<Welcome>

Before we begin please ensure that:

1. Atom text editor has been installed

   ATOM

2. Joined the Taste of Code Slack group

   slack

   tasteofcode.slack.com

3. Installed the Google Chrome browser

   chrome

Wifi: Guest@Rockstart   Password: startupsrock!

#TasteOfCode

Intro to programming

:{) Codaisseur rockstart. le comptoir de l'innovation J.P.Morgan

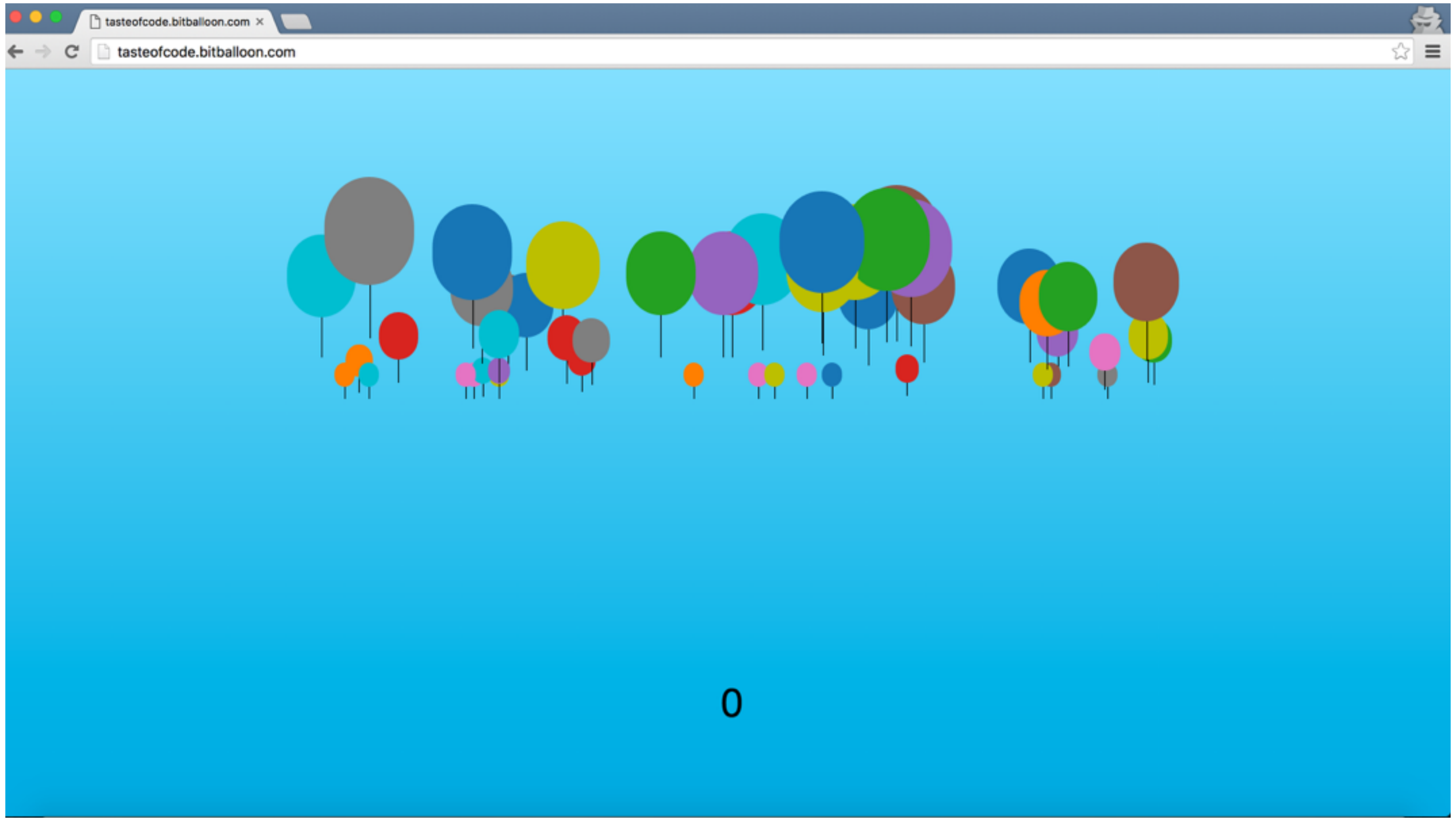
Today’s objective

:{) Codaisseur

Today's objective

:{) Codaisseur

Build an online game

