



ΠΕΡΙΕΧΟΜΕΝΑ:

1. Γενικά για το Flex
2. Στοιχίση Στοιχείων Flex
3. Ονομασία Αξόνων
4. Αναδίπλωση Στοιχείων: flex-wrap
5. Παραδείγματα: Δημιουργία μενού
6. Ασκήσεις

Γιώργος Τ.

Σμαραγδένιος Χορηγός Μαθήματος

Δημήτρης Πατριαρχέας

Χρυσός Χορηγός Μαθήματος

ΜΑΘΗΜΑ 7.1: FLEXBOX: CONTAINER

1. Γενικά για το Flex

CSS Flexible Box Layout Module:

- Οργανώνει τα παιδιά ενός στοιχείου **κατά μήκος μίας γραμμής** (οριζόντιας ή κάθετης), με «ευέλικτο» τρόπο.

Πως δουλεύει:

- Θέτουμε το **display** property σε ένα στοιχείο (π.χ. το body) ίσο με **flex**. Λέμε ότι το στοιχείο είναι **“flex container”**

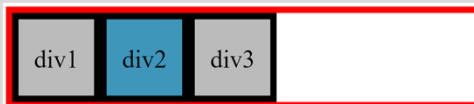
```
body {  
  display: flex;  
}
```

- τα παιδιά του στοιχείου γίνονται αυτόματα **flex-items** (ακόμη ένα είδος display) με δικές τους ιδιότητες
- και τοποθετούνται κατά μήκος μίας οριζόντιας γραμμής

Παράδειγμα 1: flex

```
<body>  
<div>div1</div>  
<div>div2</div>  
<div>div3</div>  
</body>
```

```
body {  
  display: flex;  
  border: 10px red solid;  
}
```



flex-direction: Container: Κατεύθυνση των flex-items

τιμές	row, column, row-reverse, column-reverse		
default	row		
Στοιχεία	flex containers	Κληρονόμηση	Όχι

- Το βάζουμε στο container. Τα flex-items θα εμφανιστούν:
 - row**: Οριζόντια γραμμή, ευθεία σειρά
 - row-reverse**: Οριζόντια γραμμή, αντίστροφη σειρά
 - column**: Κάθετη γραμμή, ευθεία σειρά
 - column-reverse**: Κάθετη γραμμή, αντίστροφη σειρά

Παράδειγμα 2: flex-direction

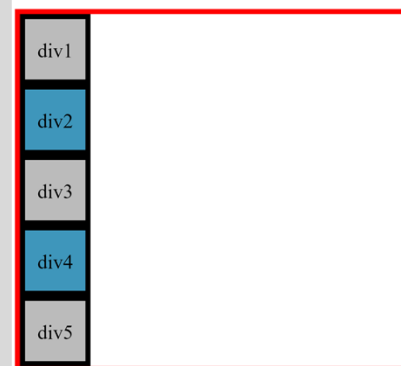
row



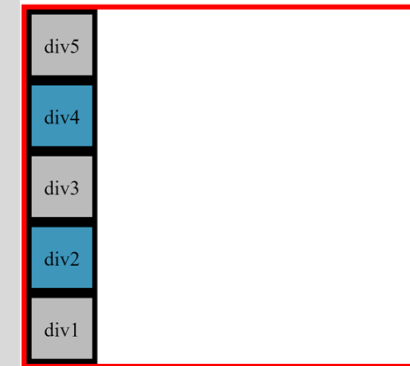
row-reverse



column



column-reverse



Παρατηρήσεις για τα flex items:

- Δεν υπάρχουν collapsing margins. Θα προβάλλονται κανονικά τα margins και το padding και του container και των items.
- Τα properties float και clear πλέον δεν δουλεύουν στα flex items
- Στο μάθημα αυτό μελετάμε ιδιότητες του container. Στο επόμενο μάθημα θα δούμε με λεπτομέρεια τη λειτουργία των flex items.

justify-content: Container: Στοιχισή των flex-items στον κύριο άξονα

τιμές	center, flex-start, flex-end, space-between, space-around, space-evenly		
default	flex-start		
Στοιχεία	flex containers	Κληρονόμηση	Όχι

- Το βάζουμε στο container. Καθορίζει τη στοίχιση των flex-items στον κύριο άξονα(βλ. παράδειγμα)

