

HTML Recap

CISC-2350-R01 | Fall 2017 | Week 4

Ruta Kruliauskaite

Today's Agenda

- Attendance
- Review: HTML Forms & Inputs
- Review homework reading & questions
- Homework show & tell
- HTML recap
- Intro to Developer Tools & in-class exercise
- Intro to midterm assignment
- Homework assignment

Class reschedule: 10/2 -> 10/5

<https://beta.doodle.com/poll/y582ne4fq2gs4xa3#table>

Review: HTML Forms and Inputs

Forms

- Allows to collect information from visitors to your site
- Can include text input, selectors, buttons, upload files, drop-down boxes, and more
- Form examples:
 - search boxes
 - shopping online
 - registering for newsletters, etc.
- These end up being heavily integrated with JavaScript and server-side programming... but not in this class.

ADDING TEXT:

Text input (single-line)

Used for a single line of text such as email addresses and names.

A single-line text input field with a light gray border and a small cursor icon on the left.

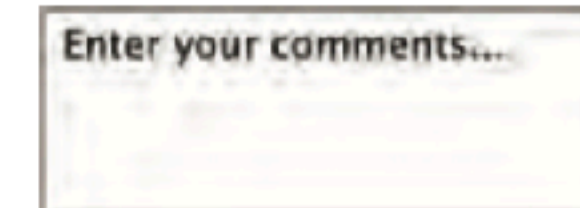
Password input

Like a single line text box but it masks the characters entered.

A password input field with a light gray border and a series of small dots representing masked characters.

Text area (multi-line)

For longer areas of text, such as messages and comments.

A multi-line text area with a light gray border and a placeholder text "Enter your comments..." at the top.

MAKING CHOICES:

Radio buttons

For use when a user must select one of a number of options.

Three radio buttons labeled "Rock", "Pop", and "Jazz". The "Rock" button is selected, indicated by a small blue dot.

Checkboxes

When a user can select and unselect one or more options.

Three checkboxes labeled "iTunes", "Last.fm", and "Spotify". The "iTunes" checkbox is checked, indicated by a small blue square.

Drop-down boxes

When a user must pick one of a number of options from a list.

A drop-down box with the text "iPod" and a small arrow icon on the right.

SUBMITTING FORMS:

Submit buttons

To submit data from your form to another web page.

A submit button with the text "Subscribe" in a rounded rectangular button.

Image buttons

Similar to submit buttons but they allow you to use an image.

An image button with the text "SUBSCRIBE" in a dark green rectangular button.

UPLOADING FILES:

