



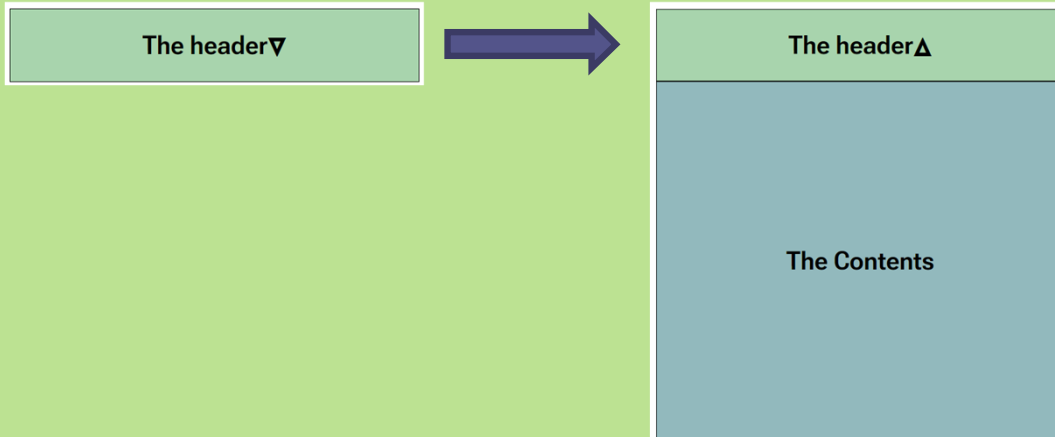
ΠΕΡΙΕΧΟΜΕΝΑ:

1. Στοιχεία Περιεχομένου
 1. Accordion
 2. Breadcrumb - Pagination
2. JS Preview
 1. Slider
 2. Tabs
3. Ασκήσεις στις Πίτες του Πίτου
 1. Άσκηση: Aside Section
 2. Άσκηση: Ενσωμάτωση Slider
4. Ανασκόπηση - Βιβλιογραφία
5. Επόμενα Βήματα

ΜΑΘΗΜΑ 12.5: ΣΤΟΙΧΕΙΑ ΠΕΡΙΕΧΟΜΕΝΟΥ

Accordion:

- Στοιχείο που όταν κάνουμε κλικ πάνω του, εμφανίζει περιεχόμενο.



- Εφαρμόζουμε την ίδια τεχνική που είδαμε στο hamburger menu (Μάθημα 12.1):
 - (Ένα checkbox που δεν εμφανίζεται και μέσω της ετικέτας του προκαλούμε την εμφάνιση/απόκρυψη του περιεχομένου)

Παράδειγμα 1: accordion

```
<div class="accordion">
  <input class="accordion-checkbox" type="checkbox" id="checkbox"/>
  <label class="accordion-label" for="checkbox">
    The header
  </label>
  <div class="accordion-content">
    The Contents
  </div>
</div>
```



1.1. Accordion

```
Sass

@mixin centered {
  display: flex;
  justify-content: center;
  align-items: center;
}

@import
url('https://fonts.googleapis.com/css2?family=Roboto+Flex:opsz,wght@8..144,400;8..144,600&display=swap');
* {
  box-sizing: border-box;
}

body {
  font-family: "Roboto Flex", sans-serif;
  font-size: 180%;
  font-weight: bold;
}

/* Mechanics */
.accordion-checkbox {
  display: none;
}

&:checked ~ .accordion-content {
  @include centered;
}

&:checked ~ .accordion-label::after {
  content: "Δ"
}

.accordion-label {
  display: block;
}

&::after {
  content: "▽";
}

.accordion-content {
  display: none;
}

/* Style */
.accordion-label {
  border: 1px black solid;
  padding: 20px;
  background-color: #A8D4AD;
  @include centered;
}

.accordion-content {
  border: 1px black solid;
  height: 400px;
  background-color: #92B9BD;
}
```

Breadcrumb:

- Σε websites με πολύ περιεχόμενο, οργανωμένο δενδρικά:
 - Θέλουμε να ενημερώνουμε το χρήστη για τις διαδοχικές επιλογές που έχουν οδηγήσει στην τρέχουσα σελίδα.
 - Και να του δώσουμε έναν εύκολο τρόπο ώστε να πλοηγηθεί σε κάποιο γονέα της τρέχουσας σελίδας.