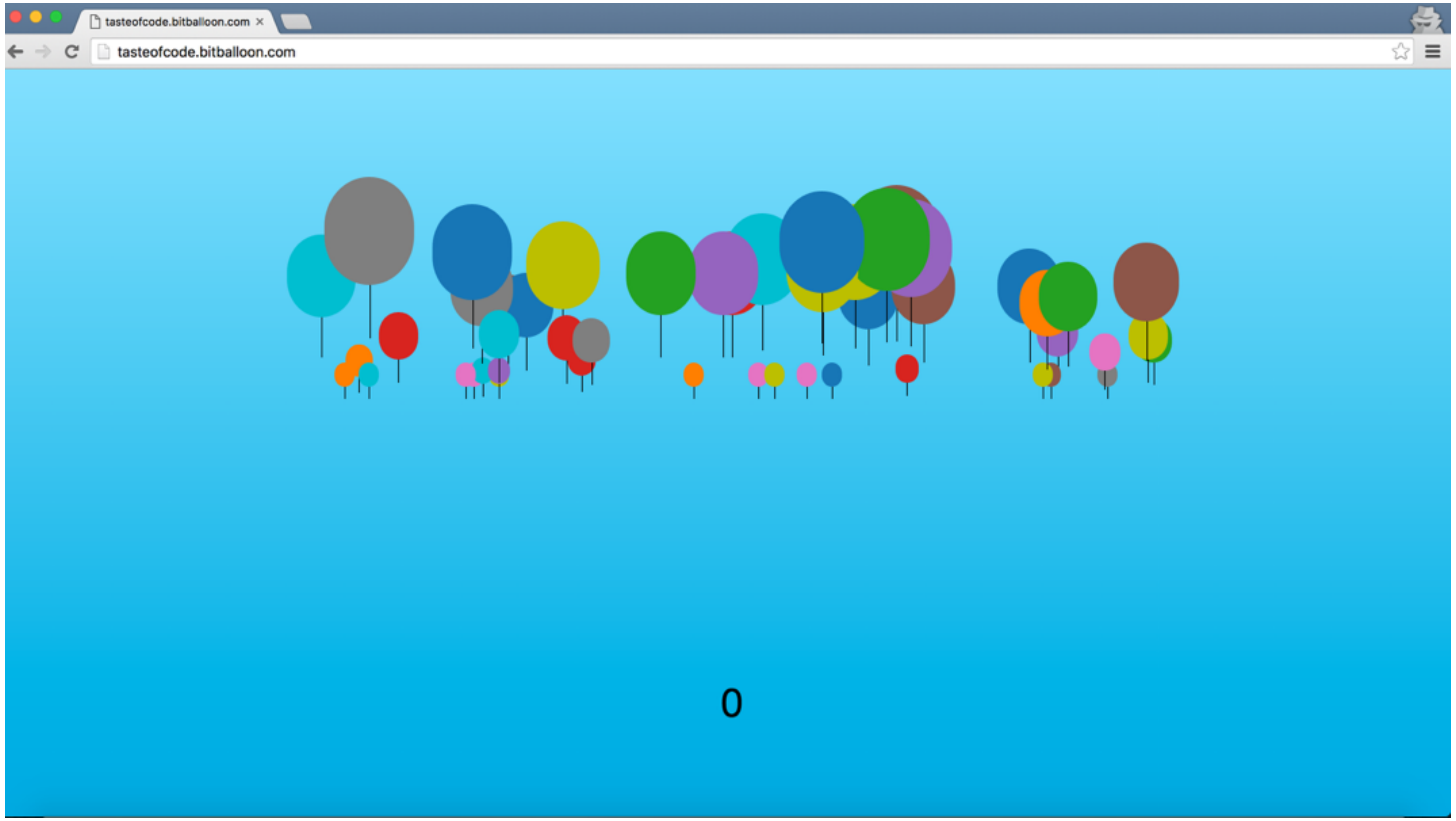


Today’s objective

:{) Codaisseur

Build an online game

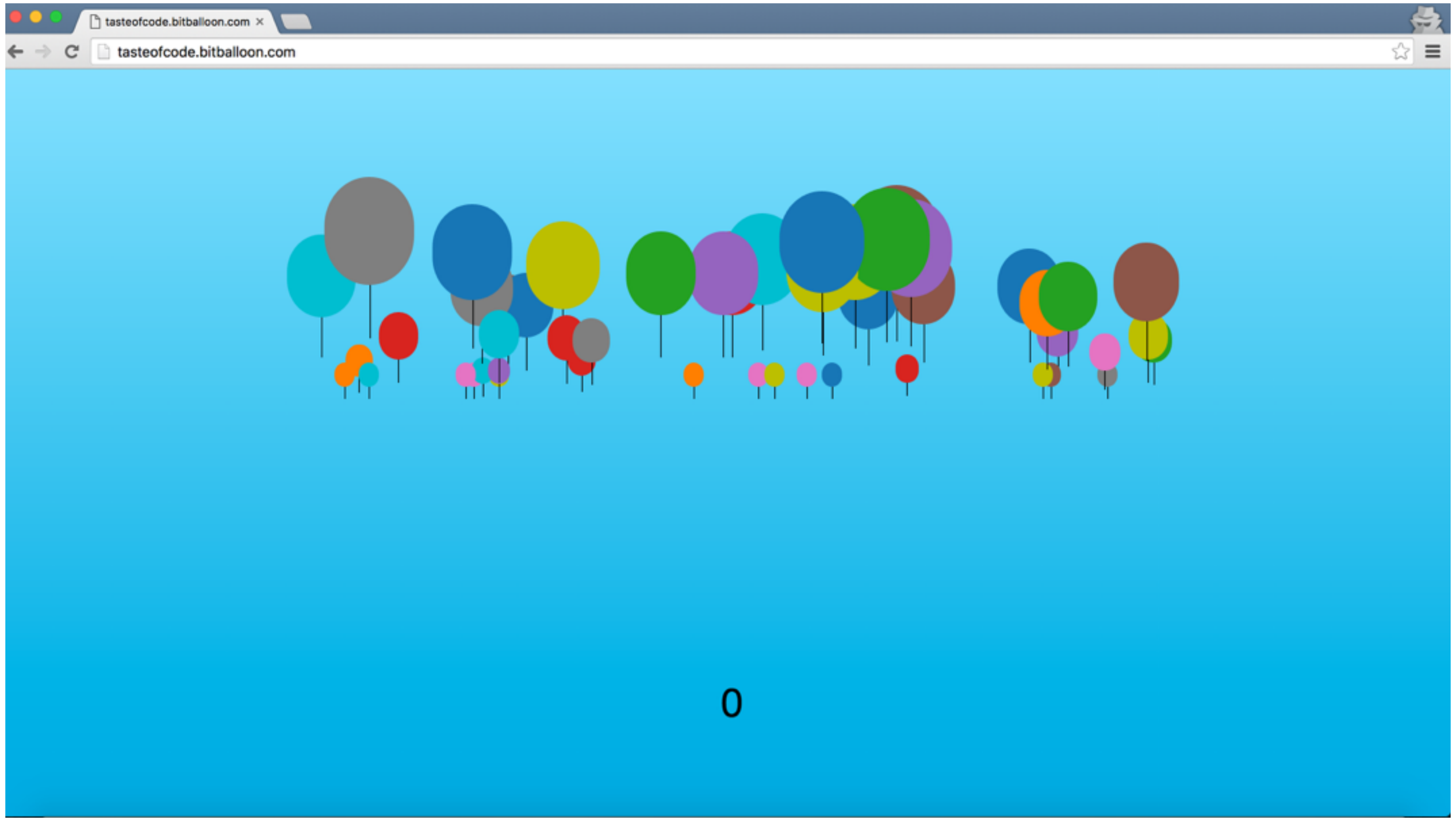
- HTML



Today's objective

Build an online game

- HTML
- CSS

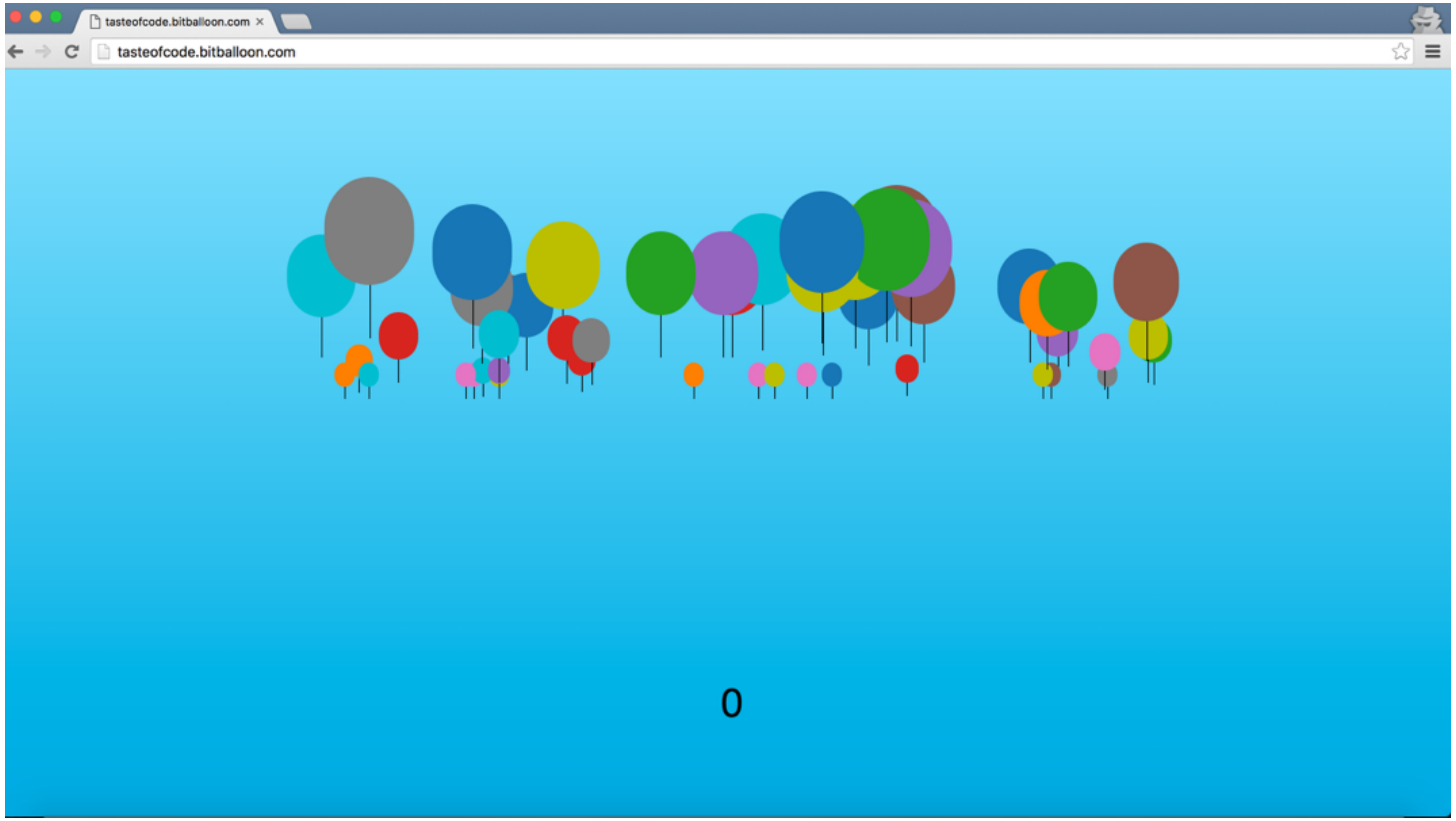


Today's objective