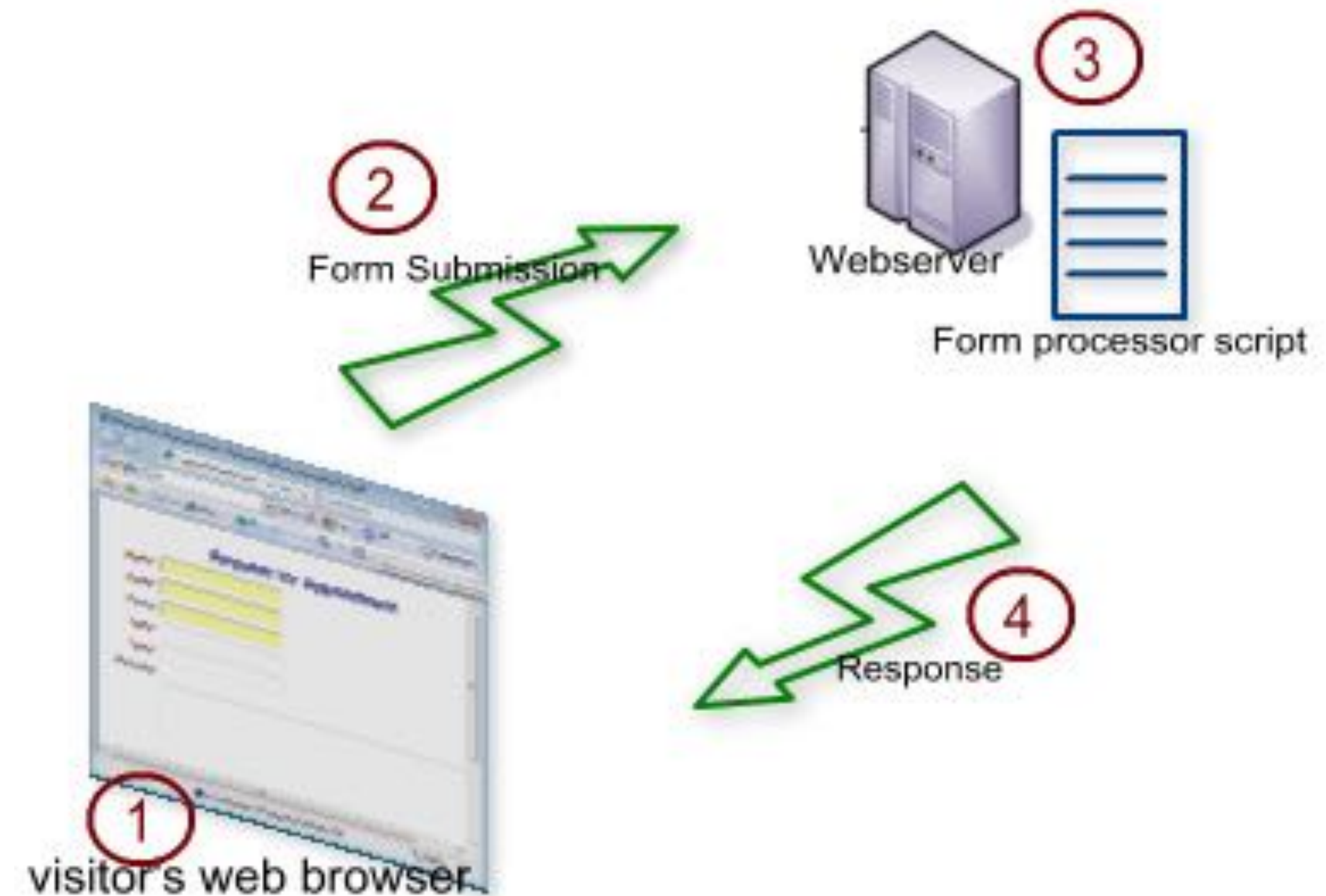
File upload

Allows users to upload files (e.g. images) to a website.

A file upload form with an "Upload" button and a "Browse..." button.

How forms work

- User fills in the form and submits information
- The name of the form and value submitted is being sent to the server
- Server processes information, saves it in the database
- Server sends back information by creating a new page in the browser



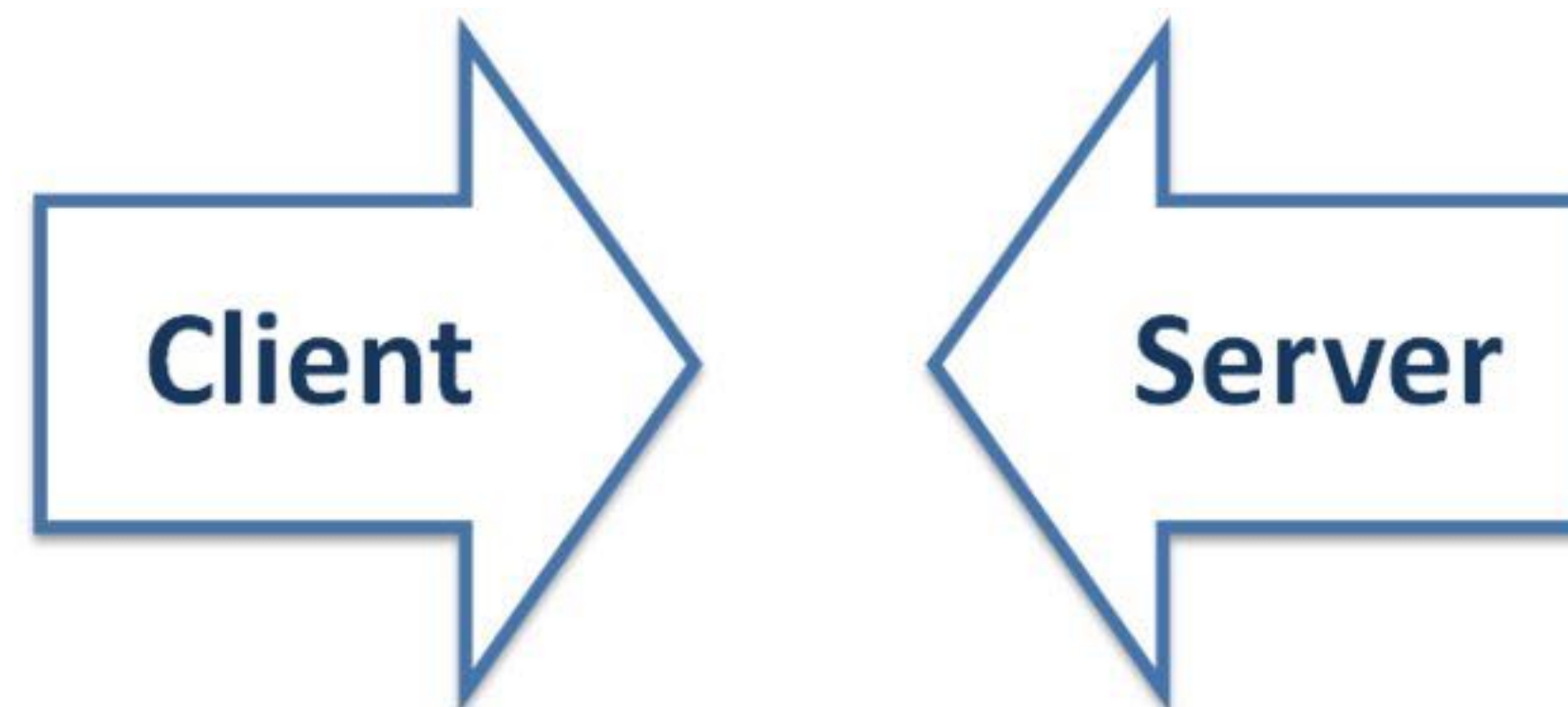
Client vs. server

Client-side programming mostly deals with the user interface with which the user interacts

in the web - what we see in the browser (HTML, CSS, JavaScript)

Server-side programming is the program that runs directly on a server

- whenever there is some user interaction needed / website is dynamic, there is some server-side programming included



Server-side programming

Usage examples:

- creating, reading, updating databases
- saving any data into database
- creating and saving user profiles

Server-side programming languages:

- JavaScript: nodejs, expressJS (routing, etc.)
- PHP
- C#, C++
- Java
- Python
- Ruby on Rails, etc.