Παράδειγμα 2: breadcrumb

```
<div class="breadcrumb">
  <ul>
    <li><a href="#">Home</a></li>
    <li><a href="#">Μαθήματα</a></li>
    <li><a href="#">CSS</a></li>
    <li><a href="#">Μάθημα 12.4: Φόρμες</a></li>
    <li>Φόρμες που "δουλεύουν"</li>
  </ul>
</div>
```

```
.breadcrumb ul {
  list-style-type: none;
  margin: 0;
  padding: 0;
  li {
    display: inline;
    a {
      text-decoration: none;
      color: #0b8181;
      &:hover {
        color: #1c2c8f;
        font-weight: bold;
      }
      &:active {
        color: #9e4031;
      }
    }
  }
  &:after {
    content: " / "
  }
  &:last-of-type:after {
    content: ""
  }
}
```

Header

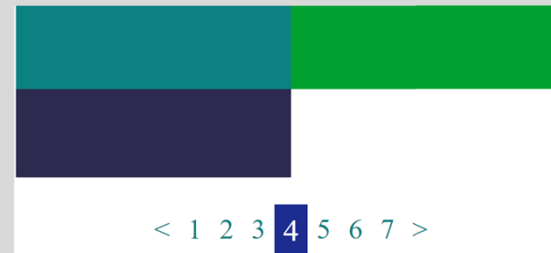
Home / Μαθήματα / CSS / Μάθημα 12.4: Φόρμες / Φόρμες που "δουλεύουν"

Pagination:

- Εμφανίζεται συνήθως κάτω από τα αποτελέσματα αναζήτησης
- Καθιστά εφικτή την πλοήγηση σε αυτά τα αποτελέσματα.

Παράδειγμα 3: pagination

```
<div class="pagination">
  <ul>
    <li><a href="#">&lt;</a></li>
    <li><a href="#">1</a></li>
    ...
    <li><a class="active" href="#">4</a></li>
    <li><a href="#">5</a></li>
    ...
    <li><a href="#">&gt;></a></li>
  </ul>
</div>
```

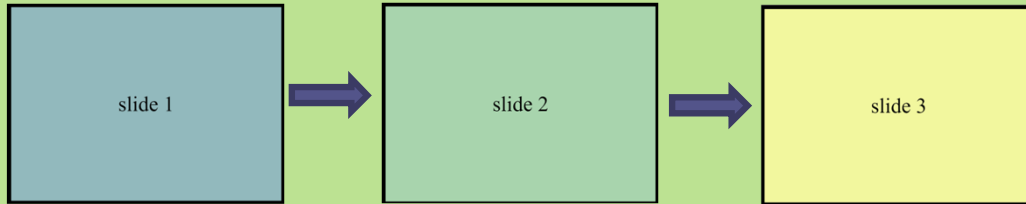


```
.pagination ul {
  list-style-type: none;
  margin: 40px;
  padding: 0;
  font-size: 2em;
  display: flex;
  justify-content: center;
  li {
    display: inline;
    a {
      text-decoration: none;
      color: #0b8181;
      padding: 10px;
      &:hover {
        color: #1c2c8f;
        font-weight: bold;
      }
      &:active {
        color: #9e4031;
      }
      &:active {
        font-size: 1.1em;
        background-color: #1c2c8f;
        color: white;
      }
    }
  }
}
```

ΜΑΘΗΜΑ 12.5: ΣΤΟΙΧΕΙΑ ΠΕΡΙΕΧΟΜΕΝΟΥ

Με την JavaScript:

- Μπορούμε να έχουμε όσες καταστάσεις θέλουμε (και όχι μόνο 2)
- Χαρακτηριστικό παράδειγμα: **Slider** (εναλλαγή μεταξύ πολλών καταστάσεων):



Παράδειγμα 2: slider

```
<body>
<div class="slider">
  <div class="slide slide1">
    slide 1
  </div>
  <div class="slide slide2">
    slide 2
  </div>
  <div class="slide slide3">
    slide 3
  </div>
  <a class="previous" onclick="plusSlides(-1)">Previous</a>
  <a class="next" onclick="plusSlides(1)">Next</a>
</div>

<script src="js/slider.js"></script>
</body>
```