:{) Codaisseur

Build an online game

- HTML
- CSS
- jQuery



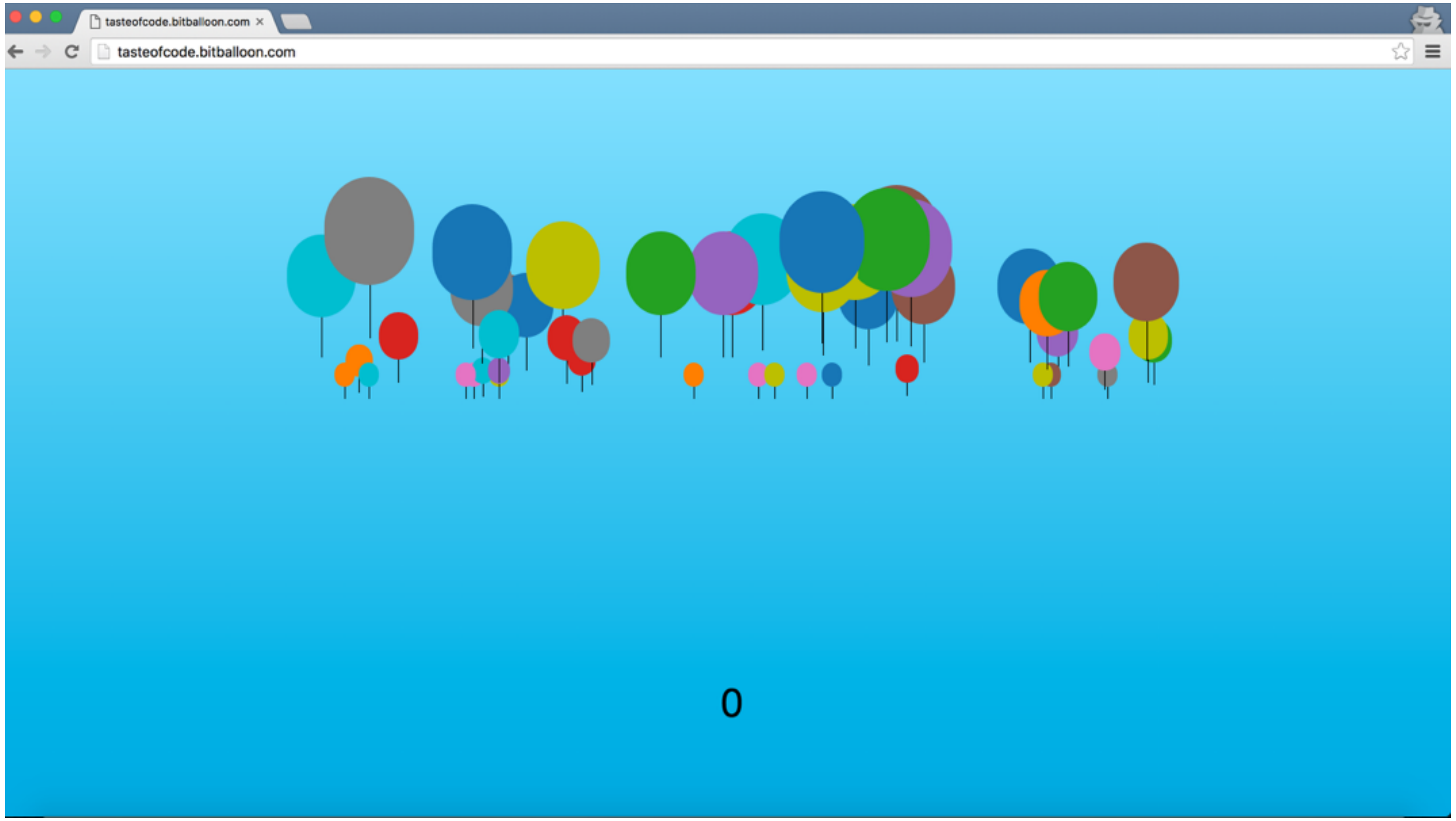
Today's objective

:{) Codaisseur

Build an online game

- HTML
- CSS
- jQuery

Put it online



HTML
Content & Structure

HTML is the language of the web.