You can read more documentation about server-side programming [here](#).

Questions from Slack

1. What is the difference between "datetime-local" and "time" input types?
2. How can you mix inputs with styling to place them on different spots of your website?
3. How is the email address automatically validated?
4. How would a comment submitted in a form change the html code of the webpage? Would it just add another <p> element below?
5. Once the information is submitted in a form, where does that information go? Does it alter the HTML code in any way?
6. Can you use C++ and HTML to submit answers from html and get an output from a user created program in C++?
7. Github file upload size: max 100MB, can resize video using: <https://video.online-convert.com/convert-to-mp4>
8. Other questions?

Homework show & tell

<http://bit.ly/2xUmxl5>

HTML recap

HTML



CSS



JS



Web Page

```
graph TD; WP[Web Page] --- HTML[HTML Content & Structure]; WP --- CSS[CSS Presentation]; WP --- JS[JavaScript Behavior]; HTML --- HTMLList[Headings, Paragraphs, Lists]; CSS --- CSSList[Font, Color, Background color, Border]; JS --- JSList[dynamic display, widgets, user interaction, click to open a popup];
```

HTML Content & Structure

Headings,
Paragraphs
Lists

CSS Presentation

Font
Color
Background color
Border

JavaScript Behavior

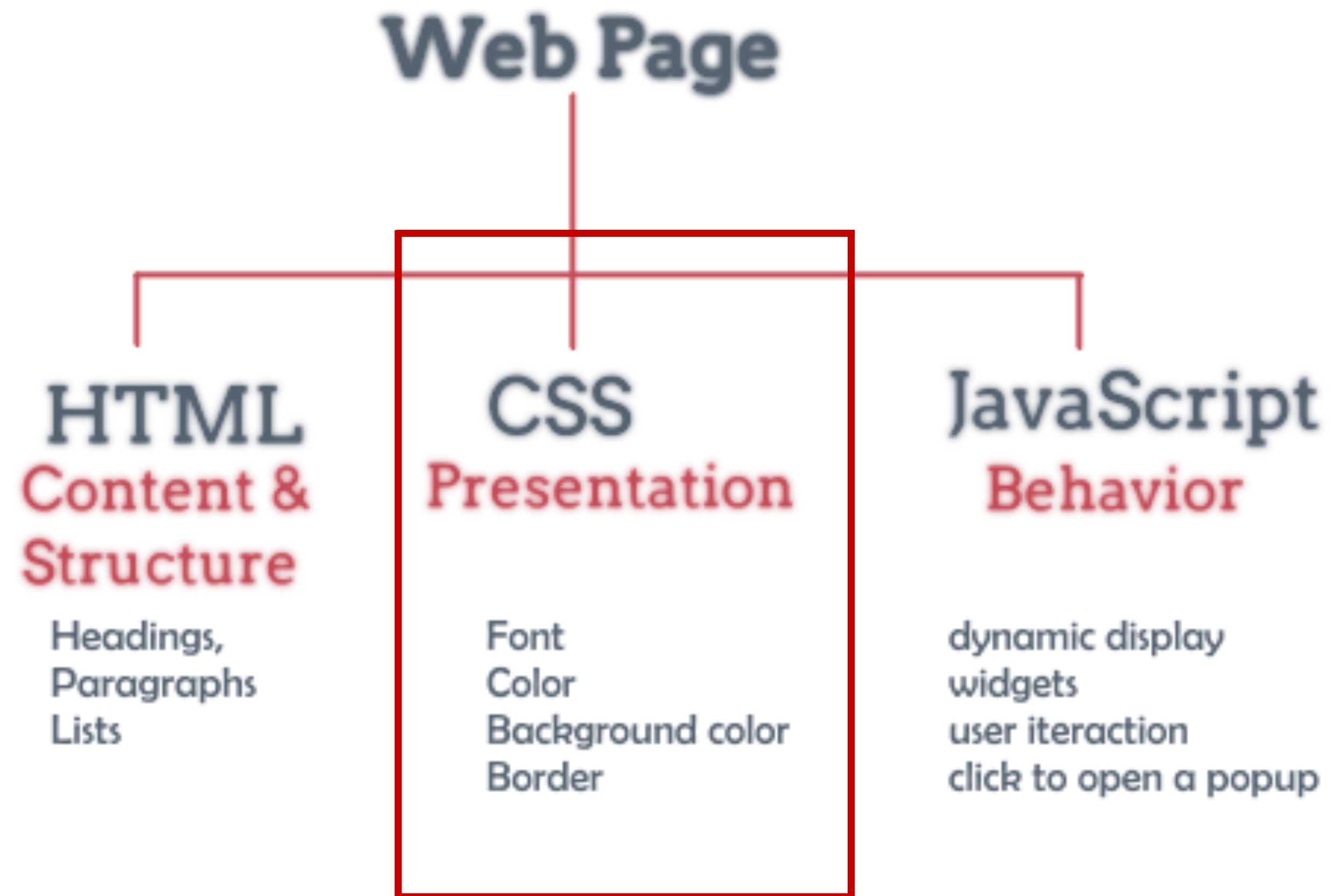
dynamic display
widgets
user interaction
click to open a popup

Topics covered

1. HTML Structure
2. HTML tags: text, lists, links, images
3. HTML Media Elements: iframes, video, audio
4. HTML Tables
5. HTML Forms & Inputs
6. File structure & multiple HTML pages
7. Github & Github Pages

More documentation on **W3 Schools** and **Mozilla**.