2.1. JS Preview: Slider

```
Sass
/* src (altered):
https://www.w3schools.com/howto/howto_js_slideshow.asp */
let slideIndex = 1;
showSlides(slideIndex);

// Next/previous controls
function plusSlides(n) {
  showSlides(slideIndex += n);
}

// Thumbnail image controls
function currentSlide(n) {
  showSlides(slideIndex = n);
}

function showSlides(n) {
  let i;
  let slides = document.getElementsByClassName("slide");
  if (n > slides.length) {slideIndex = 1}
  if (n < 1) {slideIndex = slides.length}
  for (i = 0; i < slides.length; i++) {
    slides[i].style.display = "none";
  }
  slides[slideIndex-1].style.display = "block";
}
```

```
HTML
.slide {
  display: none;
}

.slide {
  border: 10px black solid;
  font-size: 50px;
  padding: 200px;
  text-align: center;
}

.slide1 {
  background-color: #92B9BD;
}

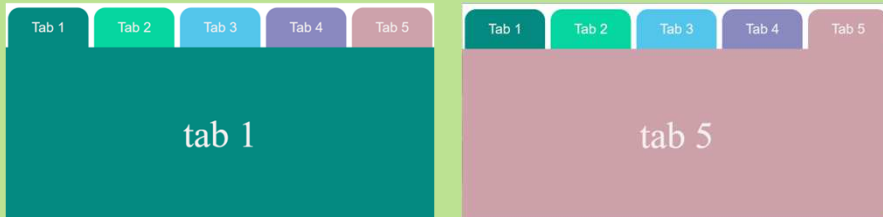
.slide2 {
  background-color: #A8D4AD;
}

.slide3 {
  background-color: #F2F79E;
}
```

ΜΑΘΗΜΑ 12.5: ΣΤΟΙΧΕΙΑ ΠΕΡΙΕΧΟΜΕΝΟΥ

Tabs:

- Εμφανίζει μόνο το περιεχόμενο που έχει σχετιστεί με ένα tab.
- Απαιτείται η εναλλαγή μεταξύ πολλών καταστάσεων (άρα και η JS)



Παράδειγμα 2: slider

```
<div class="tabs">
  <ul>
    <li><button class="tab1" onclick="tab(1)">Tab 1</button></li>
    <li><button class="tab2" onclick="tab(2)">Tab 2</button></li>
    <li><button class="tab3" onclick="tab(3)">Tab 3</button></li>
    <li><button class="tab4" onclick="tab(4)">Tab 4</button></li>
    <li><button class="tab5" onclick="tab(5)">Tab 5</button></li>
  </ul>
  <div class="tab tab1">
    <span>tab 1</span>
  </div>
  <div class="tab tab2">
    <span>tab 2</span>
  </div>
  <div class="tab tab3">
    <span>tab 3</span>
  </div>
  <div class="tab tab4">
    <span>tab 4</span>
  </div>
  <div class="tab tab5">
    <span>tab 5</span>
  </div>
</div>
```



2.2. JS Preview: Tabs

```
@use "sass:list";
$bodyVerticalMargin: 10px;
$colors: [#048A81, #06D6A0, #54C6EB, #8A89C0, #CDA2AB];
```

Sass

```
div.tabs {
  @for $var from 1 through list.length($colors) {
    .tab#{$var} {
      background-color: list.nth($colors, $var);
    }
  }
}
```

```
display: flex;
flex-direction: column;
height: calc(100vh - #{$bodyVerticalMargin});
```

```
div.tab {
  flex: 1;
  font-size: 66px;
```

```
display: flex;
justify-content: center;
align-items: center;
color: #f5f1f1;
}
```

```
ul {
  list-style-type: none;
  display: flex;
  height: 70px;
  width: 100%;
  margin: 0;
  padding: 0;

  font-size: 2em;
  li {
    width: 100%;
    margin: 0 5px;
    button {
      border-radius: 20px 20px 0 0;
      border: 0;
      height: 100%;
      width: 100%;
      color: #f5f1f1;
      font-size: 24px;
    }
  }
}
```

```
function tab(n) {
  let tabContents = document.querySelectorAll('.tab')
  for (let i = 0; i < tabContents.length; i++) {
    tabContents[i].style.display = "none";
  }
  let clicked = document.querySelectorAll("div.tab" + n);
  let item = clicked[0];
  item.style.display = "flex";
}
```

```
tab(1);
```



Άσκηση 1:

Χρησιμοποιώντας το template για flipping card που κατασκευάσαμε στο μάθημα 9.6 / Άσκηση 1:

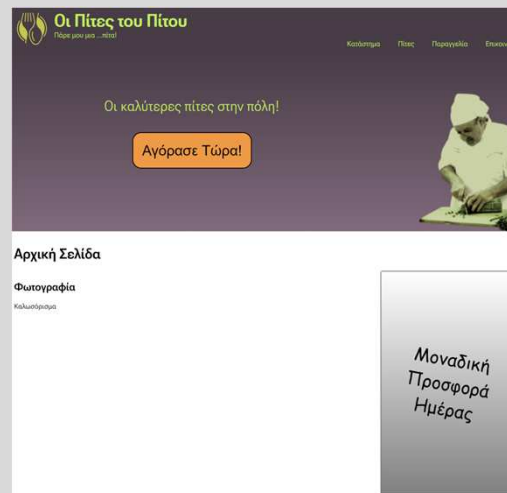
- Προσαρμόστε το aside section ώστε να έχει μία flipping card με προσφορά:



- Έπειτα κάνετε responsive το main section της σελίδας ως εξής:



κινητό, tablet

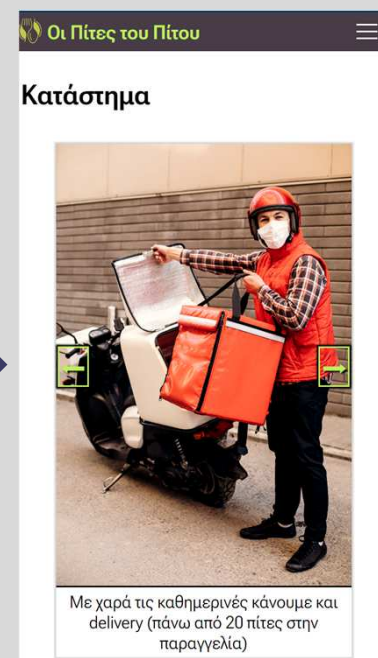
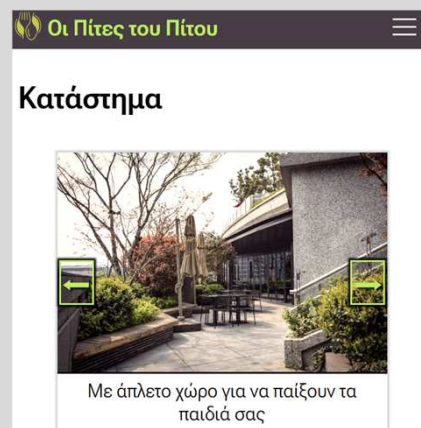


laptop και άνω

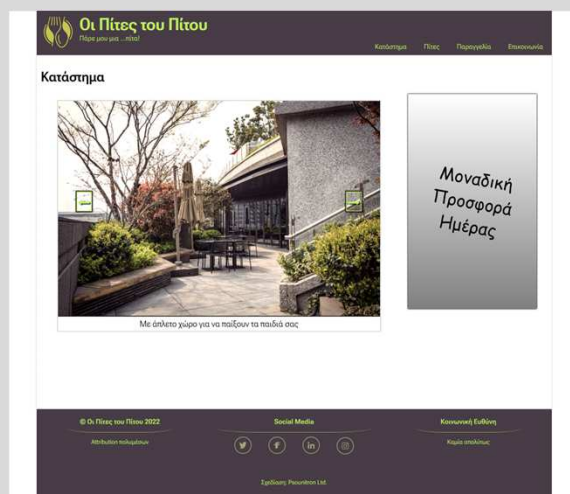
Άσκηση 1:

Εισάγετε slider με τις φωτογραφίες που εμφανίζονται στο κατάστημα:

- Ενδεικτική εμφάνιση σε κινητό:



- Σε όλες τις οθόνες, το slider θα πρέπει να πιάνει το χώρο που έχουμε προβλέψει ήδη (π.χ. για οθόνες tv):



