

# Introduction to HTML & CSS

Conflict Urbanism: InfraPolitics

Tutorial 8

Oct 31 & Nov 3, 2017

# By the end of this workshop

- Set up a Case Study for your project
- Edit an HTML Template to add your own content
- Manually create links and objects in a website
- Utilize Elements, Tags, and Attributes to add functionality and navigation to a site
- Navigate Selectors, Properties, and Values to add style to a site

CONTENT  
+ STRUCTURE (HTML)  
+ STYLE (CSS)  
+ FUNCTIONALITY (JS)

-----

A WEBSITE

# Web Basics

- Every site uses at least 2 languages:
  - HyperText Markup Language
  - Cascading Style Sheets
- Some use a 3<sup>rd</sup>:
  - JavaScript
- Modern Browsers "read" these languages to display content

# HTML

- **ELEMENTS** define the structure of a page
  - headings <h1>
  - paragraphs <p>
  - define structure <html>

# HTML

- **ELEMENTS** define the structure of a page
  - headings <h1>
  - paragraphs <p>
  - define structure <html>
- ELEMENTS are defined with **TAGS**
  - opening tag <p>
  - closing tag </p>

# HTML

- **ELEMENTS** define the structure of a page
  - headings `<h1>`
  - paragraphs `<p>`
  - define structure `<html>`
- ELEMENTS are defined with **TAGS**
  - opening tag `<p>`
  - closing tag `</p>`
- **ATTRIBUTES** give more information about ELEMENTS
  - id
  - class
  - src
  - href

# HTML

“BEGIN ITALICS”

THE TEXT

“END ITALICS”

`<i>text</i>`

## WHAT APPEARS ON THE SCREEN



*text*



# HTML Document Object Model (DOM) **Nested**

- Plain text document saved with .html suffix
- Every document needs:

<!DOCTYPE html>

<html>

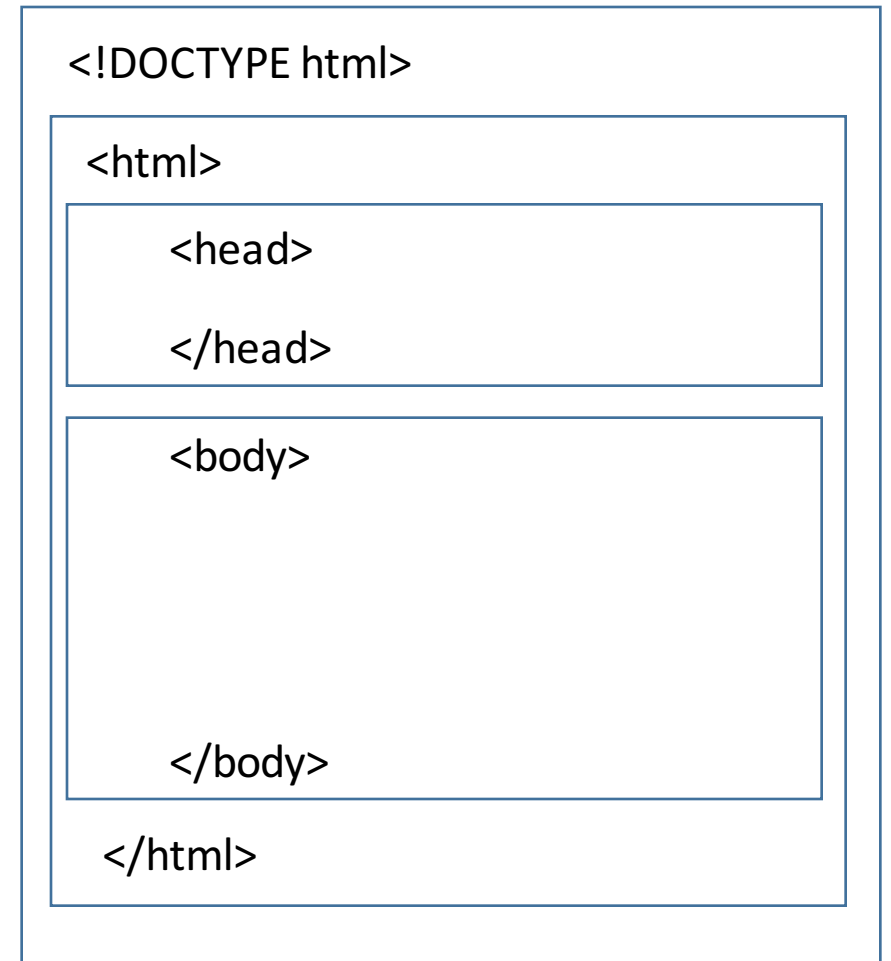
<head>

</head>

<body>

</body>

</html>



# Typical header

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
  <head>
```

```
    <meta charset="utf-8">
```

```
    <title>Hello World</title>
```

```
  </head>
```

```
  <body>
```

```
    <h1>Hello World</h1>
```

```
    <p>This is a web page.</p>
```

```
  </body>
```

```
</html>
```

# CSS: SELECTORS

- General (all paragraphs get Lobster font)
- Specific (the id = Arabic paragraph gets Colvert font)
- Followed by { }