Next week

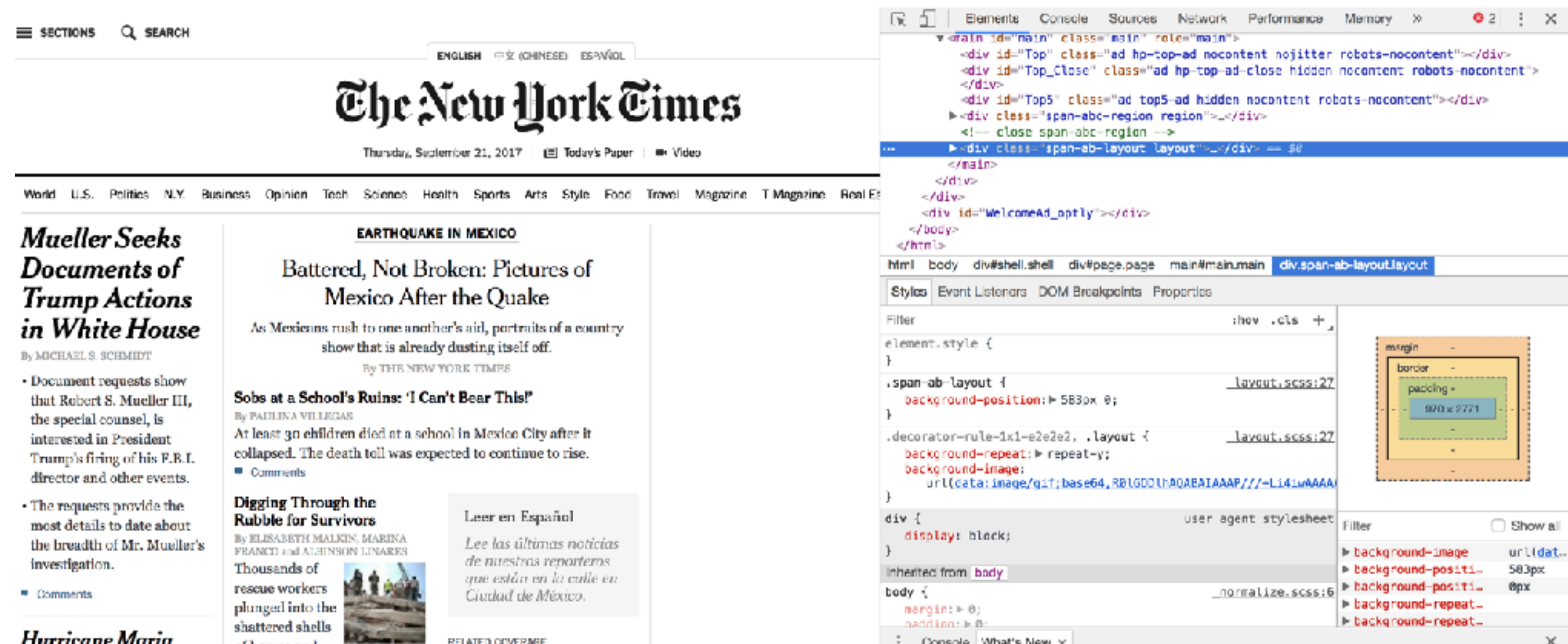


Intro to Developer tools

Developer Tools

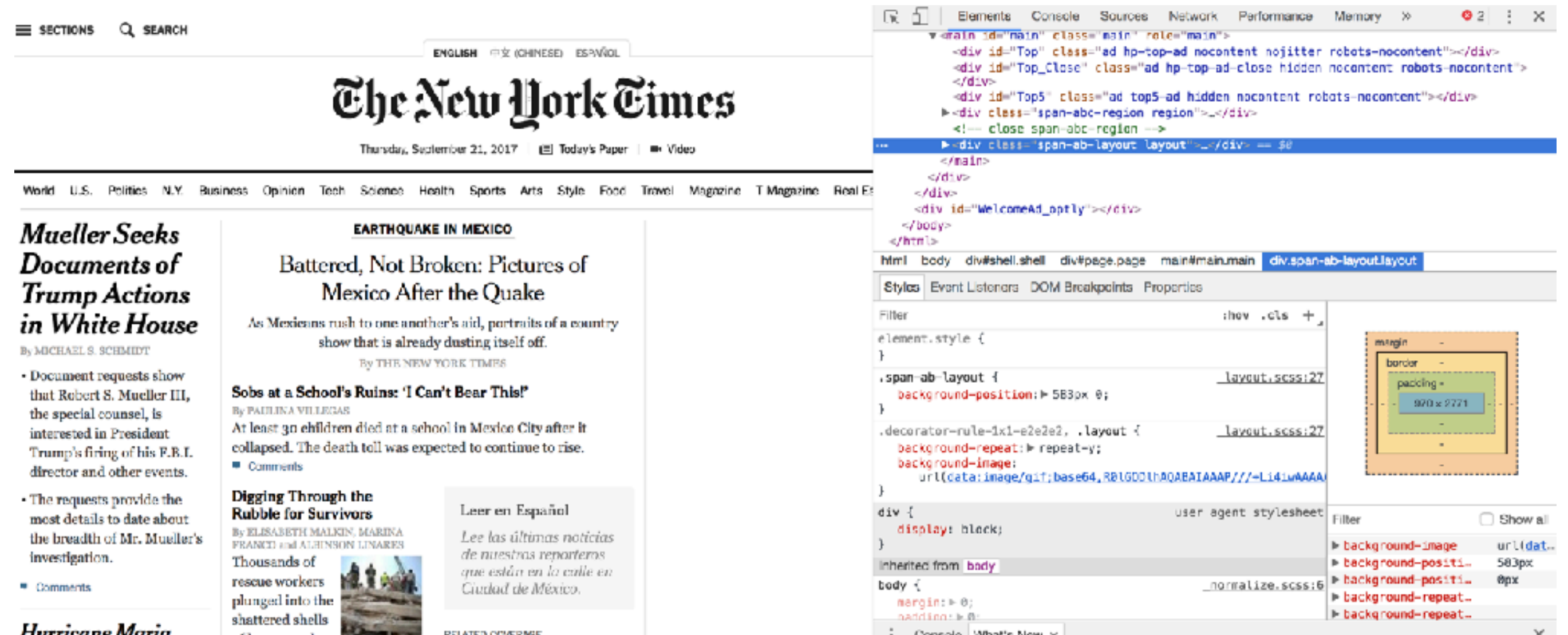
Tools that provide access to website debugging:

- to see / inspect which assets the web has (HTML, CSS, JavaScript)
- layout issues
- how long they take to load
- lets to modify the page



Accessing Developer Tools

- Chrome menu: View -> Developer -> Developer Tools
- Right click on the page -> Inspect
- Cmd + opt + I



Developer Tools Window

The screenshot displays a web browser interface with the Chrome Developer Tools window open. The browser shows a news article titled "Mueller Seeks Documents on Trump Action in White House" by Michael S. Schmidt. The Developer Tools window is open to the "Elements" tab, showing the HTML structure of the page. The "Console" tab is also visible, showing a message about mobile device throttling. The "Network" tab is open, showing a list of network requests. The "Styles" pane on the right shows the CSS rules for the selected element, including a background image and font settings. The "Pseudo" pane shows the default styles for the pseudo-element.

Elements

```
<#shadow-root (open)>
  <!--<![endif]>-->
  <!--[if IE 9]> <html lang="en" class="no-js ie9 lt-ie10 edition-domestic app-homepage"
  xmlns:og="http://opengraphprotocol.org/schema/"> <![endif]>-->
  <!--[if IE 8]> <html lang="en" class="no-js ie8 lt-ie10 lt-ie9 edition-domestic app-homepage"
  xmlns:og="http://opengraphprotocol.org/schema/"> <![endif]>-->
  <!--[if (lt IE 8)]> <html lang="en" class="no-js lt-ie10 lt-ie9 lt-ie8 edition-domestic app-homepage"
  xmlns:og="http://opengraphprotocol.org/schema/"> <![endif]>-->
  <head>...</head>
  <body>
    <style>...</style>
    <div id="suggestions" class="suggestions messenger nocontent robots-nocontent" style="display:none;">...</div>
    <div id="shell" class="shell">
      <header id="masthead" class="masthead theme-pinned-masthead" role="banner">...</header>
      <!-- close masthead -->
      <div id="masthead-placeholder" class="masthead-placeholder"></div>
      <nav id="navigation" class="navigation">...</nav>
      <!-- close navigation -->
      <nav id="mobile-navigation" class="mobile-navigation hidden">...</nav>
      <!-- close mobile-navigation -->
      <div id="navigation-edge" class="navigation-edge"></div>
      <div id="page" class="page">
        <main id="main" class="main" role="main">
          <div id="Top" class="ad hp-top-ad nocontent nojitter robots-nocontent"></div>
          <div id="Top_Close" class="ad hp-top-ad-close hidden nocontent robots-nocontent"></div>
          <div id="Top5" class="ad top5-ad hidden nocontent robots-nocontent"></div>
          <div class="span-abc-region region">...</div>
          <!-- close span-abc-region -->
          <div class="span-ab-layout layout">...</div>
        </main>
      </div>
      <div id="WelcomeAd_optly"></div>
    </body>
  </html>
```

Styles

```
Filter
element.style {
}
.span-ab-layout {
  background-position: 583px 0;
}
.decorator-rule-1x1-e2e2e2, .layout {
  background-repeat: repeat-y;
  background-image: url(data:image/gif;base64,R0lGODlhAQABAIAAAP///+Li4iwAAAAAQABA);
}
div {
  display: block;
}
Inherited from body
body {
  margin: 0;
  padding: 0;
  font-family: "nyt-cheltenham", georgia, "times new roman", times, serif;
  color: #333;
  background-color: #fff;
}
Inherited from html.flag-limitFabrikSave-on.flag-moreFollowSuggestions-
html {
  -moz-text-size-adjust: 100%;
  -webkit-text-size-adjust: 100%;
  -ms-text-size-adjust: 100%;
  font-size: 100%;
}
Pseudo ::before element
.wrap:before, .clearfix:before, .control:before, _mixins-layout.scss:
```

Console

Mobile device throttling
Simulate a mobile device's CPU and network throttling from Device Mode.