Παράδειγμα 3: justify-content

```
<p>flex-start</p>
<div class="container1">
  <div>div1</div>
  <div>div2</div>
  <div>div3</div>
  <div>div4</div>
  <div>div5</div>
</div>
```

```
.container1 {
  display: flex;
  justify-content: flex-start;
}
```

flex-start



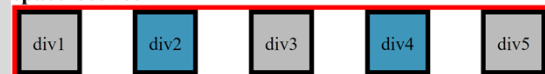
flex-end



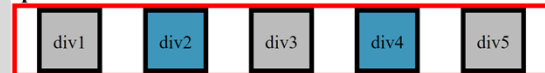
center



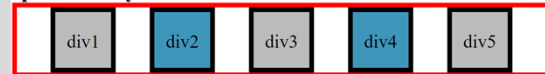
space-between



space-around



space-evenly



align-items: Container: Στοιχισή των flex-items στον δευτερεύοντα άξονα

τιμές	flex-start, flex-end, center, stretch, baseline		
default	stretch		
Στοιχεία	flex containers	Κληρονόμηση	Όχι

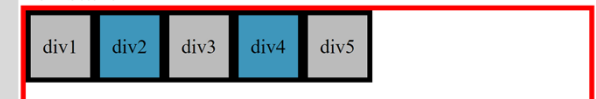
- Το βάζουμε στο container. Καθορίζει τη στοίχιση των flex-items στο δευτερεύοντα άξονα(βλ. παράδειγμα)

Παράδειγμα 4: align-items

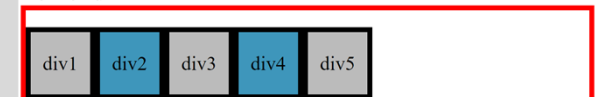
```
div {
  border: 10px red solid;
}
div div {
  border: 10px black solid;
  padding: 40px 0;
  width: 140px;
  text-align: center;
}
```

```
.container5 {
  display: flex;
  align-items: stretch;
}
div[class^="container"] {
  height: 200px;
}
```

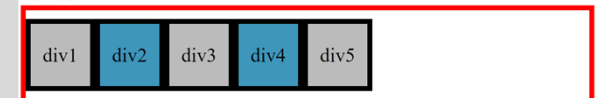
flex-start



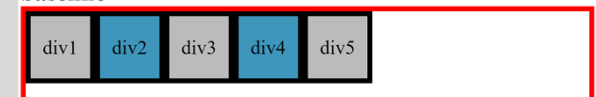
flex-end



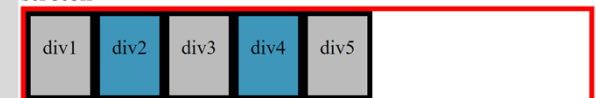
center



baseline



stretch

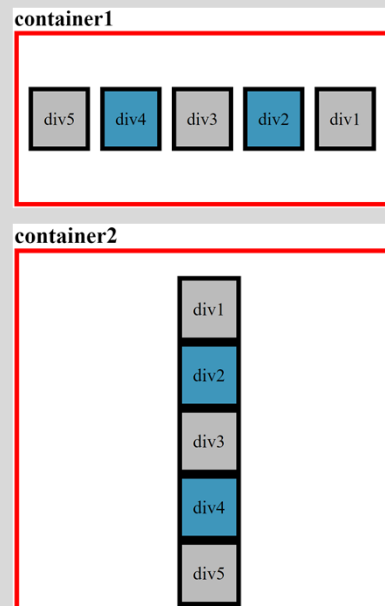


Πρωτεύων και Δευτερεύων Άξονας:

- Ανάλογα με την τιμή που έχουμε βάλει στο flex-direction:
 - row και row-reverse: κύριος άξονας είναι η γραμμή
 - justify-content: στοίχιση κατά τη γραμμή (πρωτεύων άξονας - **main axis**)
 - align-items: στοίχιση κατά τη στήλη (δευτερεύων άξονας - **cross axis**)
 - column και column-reverse: κύριος άξονας είναι η γραμμή
 - justify-content: στοίχιση κατά τη στήλη (πρωτεύων άξονας)
 - align-items: στοίχιση κατά τη γραμμή (δευτερεύων άξονας)

