EPITECH. TECHNOLOGY

# WEB DEV SEMINAR

DAY 03 - HTML/CSS

# WEB DEV SEMINAR

## Contents

In the present subject, we will address responsive design, with a mobile-first approach.

Now that you know how to build an HTML page, let's see how to make it pretty and fit screens of all sizes: desktops, tablets, smartphones, etc…


✓ Contents
- Responsive Design
- Mobile first
- Task 01
- Task 02 - Smartphone
- Task 03 - Tablet
- Task 04 - Desktop
- Task 05
  * Step 01
  * Step 02
  * Step 03
  * Step 04
  * Step 05
  * Step 06

## Responsive Design

It is an approach used to develop sites that offer an optimal reading and browsing experience to the user, regardless of their device (smartphone, tablet, reader, desktop, printer, television, …)

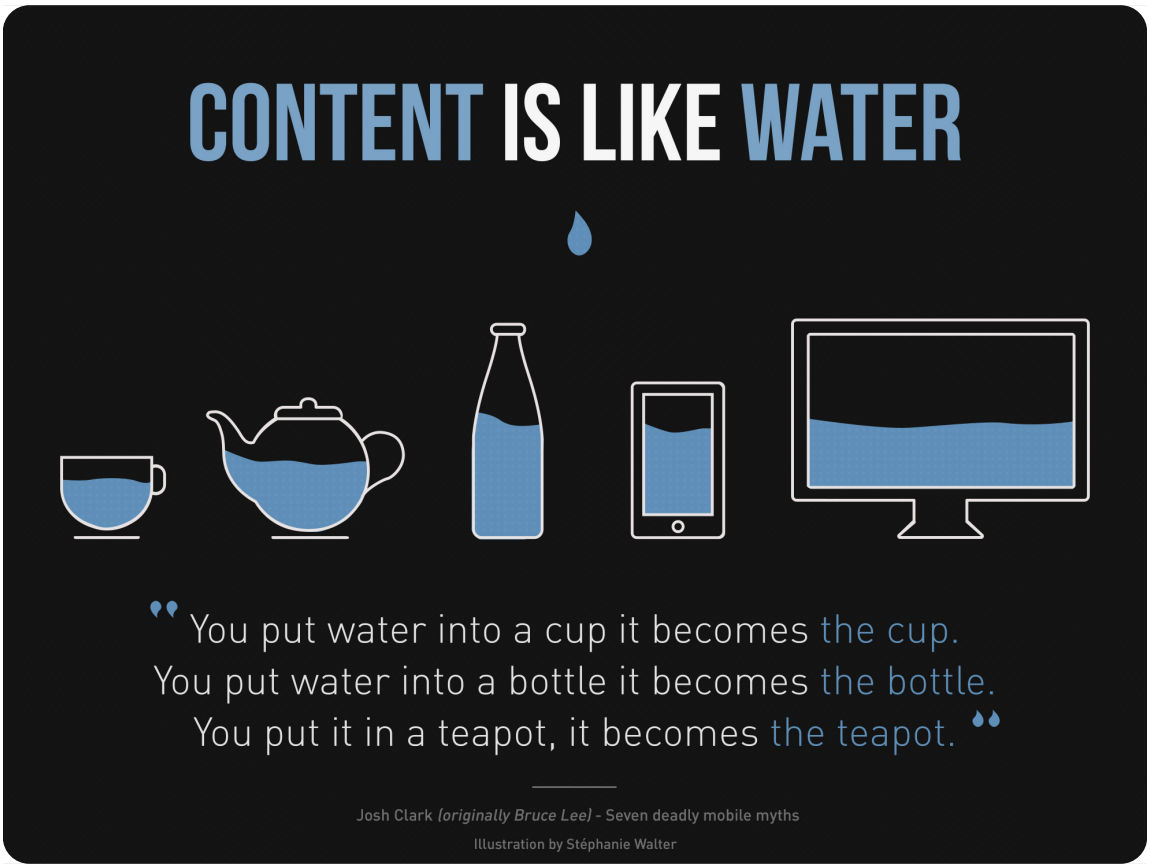
Traditionally, a web project that adopts a responsive design strategy will tend to think of the desktop version of the site as a starting point, and it will then be necessary to *reduce* this version as the size of the consultation materials decreases, according to a predefined grid. We call this **graceful degradation**.