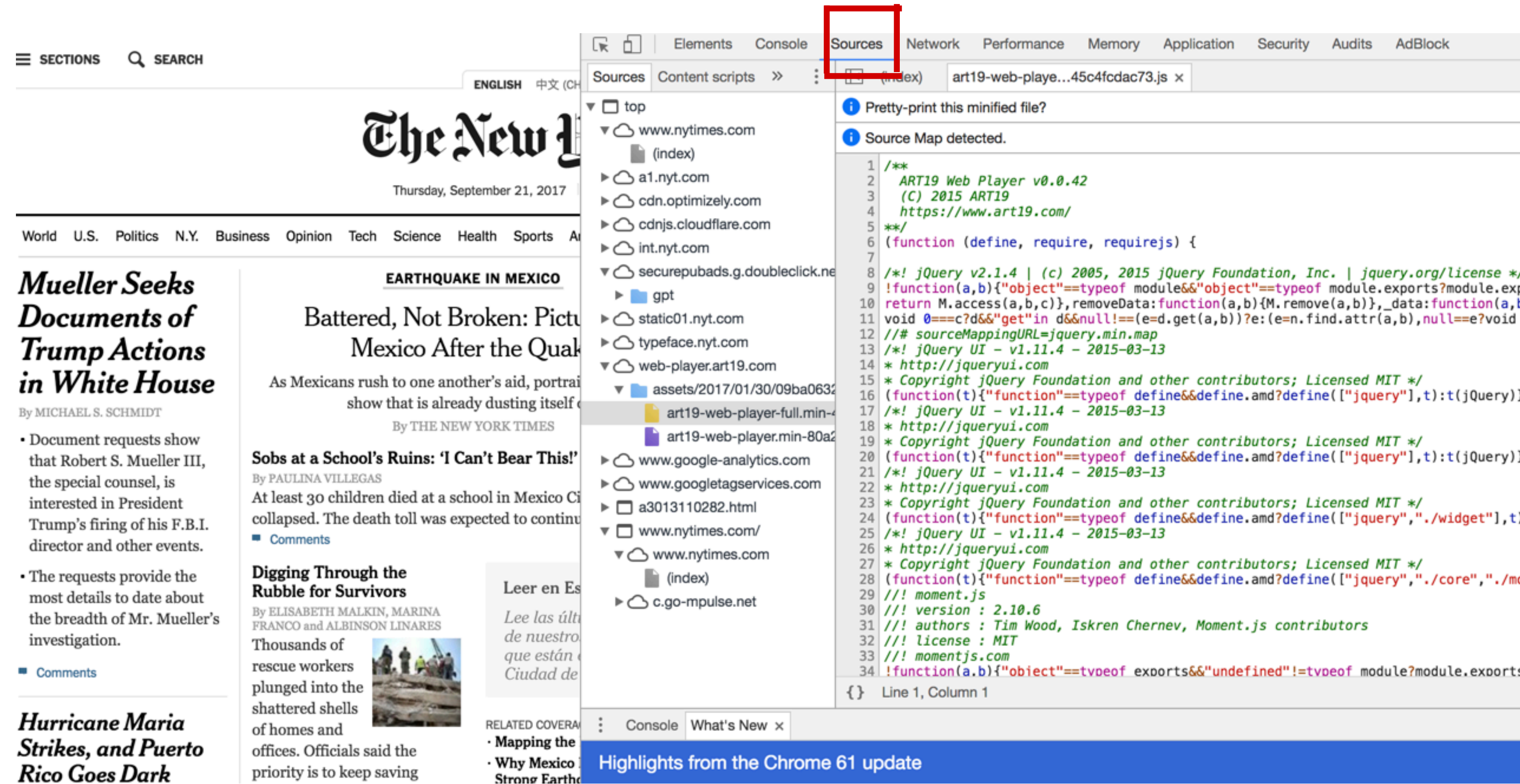
Dev Tools: Elements

- Layout and elements of the page
- Live edit of HTML, CSS

The screenshot displays the Chrome DevTools interface with the 'Elements' panel open. The left pane shows a news article snippet with headlines like 'Mueller Seeks Documents on Trump Action in White House' and 'Hurricane Maria Strikes, and Puerto Rico Goes Dark'. The middle pane shows the DOM tree with the 'div#page' element selected, revealing its nested structure including 'main', 'div#Top', and 'div#Top5'. The right pane shows the 'Styles' panel for the selected 'div#page' element, displaying various CSS rules such as 'margin: 0;', 'padding: 0;', 'font-family: "nyt-cheltenham", georgia, times, serif;', 'color: #333;', and 'background-color: #fff;'. The 'Pseudo ::before element' section is also visible at the bottom of the styles panel.