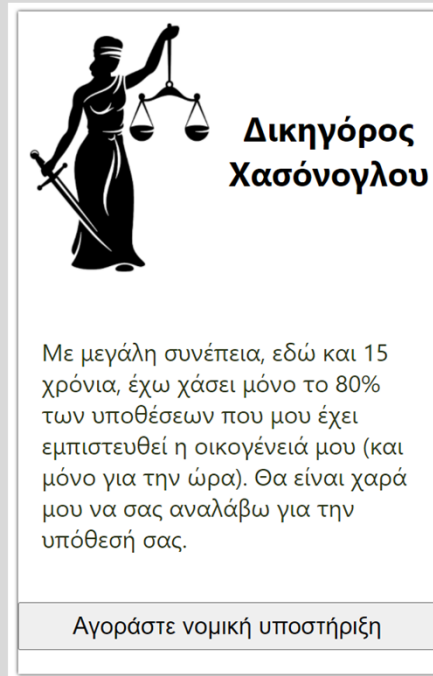
Παράδειγμα 5: axes

```
.container1 {
  display: flex;
  flex-direction: row-reverse;
  justify-content: space-evenly;
  align-items: center;
  height: 400px;
}
.container2 {
  display: flex;
  flex-direction: column;
  justify-content: flex-end;
  align-items: center;
  height: 800px;
}
```



Άσκηση 1:

- Ξεκινώντας από το «exercise01-initial.html» και εισάγοντας μόνο κανόνες CSS Flex, επιτύχετε το ακόλουθο αποτέλεσμα:



flex-wrap: Container: Αναδίπλωση flex-items όταν δεν χωράνε

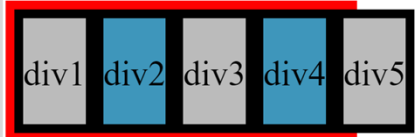
τιμές	wrap, nowrap, wrap-reverse		
default	nowrap		
Στοιχεία	flex containers	Κληρονόμηση	Όχι

- wrap: Αν τα στοιχεία δεν χωράνε στον container, τότε θα αναδιπλώνονται σε επομ. γραμμές (ή στήλες)
- nowrap (default): Γίνεται overflow
- wrap-reverse: Αν π.χ. δουλεύουμε κατά μήκος μίας γραμμής, τότε το wrap θα γίνεται στην από πάνω γραμμή κ.ο.κ.

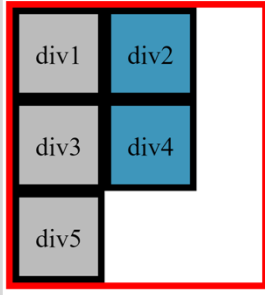
Παράδειγμα 6: flex-wrap

```
.container1 {
  display: flex;
  flex-direction: row;
  width: 400px;
}
.container2 {
  display: flex;
  flex-direction: row;
  width: 400px;
  flex-wrap: wrap;
}
.container3 {
  display: flex;
  flex-direction: row;
  width: 400px;
  flex-wrap: wrap-reverse;
}
```

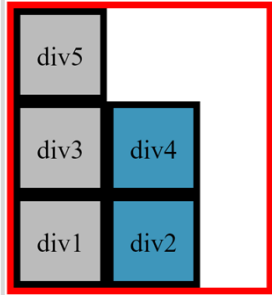
nowrap



wrap



wrap-reverse



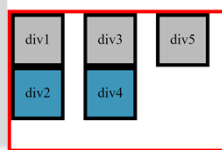
align-content: Container: Κενός χώρος μεταξύ «σπασμένων» ομάδων

τιμές	flex-start, flex-end, center, stretch, baseline, normal		
default	normal		
Στοιχεία	flex containers	Κληρονόμηση	Όχι

- Βρίσκει εφαρμογή *μόνο* όταν περιέχονται πολλές γραμμές (με wrap), οπότε και καθορίζει τον κενό χώρο μεταξύ τους.

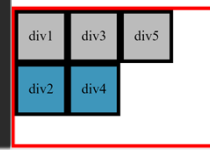
Παράδειγμα 7: align-content

align-content: normal



```
display: flex;
height: 400px;
flex-direction: column;
flex-wrap: wrap;
align-content: flex-start;
```

align-content: flex-start



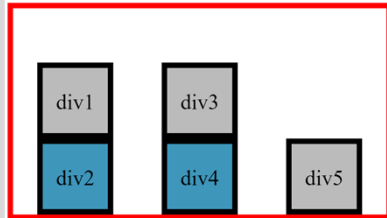
flex-flow: Container: Σύντμηση flex-direction και flex-wrap

τιμές	flex-direction flex-wrap		
default	row nowrap		
Στοιχεία	flex containers	Κληρονόμηση	Όχι