This seems logical, since it feels like the majority of the web traffic comes from desktop widescreens when it actually comes from mobile devices (and most particularly smartphones).

However, if you take the contents of a bottle of water and pour it all into a glass of water…

## Mobile first

Another way of doing things has beeing talked about: the **mobile first** approach.




It is overflowing!

As the name suggests, this approach consists of thinking about navigation, ergonomics and site content for the smallest consultation materials first, and then upscaling them. This is called **progressive enhancement**.

If this approach begins to have some success over graceful degradation, it is because it has some undeniable advantages that do not only concern mobile versions of websites.

You must use **flexbox** and the `viewport` **meta tag** for these exercises.

You'll need to find the browser feature (or use an extension) that allows you to change the viewport. Learn to use your browser's dev tools, they will be your best friends!

## Task 01

**Delivery**: `./task01/index.html` and `./task01/style.css`

Create an **index.html** page and link it to a **style.css** file.

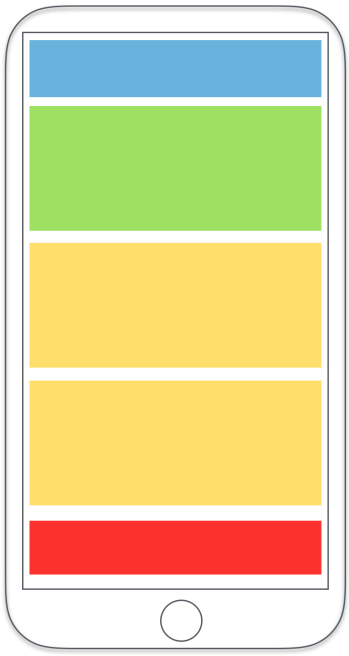
Structure your HTML with:

- ✓ a `header` containing a `nav`;
- ✓ a first `section`;
- ✓ a second `section` with six `articles`;
- ✓ a `footer`.

## Task 02 - Smartphone

**Delivery**: `./task02/index.html` and `./task02/style.css`

Reproduce this layout for a device with a maximum resolution width of 767px.



## Task 03 - Tablet

**Delivery**: `./task03/index.html` and `./task03/style.css`

Reproduce this layout for a device with a resolution width between 768px and 1024px.

