδειγμα. Έστω τα πιο κάτω τέσσερα στοιχεία **κεφαλίδα**, **πλαϊνό μενού**, **περιεχόμενο** και **υποσέλιδο** και έστω ότι αποτελούν τα βασικά κουτιά διάταξης μιας σελίδας.



Ο κώδικας της διάταξης μαζί με το σχετικό stylesheet φαίνονται πιο κάτω:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Floats</title>
    <style>
      .container{
        width: 900px;
        margin: 0 auto;
        border: 2px solid #385D8A;
        font-size: 1.5em;
      }

      header{
        height: 100px;
        background-color: #93CDDD;
      }

      nav {
        height: 300px;
        width: 200px;
        background-color: #FF99FF;
      }

      article {
        height: 500px;
        width: 650px;
        background-color: #FFC000;
      }

      footer{
        height: 100px;
        background-color: #92D050;
      }
    </style>
  </head>
  <body>
```

```

<div class="container">
  <header>Κεφαλίδα</header>
  <nav>Πλαϊνό Μενού</nav>
  <article>Περιεχόμενο</article>
  <footer>Υποσέλιδο</footer>
</div>
</body>
</html>

```

Προσέξτε ότι έχω περιορίσει το πλάτος των κουτιών **nav** & **article** μέσω της ιδιότητας **width**. Αν αυτή **απουσίαζε** το **πλάτος των κουτιών** θα ήταν στο **100% του πλάτους του υποδοχέα**. Προσέξτε επίσης ότι τα κουτιά ανεξάρτητα από το μέγεθός τους στοιβάζονται το ένα κάτω από το άλλο με την σειρά που έχουν τοποθετηθεί μέσα στον κώδικα οι σχετικές ετικέτες που τα δημιουργούν.

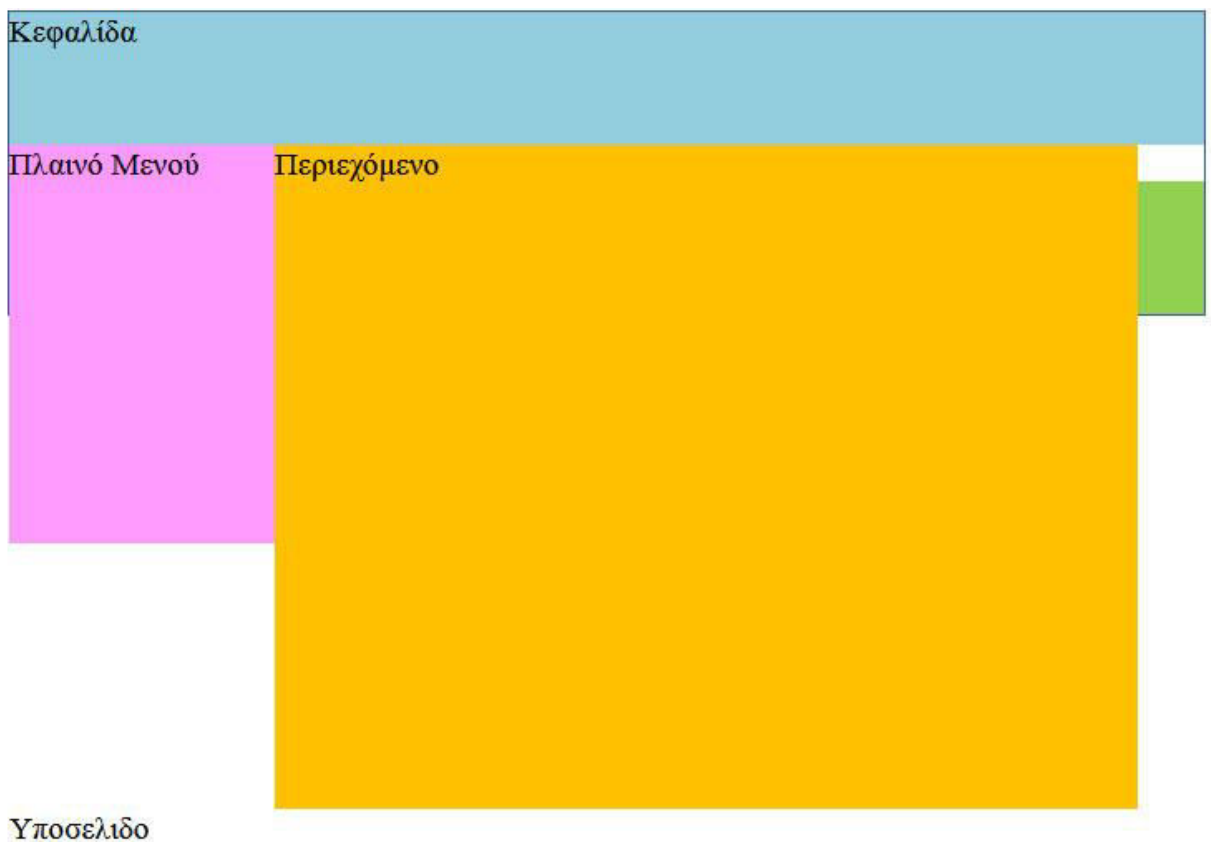
Έστω ότι θέλουμε να φέρουμε τα μεσαία κουτιά (το πλαϊνό μενού και το περιεχόμενο) το ένα δίπλα στο άλλο και να έχουμε την πιο κάτω εμφάνιση .



Θα πρέπει αρχικά και στα δύο στοιχεία να δώσουμε την ιδιότητα **float: left**.