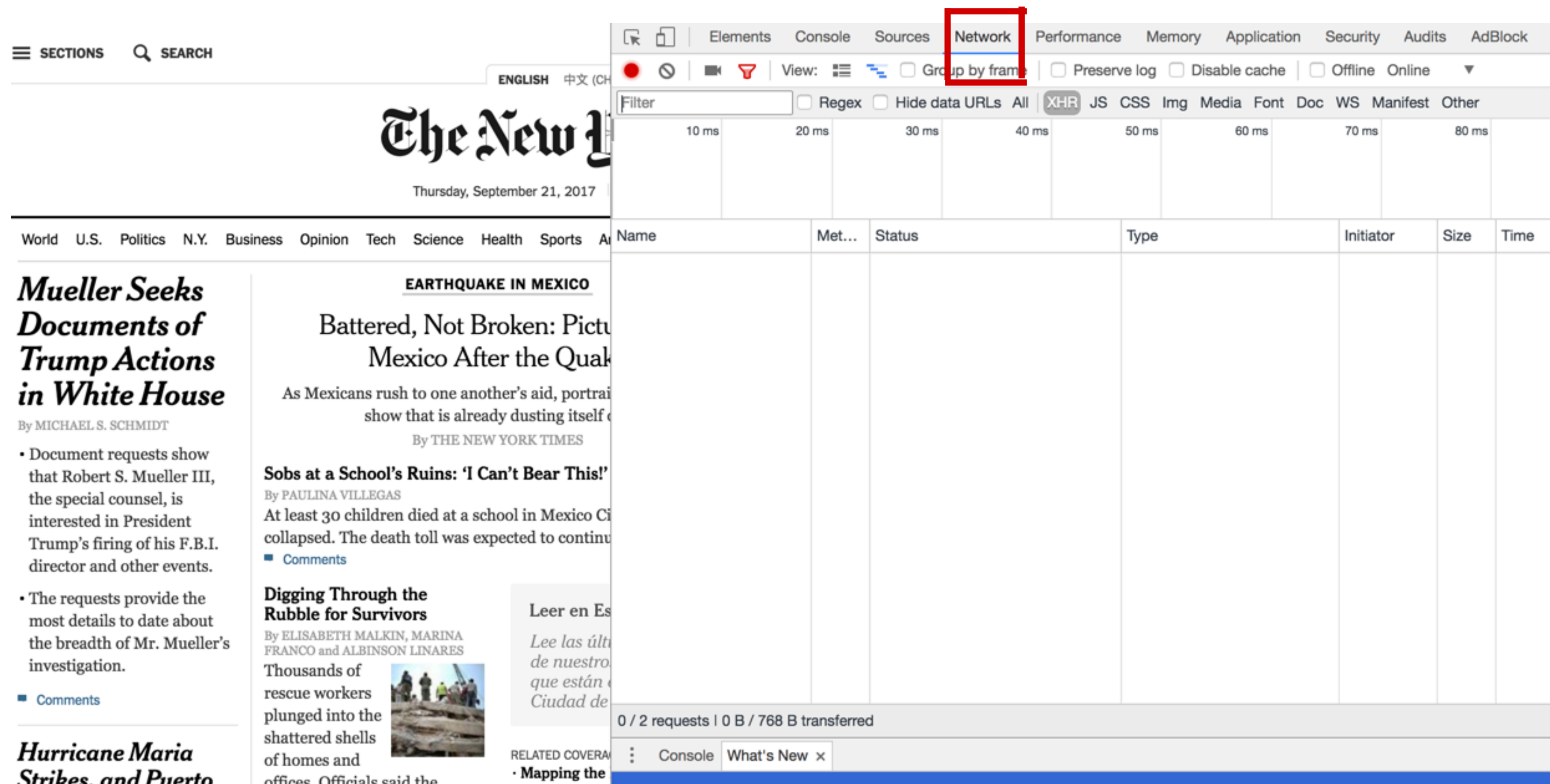
Dev Tools: Sources

- Shows sources of the page



Dev Tools: Network

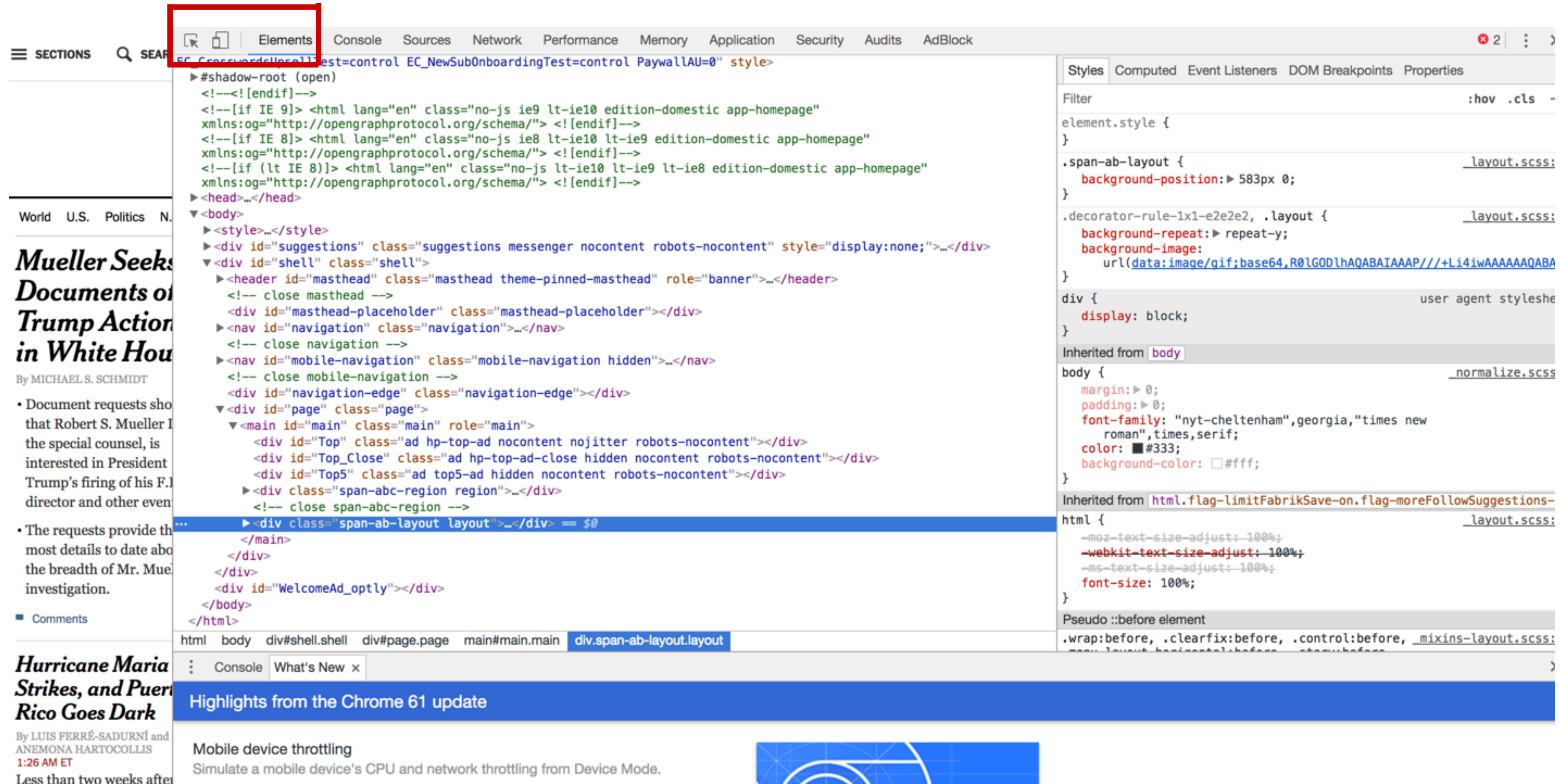
- Measures network performance



More documentation:

[https://developers.google.com/
web/tools/chrome-devtools/](https://developers.google.com/web/tools/chrome-devtools/)

Dev Tools: Elements



Demo

In-class exercise

In-class exercise:

Modify NYTimes page:

1. Access different HTML elements:
 - Change content
 - Try modifying style

Introduction to midterm

Midterm Assignment (due 10/12)

Your midterm project will be a multi-page (at least 3) website responding to one of the following assignment prompts:

- An conceptual, interactive self portrait. Make a digital interpretation of a portrait that has come before you, from painting to photography. This project is not intended to be a portfolio but some kind of creative representation of yourself and who you are that cannot be experienced in another way.
- A piece of hypertext narrative or art, similar to one of these examples <https://rhizome.org/art/artbase/collections/collection-hypertext/>
- Invent a fictional or futuristic product (a machine that could record your dreams, a sneaker that plays music, get creative!) and create a product website for it. Look here for some inspiration <http://www.webdesign-inspiration.com/web-designs/type/product>

It's an individual project.

Midterm Assignment 2

Your midterm project will be a multi-page (at least 3) website responding to one of the 3 assignment prompts (or one of your own, approved by me).

You will give a five minute presentation on your midterm to the class, explaining your idea, your code, your visual choices and your process. (More questions to guide your presentation will be provided).

You will be graded according to a rubric that I will distribute next week on concept, execution, visual design and effective communication.