What is HTML?

What is HTML?

:{) Codaisseur

HTML is a **markup language**
for describing web documents

Structure of HTML

Structure of HTML

```
<!DOCTYPE html>
<html>
  <head>
    <title></title>
  </head>
  <body>
  </body>
</html>
```

Structure of HTML

<DOCTYPE>
instruction to the web browser about the html version being used

```
<!DOCTYPE html>
<html>
  <head>
    <title></title>
  </head>
  <body>
  </body>
</html>
```

Structure of HTML

:{) Codaisseur

<DOCTYPE>
instruction to the web browser about the html version being used

<html>
indicates to the browser the start of the doc

```
<!DOCTYPE html>
<html>
  <head>
    <title></title>
  </head>
  <body>
  </body>
</html>
```

Structure of HTML

:{} Codaisseur

<DOCTYPE>
instruction to the web browser about the html version being used

<html>
indicates to the browser the start of the doc

<head>
contains information about the HTML doc: title, styles, scripts, etc.

```
<!DOCTYPE html>
<html>
  <head>
    <title></title>
  </head>
  <body>
  </body>
</html>
```

Structure of HTML

<DOCTYPE>
instruction to the web browser about the html version being used

<html>
indicates to the browser the start of the doc

<head>
contains information about the HTML doc: title, styles, scripts, etc.

<title>
defines the title of the doc

```
<!DOCTYPE html>
<html>
  <head>
    <title></title>
  </head>
  <body>
  </body>
</html>
```

Structure of HTML

:{) Codaisseur

<DOCTYPE>
instruction to the web browser about the html version being used

<html>
indicates to the browser the start of the doc

<head>
contains information about the HTML doc: title, styles, scripts, etc.

<title>
defines the title of the doc

<body>
contains the visual contents of HTML doc

```
<!DOCTYPE html>
<html>
  <head>
    <title></title>
  </head>
  <body>
  </body>
</html>
```

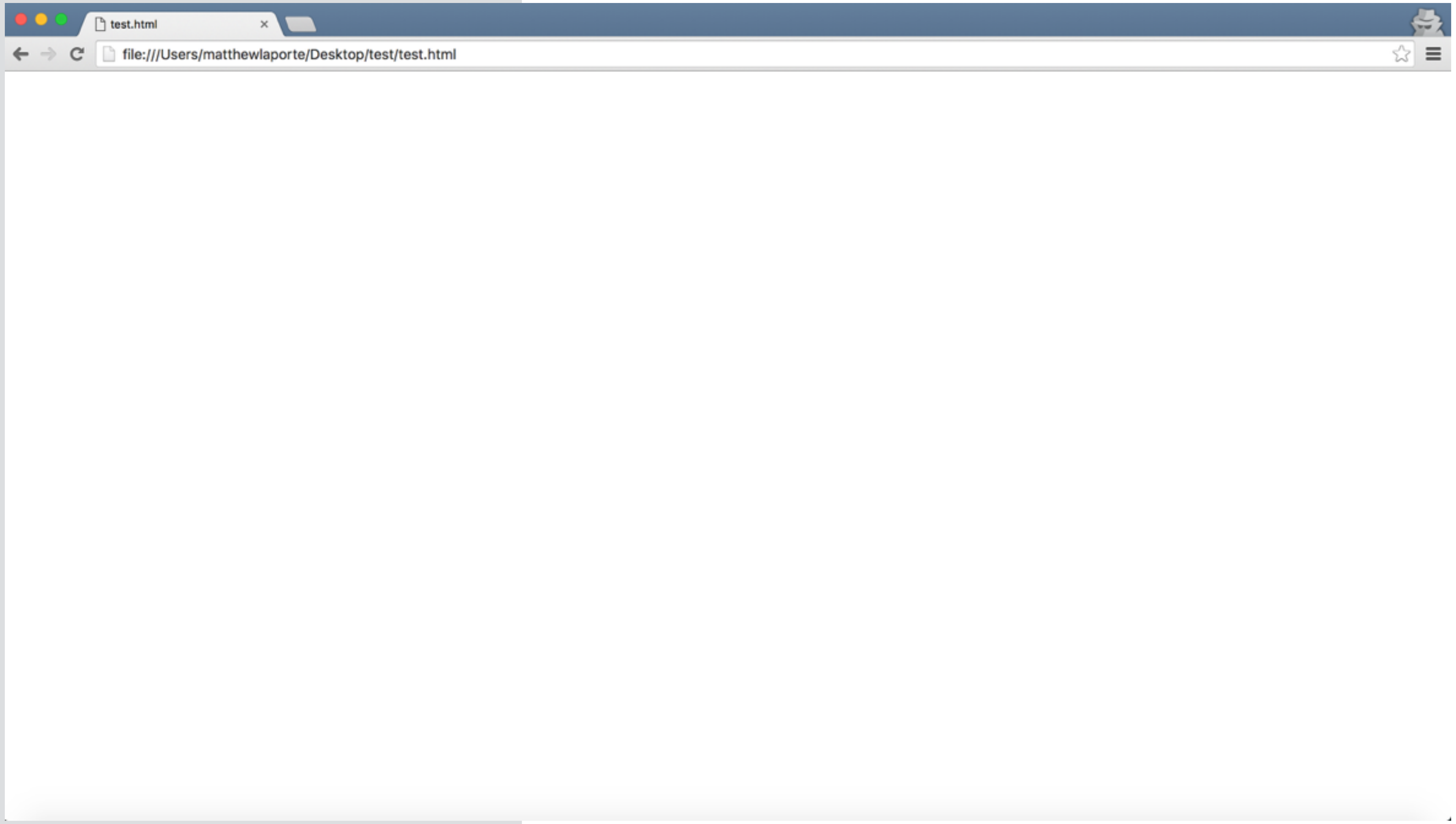
HTML Structure

```
<!DOCTYPE html>
<html>
  <head>
    <title></title>
  </head>
  <body>
  </body>
</html>
```