```
nav {  
    height: 300px;  
    width: 200px;  
    float: left;  
    background-color: #FF99FF;  
}  
  
article {  
    height: 500px;  
    width: 650px;  
    float: left;  
    background-color: #FFC000;  
}
```

Η χρήση όμως της float στα κουτιά έχει την πιο κάτω παρενέργεια:



Τα στοιχεία που βρίσκονται κάτω από το τελευταίο κουτί στο οποίο επιβάλλεται η `float` δεν ακολουθούν την κανονική ροή. Για να τα επαναφέρουμε ακολουθούμε τα πιο κάτω βήματα:

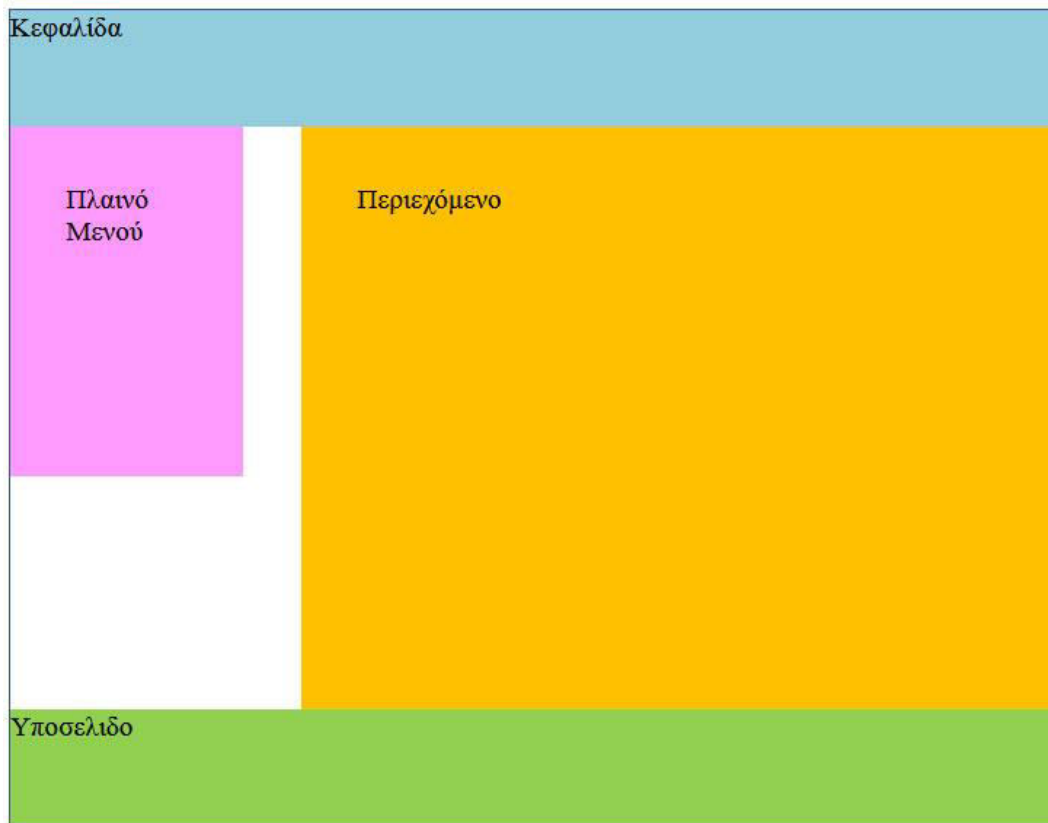
α) Μέσα στον html κώδικα και ακριβώς μετά το τελευταίο στοιχείο στο οποίο θα επιβάλλουμε την ιδιότητα `float` τοποθετούμε ένα αόρατο στοιχείο όπως είναι η ετικέτα `<br>` και της αποδίδουμε ένα όνομα μέσω της παραμέτρου `class` - συνήθως το όνομα `clearfloat`.