Midterm presentations: Thursday, October 12. There might be guest critics.

Midterm Assignment 3

Your midterm project will include the following:

- Must be posted to Github pages in proper file structure.
- Your code should be neat and well organized, as well as well commented out
- The HTML and CSS code is efficient (no duplications), with CSS in an external style.css file

Required web elements include:

- Minimum of three HTML pages
- Media elements (video, audio, iframes)
- Appropriate usage of HTML elements: text (h, p, ul, ol, li), lists, links, images, div, span
- Use of Classes and IDs in your HTML and as CSS selectors
- CSS color selections
- CSS font selections

Homework assignment

Homework

Due Sunday by 6pm

- 1. Start thinking about your midterm concept - we'll discuss next week**
- 2. Read:**
 - Documentation on developer tools: <https://developer.chrome.com/devtools>
- 3. Monday web review presentation - Jenna & Salvatore (submit link to your site by Sunday 6pm)**
- 4. Using developer tools modify one of the following websites:**
 1. The New York Times: <https://www.nytimes.com/>.
 2. Mashable: <http://mashable.com/>
 3. Wired: <https://www.wired.com/>
 4. Or your own choice page
- 5. Do the following:**
 1. Access existing HTML elements through Elements window and change the content in them:
 1. Text in headings, paragraphs
 2. Change images src
 2. Access existing HTML elements and also try modifying the style
 3. Try experimenting what else you can do (e.g. can you add additional HTML elements?)
- 6. Make two screenshots: one of original page and the second one of your creative creation and upload to Github by Sunday 6pm (create new folder for this week)**
- 7. Post link to it on #general channel on Slack by Sunday 6pm.**
- 8. Office hours:** today 6:45-7:45pm. Sign up [here](#).