Παράδειγμα 8: flex-flow

```
.container1 {
  display: flex;
  flex-flow: column wrap;
  justify-content: flex-end;
  align-items: center;
  height: 400px;
}
```

flex-flow



Μέσω FlexBox η δημιουργία μενού γίνεται πολύ εύκολη:

- Κορμός HTML
 - header που περιέχει nav
 - nav με λίστα (ul, li) που περιέχει τους συνδέσμους
 - Η λίστα μετατρέπεται σε flex και έπειτα αρχίζουμε την εφαρμογή στυλ της αρεσκείας μας

Παράδειγμα 9: menu1

```
<header>
<nav>
<ul>
<li><a href="">Home</a></li>
<li><a href="">Products</a></li>
<li><a href="">About</a></li>
<li><a href="">Contact</a></li>
</ul>
</nav>
</header>
```

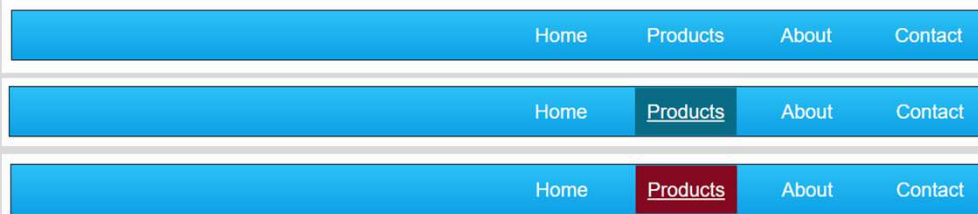
```
nav ul {
display: flex;
justify-content: flex-end;
outline: 1px solid;
padding: 10px;
list-style-type: none;
}
nav li {
width: 100px;
}
nav a {
text-decoration: none;
color: inherit;
}
nav a:visited {
}
nav a:hover {
}
nav a:active {
}
```

Home Products About Contact

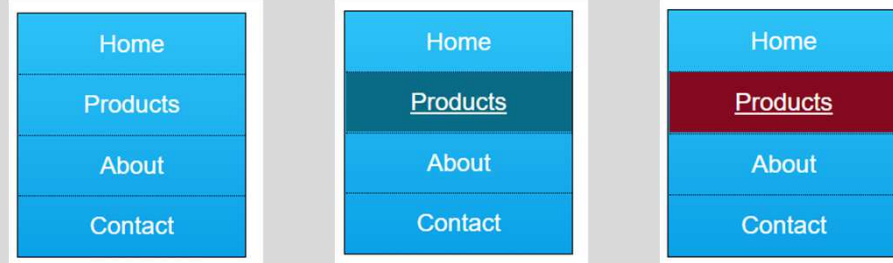
Παράδειγμα 10: menu2

```
body {
max-width: 800px;
margin: auto;
font-family: sans-serif;
}
nav {
background-image: linear-gradient(to
bottom, #2ec2f8, #0ca0e7);
}
nav ul {
list-style-type: none;
display: flex;
justify-content: flex-end;
border: 1px solid black;
color: white;
}
```

```
nav li {
width: 100px;
padding: 10px;
text-align: center;
}
nav a {
text-decoration: none;
color: inherit;
padding: 10px;
}
nav a:hover {
text-decoration: underline;
background-color: #096b85;
}
nav a:active {
text-decoration: underline;
background-color: #850920;
}
```