HTML Structure

:{) Codaisseur

```
<!DOCTYPE html>
<html>
  <head>
    <title></title>
  </head>
  <body>
  </body>
</html>
```



HTML Structure

:{) Codaisseur

HTML Structure

:{) Codaisseur

```
<!DOCTYPE html>
<html>
  <head>
    <title>Taste of Code</title>
  </head>
  <body>
    <h1>Agenda</h1>

    <p>On the menu today:</p>

    <ul>
      <li>Structure with HTML</li>
      <li>Styling with CSS</li>
      <li>Happiness with Lunch</li>
    </ul>
  </body>
</html>
```

index.html

HTML Structure

<h1>
defines a heading; <h1> - <h6>

```
<!DOCTYPE html>
<html>
  <head>
    <title>Taste of Code</title>
  </head>
  <body>
    <h1>Agenda</h1>

    <p>On the menu today:</p>

    <ul>
      <li>Structure with HTML</li>
      <li>Styling with CSS</li>
      <li>Happiness with Lunch</li>
    </ul>
  </body>
</html>
```

index.html

HTML Structure

:{) Codaisseur

<h1>
defines a heading; <h1> - <h6>

<p>
defines a paragraph

```html
<!DOCTYPE html>
<html>
 <head>
  <title>Taste of Code</title>
 </head>
 <body>
  <h1>Agenda</h1>

  <p>On the menu today:</p>

  <ul>
   <li>Structure with HTML</li>
   <li>Styling with CSS</li>
   <li>Happiness with Lunch</li>
  </ul>
 </body>
</html>
```

index.html

HTML Structure

:{) Codaisseur

<h1>
defines a heading; <h1> - <h6>

<p>
defines a paragraph

<ul>
defines an unordered list

```
<!DOCTYPE html>
<html>
 <head>
  <title>Taste of Code</title>
 </head>
 <body>
  <h1>Agenda</h1>

  <p>On the menu today:</p>

  <ul>
   <li>Structure with HTML</li>
   <li>Styling with CSS</li>
   <li>Happiness with Lunch</li>
  </ul>
 </body>
</html>
```

index.html

HTML Structure

:{) Codaisseur

<h1>
defines a heading; <h1> - <h6>

<p>
defines a paragraph

<ul>
defines an unordered list

<li>
defines a list item in a list

```html
<!DOCTYPE html>
<html>
 <head>
  <title>Taste of Code</title>
 </head>
 <body>
  <h1>Agenda</h1>

  <p>On the menu today:</p>

  <ul>
   <li>Structure with HTML</li>
   <li>Styling with CSS</li>
   <li>Happiness with Lunch</li>
  </ul>
 </body>
</html>
```

index.html

HTML Structure

```html
<!DOCTYPE html>
<html>
  <head>
    <title>Taste of Code</title>
  </head>
  <body>
    <h1>Agenda</h1>

    <p>On the menu today:</p>

    <ul>
      <li>Structure with HTML</li>
      <li>Styling with CSS</li>
      <li>Happiness with Lunch</li>
    </ul>
  </body>
</html>
```

