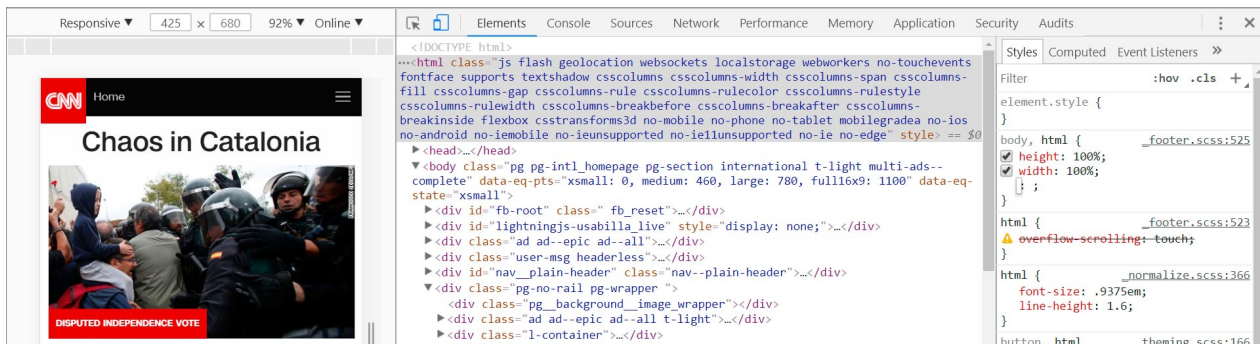


## Exercise 4 – Responsive Web Design (RWD)



### Requirements:

This assignment is a training in constructing a responsive website.

You must grab content from an online news site like CNN or Boston Globe and migrate it into your project.

Choose several articles (minimum 3) and deploy them into a semantic html5 outline.

Style all content.

Then your task is to apply the techniques of Responsive Web Design to your project.

Finally, - upload to your webhost, validate all code and test for performance.

Hand-in an URL to Canvas.

Deadline: Before startup at next lecture 8:30 am

**The main challenge is NOT to make a 1:1 visual representation of CNN or Boston Globe. The main challenge is training the techniques that make up RWD!**

### Planning:

1. Make a sketch on a piece of paper of your planned project (page).
2. Functional template:  
Based on your sketch, you can now construct a template for your website.  
Use Css Layout Generator. Link: <http://csslayoutgenerator.com/>  
Template Requirements: HTML5, reset.css (Eric Meyer), fluid layout, no columns.

### Html:

3. The task now is to include all data in an HTML5 document. The page should have title, headings and semantic tags that describe the content of the page and "outline" (according to your sketch) in the best way.  
In addition, of course, other tags may be used where found appropriate.
4. Apply Viewport:  
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
5. Validate the html. <http://validator.w3.org/>

### CSS:

6. The task is now to style the page. Start with a narrow viewport of 240px. You have free hands when it comes to the visual appearance, but you must show an ability to be consistent.
7. RWD: All layout should be fluid and all images must be flexible.  
Fluid Layout:
  - all width, margin and padding properties in % (or rem's, em's - not px)Flexible Images and Media:
  - all images scaled in % (not px)
8. Test the page on a smartphone or in a mobile simulator. You will find links to test simulators on Canvas in the Device Emulators folder.
9. You have now implemented a layout for mobile 😊  
The task now is to make your solution "responsive" so that it also works on tablets and laptops.
10. Media Queries: Insert breakpoints: 320px, 480px, 768px and 1024px. You may add more or less breakpoints if your layout requires it. (Use your eyes!)  
Hint: Apply different background-color, i.e. the header-element, as you develop so that you can visually follow your breakpoints when testing your layout.
11. Validate html: Markup Validation Service: <http://validator.w3.org/>  
Validate css: CSS Validation Service: <https://jigsaw.w3.org/css-validator/>

#### Implementation:

12. Upload the solution to your web host. Test the solution.
13. Test page performance: With throttling set to "Slow 3G", the goal is to keep Page Load Time in less than 10 sec.!

#### Deadlines:

14. Upload a link to your online solution to Canvas (Opgaver -> Coding and Programming: Exercise 3).  
Deadline: Before startup at next lecture 8:30 am