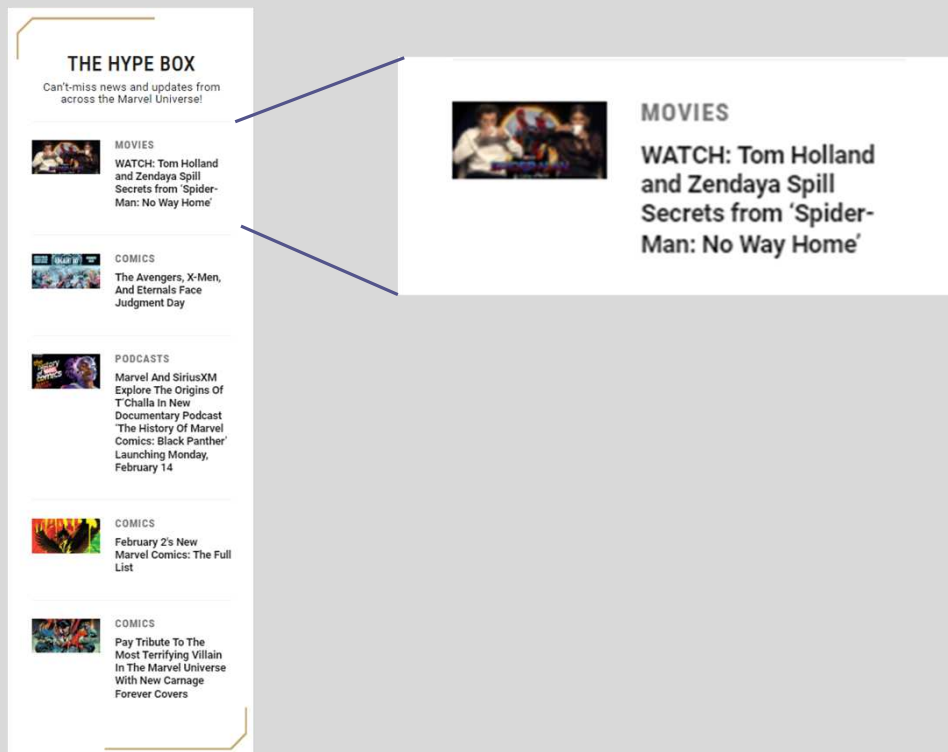


Παράδειγμα 11: menu3



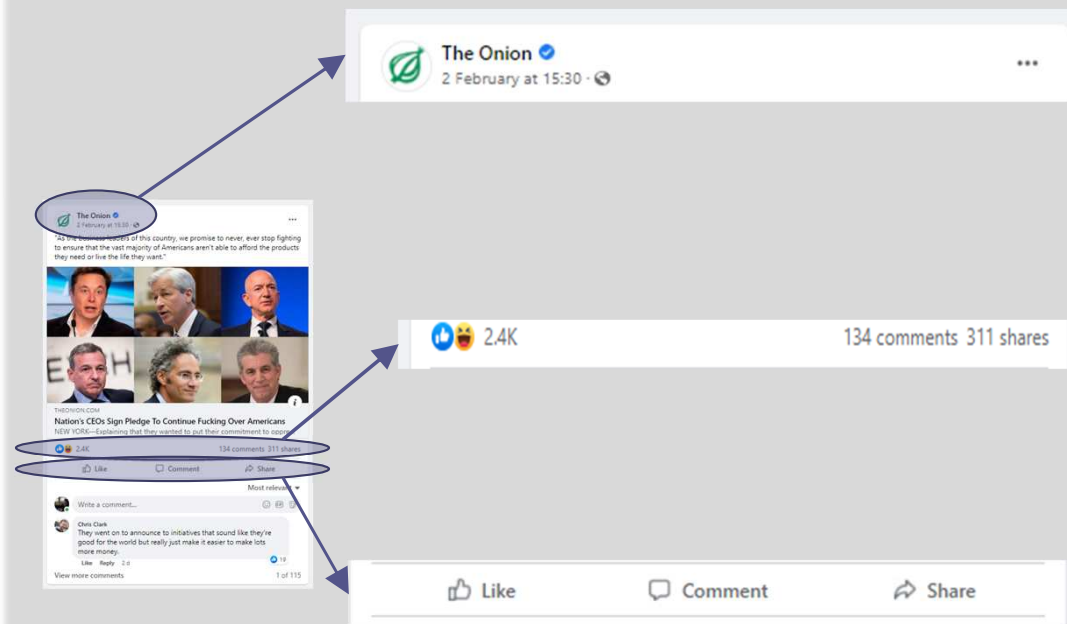
Άσκηση 2: Marvel Comics

- Συνέχεια της άσκησης 2 / μάθημα 5.3 (αρχείο exercise01-initial.html)
- Οργανώστε κάθε «hyper-item» με το flexbox, ώστε να έχει την επιθυμητή μορφή όπως φαίνεται στην εικόνα:



Άσκηση 3: Facebook

- Συνέχεια της άσκησης 3 / μάθημα 5.3 (αρχείο exercise02-initial.html)
- Με χρήση του flexbox, προσεγγίστε καλύτερα την εμφάνιση των ακόλουθων περιοχών του post του facebook:



Σημείωση:

- Θα χρειαστεί να αλλάξετε και το μέγεθος της εικόνας (με το property width και ενδεικτική τιμή 100px)

Σημείωση:

- Θα χρειαστούν και τροποποιήσεις στον κώδικα HTML με προσθήκη επιπλέον DIVs. Προχωρήστε σε όποιες αλλαγές κρίνετε απαραίτητες.