```
p {  
  margin: 0;  
  font-family: 'Lobster';  
  color: 'red';  
}
```



NOTE THE SEMICOLONS AT  
THE END OF EVERY LINE

# CSS: PROPERTIES

- Properties
  - font-size
  - font-family
  - color
- border
- background-color
- padding
- margin



# CSS: VALUES

- Specify the behavior of a property

# TYPE SELECTORS

- Select all the elements by their type (i.e., all div's)

In the CSS file:

```
div {property: value;  
    property: value;  
}
```

In the HTML file:

```
<div> ... great content... </div>  
<div> ... greater content... </div>
```

# CLASS SELECTORS

- More specific than type selectors
- Select based on class attribute value (group of elements)

In the CSS file:

```
.bantu {property: value;  
        property: value;  
}
```

In the HTML file:

```
<div class = "bantu"> ... weebale... </div>  
<p class = "bantu"> ... oli otya... </div>
```

# ID SELECTORS

- More precise than class selectors
- One element at a time

In the CSS file:

```
#nilotic {property: value;  
          property: value;  
          }
```

In the HTML file:

```
<div id = "nilotic"> ... yoga noi... </div>  
<p id = "nilotic"> ... yoga swam... </div>
```



# HTML + CSS

- Save the CSS file in the SAME FOLDER as the .html file
- Each HTML file should only have one CSS file (we will break this rule soon)
- Point the HTML document to read the CSS document in the <head>

<head>

    <link rel =“stylesheet” href=“main.css”>

</head>

# Notes to Self

- Commenting code

## HTML

```
<!-- this does that -->
```

## CSS

```
/* this looks like that */
```

# The Case Study

# Folder Structure

- Everything natively in your site must be in the same folder (i.e., text, images (incl. maps), reports, audio, etc.).
- The only exception is things you point to on the internet (i.e., YouTube Videos, map tiles, etc.).
- The Case\_Study.zip folder has the HTML and CSS documents

# Open the files

- Open the .html and the .css files with a text editor:
  - Sublime
  - Text Wrangler
  - Another text editor of your choice (not MS Word or Google Docs)
- Open the .html file with Google Chrome
  - If Chrome is not your default, 'right click' or 'command+click'

# Understand the Document

- Header
  - Metadata
  - Link to bootstrap
  - Link to css
- Body
  - Div
  - Case Study Title
  - Internal link structure
  - Embedding maps, videos, documents, plotly, etc.
  - Footnotes

# Inspect the HTML

- View > Developer > Developer Tools

## TO DO

- Change project title (39)
- Make another Section (55)
- Add an image (66)
- Embed a map you made (81)
- Update a Footnote (63 & 96)



# Header

```
<!DOCTYPE html><html lang="en">  
<head>
```

Add: Project Keyword Description & Name

```
<meta name="description" content="">  
<meta name="author" content="">
```

# Body - Introduction

Add: Case Study Title

```
<h5>Case Study | Type Title Here </h5>
```

-Add: Image Header: File Size: 1800x450.png.

Save your file as 'Header\_image\_1800x450.png' in the <img> folder. Use Illustrator Template in folder if you require sizing

```

```

# Body - Content

- Add to and change the sections as needed
- Text w PDF
- Text w Image
- Text w Video
- Text w Interactive
- Text w Map

# Citations

- Footnotes
- `<div class="footnotes">`

# Complete the Tutorial

- Edit the tutorial with:
  - Name
  - Tentative Project title
  - Tentative 1-2 sentence description

# Cascading SS

1. Look in the style sheet. If there are no defined styles, use browser default
2. Find selectors marked **!important** and apply those and don't change.
3. The more specific the style selector, the higher the precedence it has
  1. `<p id="swahili">` is more specific than
  2. `<p class="bantu">` is more specific than
  3. `<p>`
4. If two rules apply to the same element and have the same rank, the one that was loaded **last** will be applied.
  1. The style sheet is read from top to bottom
  2. styles are applied on top of one another.