index.html

HTML Structure

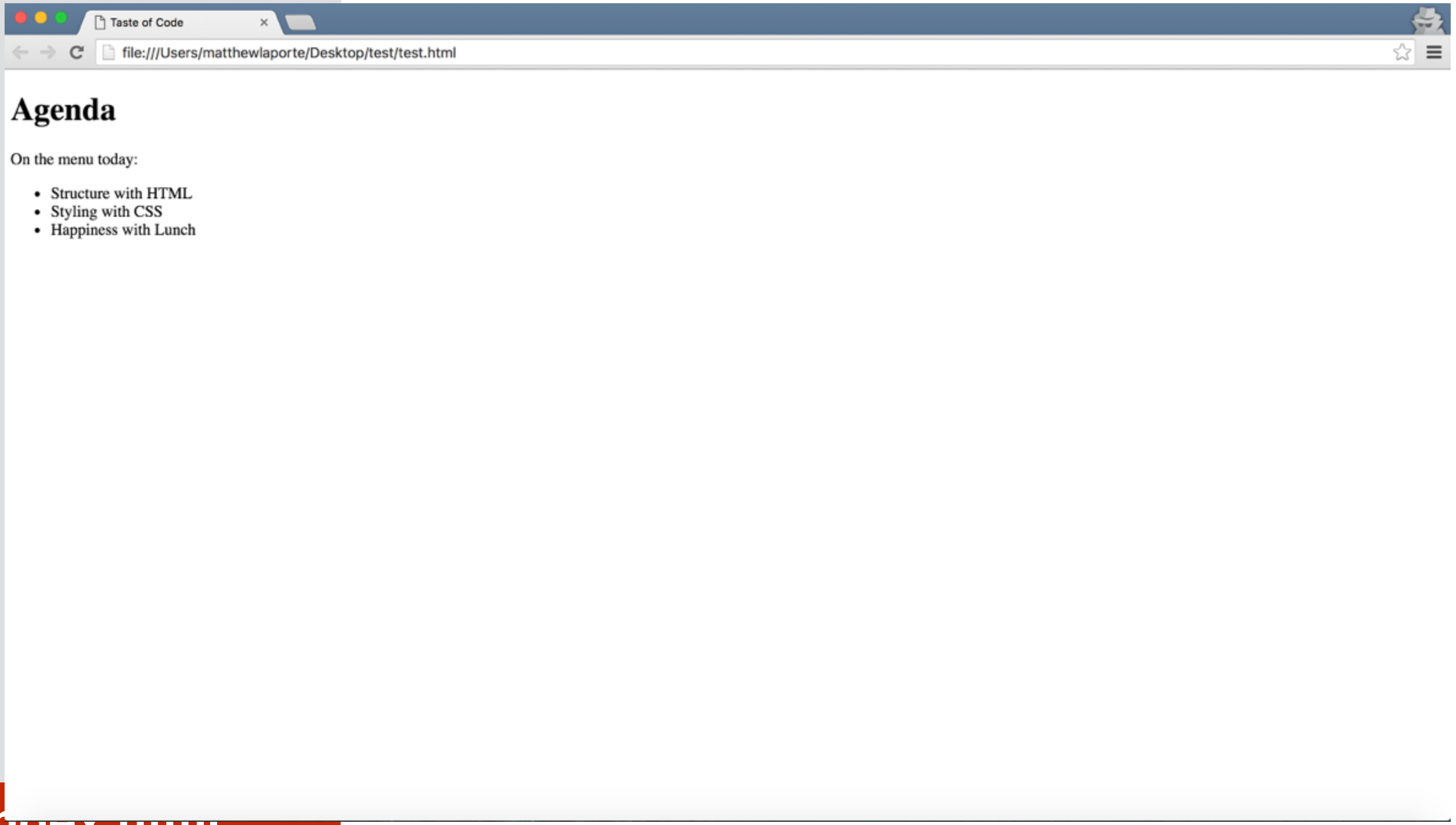
:{) Codaisseur

```
<!DOCTYPE html>
<html>
 <head>
  <title>Taste of Code</title>
 </head>
 <body>
  <h1>Agenda</h1>

  <p>On the menu today:</p>

  <ul>
   <li>Structure with HTML</li>
   <li>Styling with CSS</li>
   <li>Happiness with Lunch</li>
  </ul>
 </body>
</html>
```

index.html



Agenda

On the menu today:

- Structure with HTML
- Styling with CSS
- Happiness with Lunch

<Tags>…</Tags>

<Tags>...</Tags>

:{) Codaisseur

```
<!DOCTYPE html>
<html>
 <head>
  <title>Taste of Code

 <body>
  <h1>Agenda

  <p>On the menu today:

  <ul>
   <li>Structure with HTML
   <li>Styling with CSS
   <li>Happiness with Lunch
```

<Tags>...</Tags>

```
<!DOCTYPE html>
<html>
  <head>
    <title>Taste of Code</title>
  </head>
  <body>
    <h1>Agenda</h1>

    <p>On the menu today:</p>

    <ul>
      <li>Structure with HTML</li>
      <li>Styling with CSS</li>
      <li>Happiness with Lunch</li>
    </ul>
  </body>
</html>
```