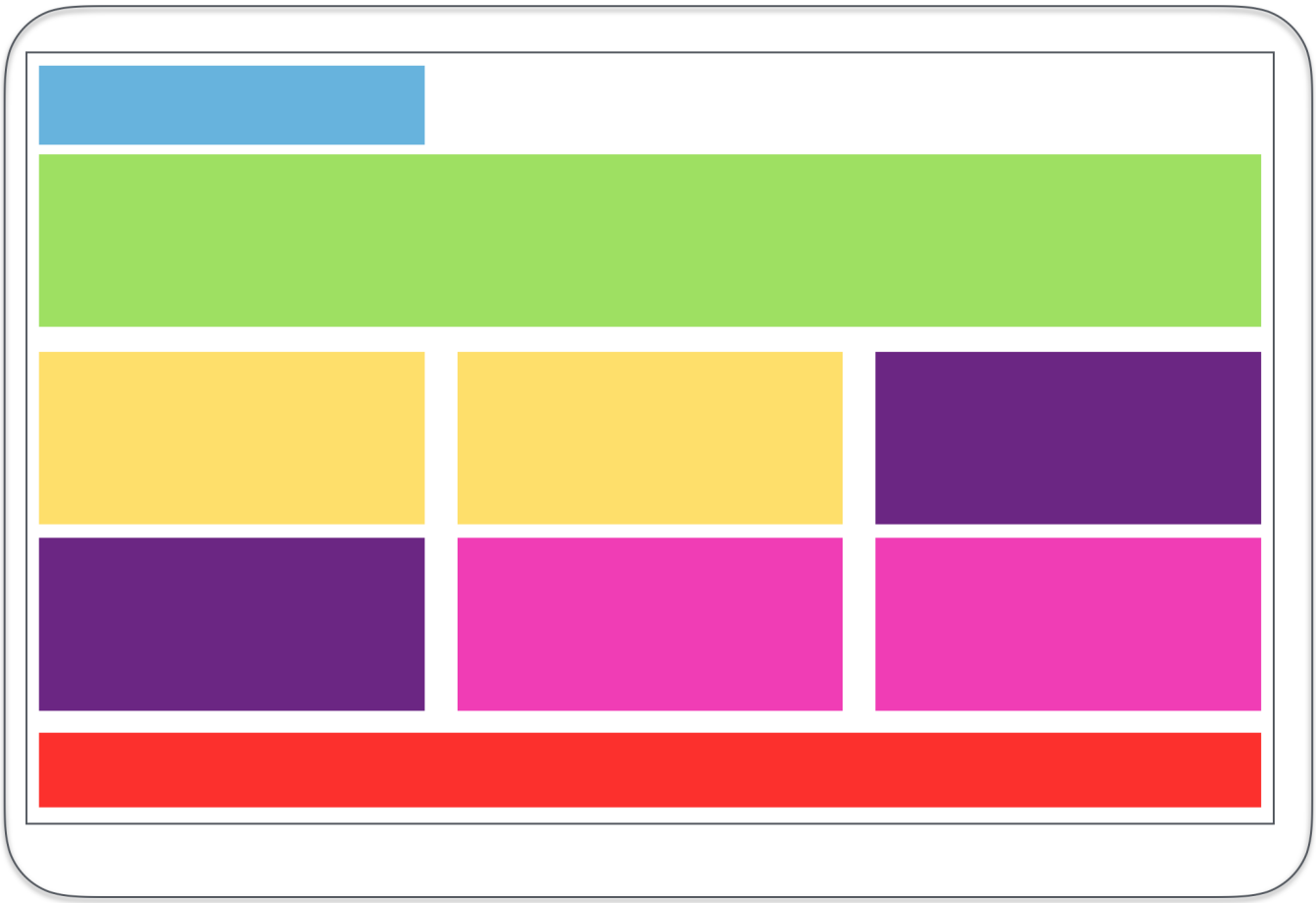
As you can see, two new articles appeared in the second section.

## Task 04 - Desktop

**Delivery**: `./task04/index.html` and `./task04/style.css`

Reproduce this layout for a device with a minimum resolution width of 1024px.

Again, you can see that two new articles appeared in the second section.



## Task 05

**Delivery**: `./task05/index.html`

For the next exercises, you must only use the **Materialize** CSS framework: do not create any CSS rule.
The HTML `style` attribute is forbidden.

Titles must be correctly implemented:

- ✓ LASTNAME FIRSTNAME
- ✓ Heading
- ✓ About me
- ✓ Story
- ✓ Profil
- ✓ Skills
- ✓ Portfolio
- ✓ Last projects
- ✓ Contact

### Step 01

---

Create an HTML page `index.html` and put the **jQuery** and **Materialize**'s CDNs inside. Apply a light grey background color to the page. Add a div as wide as the page, with a color of your choice as background. Create 2 titles:

- ✓ one that contains your name in white ;
- ✓ one that contains your profile heading in grey with a white background.

### Step 02

---

Create a container div that contains a row with "About me" as a title. In this row, put a first div which must be:

- ✓ 12 columns large for a small sized screen ;
- ✓ 4 columns large for a medium to large screen.

Add to it a "Story" title, and a quick explanation about your professional background.

### Step 03

Inside the “About me” row, make a div of the same size as the “Story” one, with “Profil” as title. It’s background color is the one you chose before.

Make a card whit white inner text and a responsive round profile picture next to it.
After this card, add a div `card-action` containing links to your **Github** and **Linkedin** profiles.

### Step 04

The last div in the “About me” row is titled “Skills” and has the same size as the 2 previous divs. Inside it, there is a table with a grey row every 2 rows.

This table contains the languages, frameworks you know with a small comment for each one.

### Step 05

Create new rows, with “Portfolio” and “Last projects” as titles. They must contain 3 pictures each.

### Step 06

Create a row titled “Contact”. Make a new row inside it, containing a full column sized form.

In that form, make a first row, with 2 inputs: “Lastname” and “Firstname”. Each one must be half the row size.

Make a second row containing a full column sized email input. The last row will contain a full column sized text area, with a “Your message” placeholder.

Titles and buttons must be centered.

v 4.2