4. Ανασκόπηση - Βιβλιογραφία



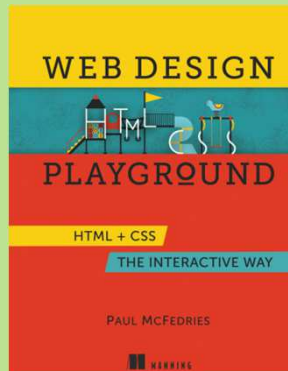
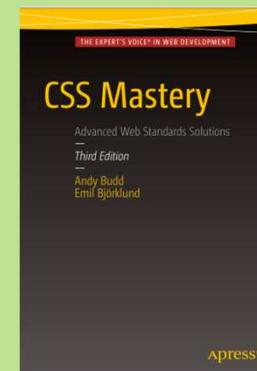
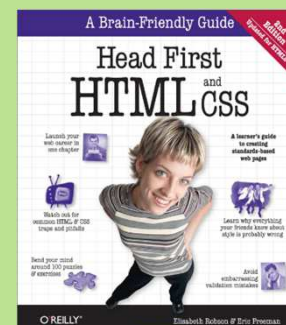
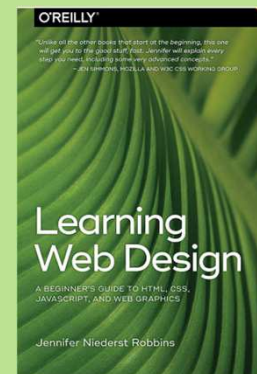
Ανασκόπηση:

- 1-3: Τα Βασικά
- 4: Κείμενο
- 5: Κουτί
- 6-7-8: Layout: Old, Flex, Grid
- 9: Γραφικά
- 10: RWD
- 11: SASS
- 12: Project

important	Style που έχει μαρκιαριστεί ως imp (author < inline < user)
user	User Style (στυλ χρήστη στον browser)
inline	Inline Style στον κώδικα HTML
specificity	Έχουν οριστεί τιμές στο property επιλογής. Ο νικητής επιλέγεται με εξειδίκευση (ή πρώτης διαφοράς)
last rule	Έχει οριστεί τιμή στο property σε επιλογές και υπάρχει ισοπαλία σε αυτός που είναι τελευταίος στη σειρά
inherited	Δεν έχει οριστεί τιμή επιλογής. Το προτιμάει του γονέα του
	Δεν έχει οριστεί τιμή επιλογής. Το προτιμάει του γονέα του



Βιβλιογραφία Σειράς:



Ιστοσελίδες:

- mdn css
- w3schools
- css-tricks
- codepen
- freecodecamp
- stackoverflow ;-)

Special Αναφορά για Περαιτέρω Διάβασμα CSS:



Που βρισκόμαστε:

- HTML-CSS-SASS (ολοκληρωμένες)



Επόμενα Βήματα (εναλλακτικές κατευθύνσεις):

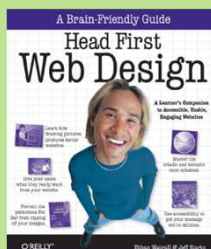
1. JavaScript:

- Γλώσσα Προγραμματισμού με την οποία μπορούμε να ορίσουμε συμπεριφορά στη σελίδα μας. HTML + CSS + JS = **Front End Programming**
- (όπως π.χ. στο Slider, πλέον μπορούμε να κάνουμε «κλικ» και να αλλάζει οποιοδήποτε σημείο της σελίδας)



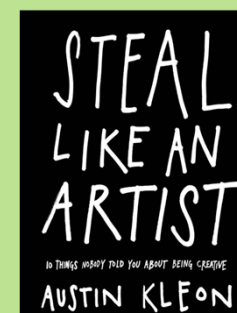
2. Δημιουργία projects:

- Βρείτε μικρούς επαγγελματίες και κατασκευάστε ολοκληρωμένα επαγγελματικά project.
- Με τις πληροφορίες σχετικά με τις φόρμες, μπορούμε να κατασκευάσουμε απλές επαγγελματικές σελίδες.
- Το ακόλουθο βιβλίο έχει μία επαγγελματική προσέγγιση (συναντήσεις με πελάτη, διαδοχικοί επανασχεδιασμοί, χρεώσεις κ.λπ.)



3. Web Design:

- Επέκταση των γνώσεων μας περί design και UX/UI.
- Προτεινόμενη Βιβλιογραφία:



4. Back-End Programming:

- Επιλογή ενός περιβάλλοντος και μελέτη του σε βάθος, π.χ.
 - Python/Django
 - PHP/Laravel
 - Java/Spring
 - Κ.Ο.Κ.
- Απαραίτητο για μεγάλα project όπου πλέον θα είναι δυνατή:
 - Η αποθήκευση/ανάσυρση δεδομένων (σε κάποια βάση π.χ. MySQL) και δημιουργία δυναμικών σελίδων
 - Η δημιουργία λογαριασμών χρήστη και ρόλων χρήστη.

Προτεινόμενη σειρά:

- Παράλληλα τα 2 και 3. Έπειτα το 1. Έπειτα το 4 (βλ. και βίντεο)
- google also: "front end roadmap" and ...keep learning!!