```
<body>
  <div class="container">
    <header>Κεφαλίδα</header>
    <nav>Πλαϊνό Μενού</nav>
    <article>Περιεχόμενο</article>
    <br class="clearfloat">
    <footer>Υποσέλιδο</footer>
  </div>
</body>
```

β) Στη συνέχεια συνδυάζω τον πιο κάτω κανόνα μορφοποίησης με το στοιχείο `clearfloat`.

```
.clearfloat {
  clear: both;
}
```

Αν θέλω τώρα να στήσω την πρώτη στήλη αριστερά και τη δεύτερη δεξιά όπως φαίνεται πιο κάτω:



τότε στο stylesheet θα δώσω στην αριστερή στήλη **float: left** και στην δεξιά στήλη **float: right**.

```
nav {  
  height: 300px;  
  width: 200px;  
  float: left;  
  background-color: #FF99FF;  
}  
  
article {  
  height: 500px;  
  width: 650px;  
  float: right;  
  background-color: #FFC000;  
}
```

Η προσθήκη του **<br class="clearfloat">** και του σχετικού κανόνα θα γίνουν όπως πριν.

ΠΡΟΣΟΧΗ! Καλό είναι να ορίζω τα πλάτη των στηλών με την βοήθεια ποσοστών. Με αυτό τον τρόπο καταφέρνω να δημιουργήσω **προσαρμοστικές στήλες**. Έτσι αν θέλω και οι δύο μαζί να καταλαμβάνουν όλο το πλάτος του υποδοχέα στον οποίο περιέχονται τότε δίνω π.χ. 30% για την πρώτη και 70% για τη δεύτερη. Το άθροισμά τους πρέπει να είναι το 100%.

Σ την περίπτωση που όλες οι στήλες μαζί καταλαμβάνουν το 100% του πλάτους του υποδοχέα τότε είναι καλό να συνδυάζω με αυτές και την ιδιότητα **box-sizing** με τιμή **border-box**, ώστε αν θελήσω να προσθέσω κάποια paddings προκειμένου το περιεχόμενο να μην κολλάει στα άκρα των κουτιών αυτό να γίνει χωρίς να επηρεαστεί το συνολικό πλάτος της στήλης (δεδομένου ότι το πλάτος width ενός κουτιού αναφέρεται στο πλάτος του περιεχομένου του - του κειμένου - και τυχόν αποστάσεις που δίνονται μέσω των paddings προστίθενται στο συνολικό πλάτος του κουτιού αυξάνοντάς το). Κάνοντας λοιπόν χρήση αυτής της ιδιότητας το πλάτος της στήλης παραμένει σταθερό και προσαρμόζεται το πλάτος του περιεχομένου προκειμένου να συμπεριληφθούν τα paddings.

Πιο κάτω φαίνονται οι προσθήκες που πρέπει να γίνουν στο css:

```
.container{
    max-width: 900px;
    margin: 0 auto;
    border: 2px solid #385D8A;
    font-size: 1.5em;
}

header{
    height: 100px;
    background-color: #93CDDD;
}

nav, article{
    box-sizing: border-box;
    padding: 2em;
}

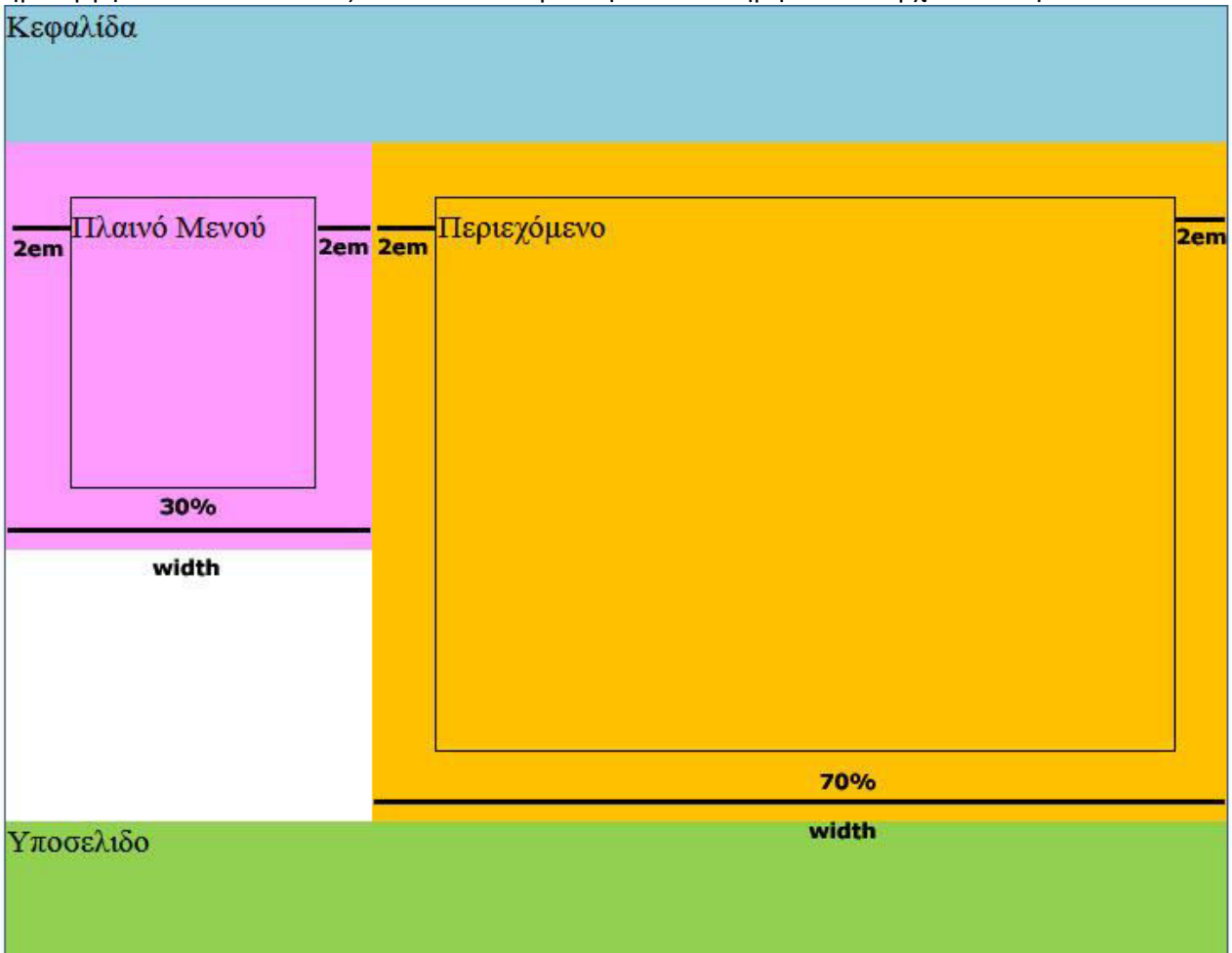
nav {
    height: 300px;
    width: 30%;
    float: left;
    background-color: #FF99FF;
}
```

