LEARN WEB

ale66

github.com/ale66/learn-web

THE LIMITS OF HTML

github.com/ale66/learn-web

THE PAGE ORIENTATION

So far, each HTML page is a stand-alone, complete page

Even recurrent elements, e.g., links to reference pages, are repeated

pages might share a common CSS file, but not the HTML structure itself

CASE STUDY

- unimi.it offers more than 150 study programs
- each programme page has a similar structure and most information is actually replicated
- yet, in principe we should come up with 150+ HTML pages

IDEA: SEMANTIC TAGGING

recurrent parts of the page are annotated by *invisible*, semantic tags

altough invisible, the tags guide presentation, and can be replicated across the whole platform, e.g., `unimi.it`

STATIC LAYOUT

github.com/ale66/learn-web

SO FAR

pages are intended as rectangular: relatively narrow and tall

- like the book page
- like the computer screen

SOLUTIONS, A

The CSS may define *boxes* with padding around them

The boxes may *float* so the browser will assign it dynamically to the available *viewport*

However, browsers present elements one after the other, until space on the right ends and they continue on a new line below: **undesirable effects and even collapse**

EXPERIENCE: FLOATING ELEMENTS

- Ex. 8.1: Floating elements

Elements start overflowing to the *line* below

Explict placing: not recommended

- Ex. 8.2: Positioning elements

EXPERIENCE: BOXES WITH PADDING

Creating boxes of text: Ex. 9.1

Padding around them: Ex. 9.2

For reference: a better finish

- Ex. 9.3: Applying a Simple Border
- Ex. 9.4: Controlling the Margins

EXERCISE

Try *pair coding* with the Live Share extension of VS Code

github.com/ale66/learn-web

- in your project, Box up the official text from `unimi.it` and make it float
- make a menu of navigation links and float it (details in past presentations)

LANDING PAGES

github.com/ale66/learn-web

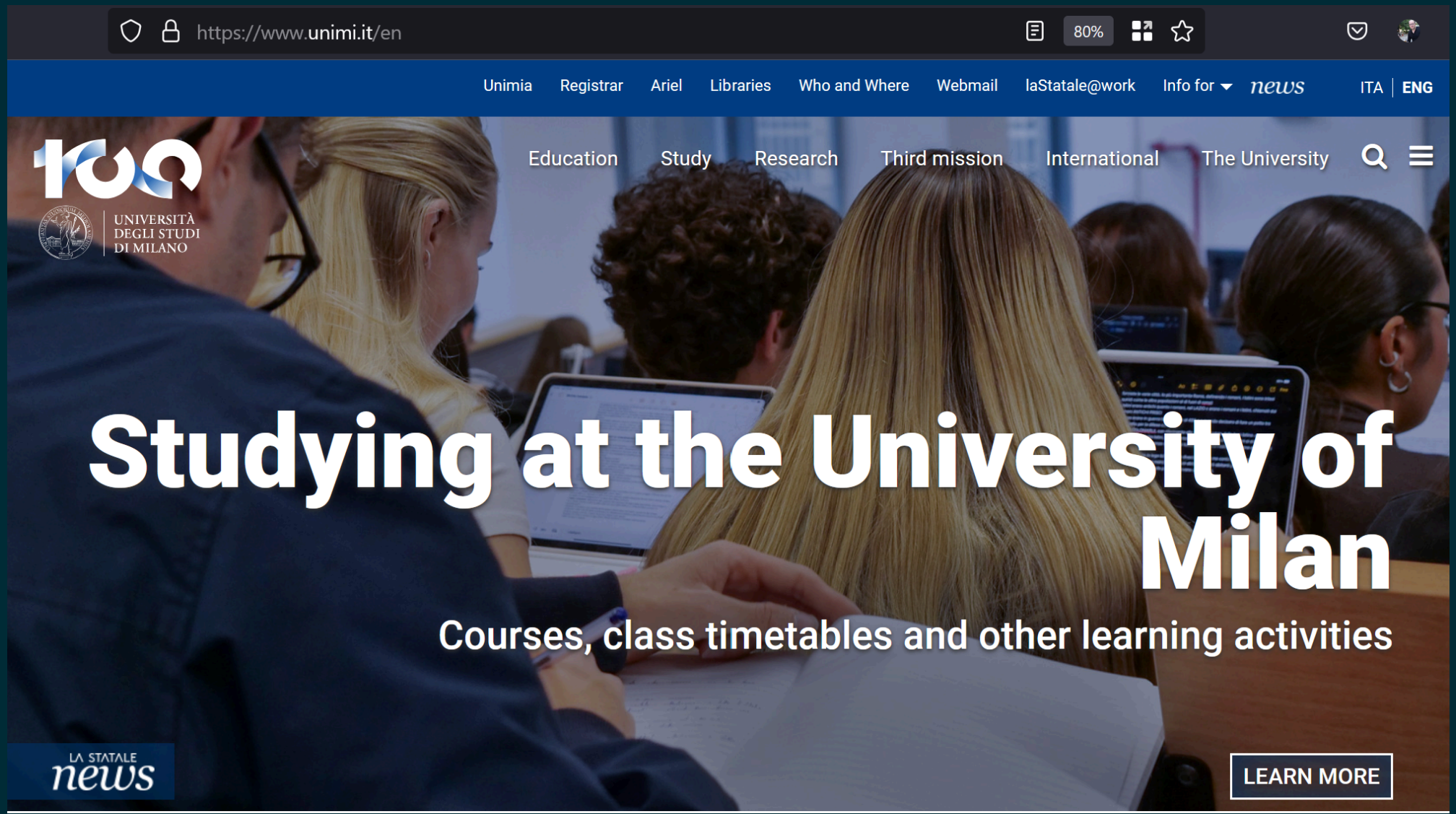
Hero page:

- seeks an emotional response
- graphics takes over from text

Landing page

- extends hero
- the page that search engines point
- seeks to entice and guide a deeper visit

EXAMPLE LANDING PAGE



github.com/ale66/learn-web

MAKE A LANDING PAGE

- Create a *hero image* that is fixed wrt. scrolling but also contains the claim
- go picture-in-picture, e.g.

github.com/ale66/learn-web

Fixed *landing*:

CSS

```css
.header {
    background: url(/images/landing-page-header-bg.jpg);
    background-attachment: fixed;
    background-position: right center;
    background-size: cover;
    padding-bottom: 1em;
    width: 100vw;
    height: 100vh;
}
```

Prevent the hero image from scrolling with the content.

HTML

```html
<div class="header">
</div>
```

Left picture is the *what*, background picture is the *where* or *how*

lab on *Velasca?* the other way around? Keep *lab* and choose an aspirational picture that works for you and your friends

THE IN-PICTURE

CSS

```
.header-image {
    float: left;
    width: 33%;
    margin-top: 3em;
    padding-right: 3em;
    text-align: right;
}
```

The CSS code for the image

HTML

```
<div class="header">
 <div class="header-image">
   <img src="/images/landing-page-book-cover.png"
alt="Front cover">
 </div>
</div>
```

The image is added as a standard HTML `img` element.

THE IN-TEXT

CSS

```
.header-info {
    float: right;
    width: 67%;
    margin-top: 4em;
    color: white;
}
```

The CSS code for the product info div

HTML

```
<div class="header">
   <div class="header-image">
     <img src="/images/landing-page-book-cover.png" alt="Front
cover">
   </div>
   <div class="header-info">
      <div class="surtitle">Coming Soonish!</div>
      <h1 class="title">The Web Designer</h1>
      <h3 class="subtitle">
         A story of HTML, CSS, and the big city</h3>
      <p class="intro">
         She knew HTML. She knew CSS. But did she know love?
Read this destined-to-be-remaindered novel that <em>The New
York Times Book Review</em> described as &ldquo;reasonably
grammatical&rdquo; and the <em>Times Literary Supplement</em>
called &ldquo;bathroom-worthy.&rdquo;  Pre-order your paperback
copy now for just $14.99.
     </p>
  </div>
</div>
```

The info is added within a div element.

Check the final effect with the Live server VS Code extension

github.com/ale66/learn-web

SOLUTIONS, B

Complex CSS files can define three (computer, tablet and smartphone) or more *locales*

sense the device and apply the appropriate styles

Example: section with links collapses into the *hamburger* menu

FOR REFRENCE: `before::` AND `after::`

::AFTER

We can avoid mis-rendering by forcing browsers to create the visual effects in a certain order

The `::` notation is for *pseudo-elements*: we don't see them in the HTML but they go into the *final* page.

The CSS is now changing the HTML!

::BEFORE

apply a certain style before the browser starts rendering the element

See the example from w3schools

```
p::before {
  content: "Read this -";
  background-color: yellow;
  color: red;
  font-weight: bold;
}
```

`::before, ::after` and `::first-letter` are advanced CSS language

we only cover them by example

See reference

STYLING TABLES AND OTHER STRUCTURED TEXTS

See the relevant self-learning material at

w3chools and

bootstrap