```

article {
  height: 500px;
  width: 70%;
  float: left;
  background-color: #FFC000;
}
footer {
  height: 100px;
  background-color: #92D050;
}
.clearfloat {
  clear: both;
}

```

Και το τελικό αποτέλεσμα όπου φαίνεται η προσαρμογή του περιεχομένου προκειμένου να δημιουργηθούν οι αποστάσεις των 2em και παράλληλα να διατηρηθούν τα αρχικά πλάτη.



Floats και οριζόντια μπάρα πλοήγησης

Ένα άλλο πρόβλημα που προκύπτει από την επιβολή της ιδιότητας float παρουσιάζεται όταν κατά την κατασκευή της οριζόντιας μπάρας πλοήγησης επιβάλω την ιδιότητα float: left στα στοιχεία που θα λειτουργήσουν ως κουμπιά προκειμένου να τα παρατάξω το ένα δίπλα στο άλλο. Τα στοιχεία αυτά βρίσκονται μέσα σε έναν υποδοχέα συνήθως μέσα σε ένα στοιχείο `<div>`. Μετά την επιβολή της ιδιότητας ο υποδοχέας συρρικνώνεται και τα στοιχεία πέφτουν έξω από το κουτί. Η επαναφορά γίνεται με έναν από τους εξής δύο τρόπους:

A' Τρόπος: Στον υποδοχέα - στο στοιχείο `<div>` επιβάλω την ιδιότητα **overflow: hidden**

B' Τρόπος: Στον υποδοχέα - στο στοιχείο `<div>` επιβάλω την ιδιότητα **display: flow-root**

Και οι δύο τρόποι έχουν το ίδιο αποτέλεσμα.

Ας δούμε το πιο κάτω παράδειγμα:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Floats</title>
    <style>
      .container{
        background-color: black;
        color: white;
      }
      .div1, .div2{
        width: 100px;
        height: 100px;
        border-radius:10%;
        float: left;
        margin: 10px;
        text-align: center;
        line-height: 100px;
      }
      .div1{
        background-color: cyan;
      }
      .div2{
        background-color: red;
      }
    </style>
  </head>
  <body>
    <div class="container">
      Good Morning
      <div class="div1">Div1</div>
      <div class="div2">Div2</div>
    </div>
  </body>
</html>
```

Οι δύο div μετά την επιβολή των floats πέφτουν έξω από το κουτί container με τον ίδιο τρόπο που πέφτουν και τα στοιχεία της οριζόντιας μπάρας. Ο υποδοχέας εδώ περιέχει και τη φράση Good Morning γι αυτό και δεν συρρικνώνεται εντελώς.



Για να ταβάλω μέσα στον υποδοχέα θα πρέπει να προσθέσω στον υποδοχέα την ιδιότητα **overflow: hidden** ή την **display: flow-root**

```
.container{
    background-color: black;
    color: white;
    display: flow-root;
}
```



Επίσης στο ίδιο αποτέλεσμα καταλήγω αν αμέσως μετά την Div2 προσθέσω το στοιχείο `<br>` με κλάση `clearfloat`

```
<div class="container">
    Good Morning
    <div class="div1">Div1</div>
    <div class="div2">Div2</div>
    <br class="clearfloat">
</div>
```

και συνδυάσω με αυτό το στοιχείο τον γνωστό κανόνα

```
.clearfix {  
  clear: both;  
}
```

Αν μετά την Div2 ακολουθούσαν και άλλα στοιχεία στα οποία δεν είχε επιβληθεί η ιδιότητα `float: left` τότε για να τα φέρω να κάτσουν όλα σωστά μέσα στον υποδοχέα πρέπει υποχρεωτικά να δουλέψω με το `<br>`.