<Tags>...</Tags>

:{) Codaisseur

```
<!DOCTYPE html>
<html>
 <head>
  <title>Taste of Code</title>
 </head>
 <body>
  <h1>Agenda</h1>

  <p>On the menu today:</p>

  <ul>
   <li>Structure with HTML</li>
   <li>Styling with CSS</li>
   <li>Happiness with Lunch</li>
  </ul>
 </body>
</html>
```

<h1>Agenda

Opening tag = starts heading

<Tags>...</Tags>

:{) Codaisseur

```
<!DOCTYPE html>
<html>
  <head>
   <title>Taste of Code</title>
  </head>
  <body>
   <h1>Agenda</h1>

   <p>On the menu today:</p>

   <ul>
     <li>Structure with HTML</li>
     <li>Styling with CSS</li>
     <li>Happiness with Lunch</li>
   </ul>
  </body>
</html>
```

<h1>Agenda</h1>

Opening tag = starts heading
Closing tag = stops heading

✎ Exercise

✎ Exercise

Create your first HTML document utilising title, h1, p and ul/li elements:

✎ Exercise

Create your first HTML document utilising title, h1, p and ul/li elements:

✎ Exercise

Create your first HTML document utilising title, h1, p and ul/li elements:

- Make a new folder for your project

✎ Exercise

Create your first HTML document utilising title, h1, p and ul/li elements:

- Make a new folder for your project
- In atom, create a new file called **index.html**

✎ Exercise

Create your first HTML document utilising title, h1, p and ul/li elements:

- Make a new folder for your project
- In atom, create a new file called **index.html**
- Follow the basic HTML structure

✎ Exercise

Create your first HTML document utilising title, h1, p and ul/li elements:

- Make a new folder for your project
- In atom, create a new file called **index.html**
- Follow the basic HTML structure

```
<!DOCTYPE html>
<html>
  <head>
    <title></title>
  </head>
  <body>
  </body>
</html>
```

CSS

Style & Presentation

CSS describes how HTML looks.

What is CSS?

What is CSS?

CSS is a **stylesheet language** that describes the presentation of an HTML document.

What is CSS?

CSS is a **stylesheet language** that
describes the presentation of an HTML document.

text color

What is CSS?

CSS is a **stylesheet language** that
describes the presentation of an HTML document.

text color
fonts style

What is CSS?

:{) Codaisseur

CSS is a **stylesheet language** that
describes the presentation of an HTML document.

text color
fonts style
spacing between elements

What is CSS?

CSS is a **stylesheet language** that
describes the presentation of an HTML document.

text color
fonts style
spacing between elements
background images

CSS Structure

```
<!DOCTYPE html>
<html>
  <head>
    <title>Taste of Code</title>

    <style>
      h1 {
                 color: red;
               }
    </style>

  </head>
  <body>
    <h1>Agenda</h1>
       ...
  </body>
</html>
```

index.html

CSS Structure

:{) Codaisseur

<style>
defines style information for an HTML doc;
how elements should render in the browser

```
<!DOCTYPE html>
<html>
 <head>
  <title>Taste of Code</title>

  <style>
   h1 {
          color: red;
         }
  </style>

 </head>
 <body>
  <h1>Agenda</h1>
      ...
 </body>
</html>
```

index.html

CSS Structure

:{) Codaisseur

<style>
defines style information for an HTML doc; how elements should render in the browser

h1 {}
element; holds properties that can alter it’s style

```
<!DOCTYPE html>
<html>
  <head>
    <title>Taste of Code</title>

    <style>
      h1 {
              color: red;
            }
    </style>

  </head>
  <body>
    <h1>Agenda</h1>
       ...
  </body>
</html>
```

index.html

CSS Structure

<style>
defines style information for an HTML doc; how elements should render in the browser

h1 {}
element; holds properties that can alter it's style

color
property

```
<!DOCTYPE html>
<html>
 <head>
  <title>Taste of Code</title>

  <style>
   h1 {
         color: red;
       }
  </style>

 </head>
 <body>
  <h1>Agenda</h1>
     ...
 </body>
</html>
```

index.html

CSS Structure

<style>
defines style information for an HTML doc; how elements should render in the browser

h1 {}
element; holds properties that can alter it’s style

color
property

red
value

```
<!DOCTYPE html>
<html>
  <head>
    <title>Taste of Code</title>

    <style>
      h1 {
            color: red;
          }
    </style>

  </head>
  <body>
    <h1>Agenda</h1>
       ...
  </body>
</html>
```

index.html

CSS Structure

:{) Codaisseur

<style>
defines style information for an HTML doc; how elements should render in the browser

h1 {}
element; holds properties that can alter it's style

color
property

red
value

Something to note:

A colon(":") proceeds the property and a semi-colon (";") proceeds the value

```
<!DOCTYPE html>
<html>
  <head>
    <title>Taste of Code</title>

    <style>
      h1 {
              color: red;
           }
    </style>

  </head>
  <body>
    <h1>Agenda</h1>
       ...
  </body>
</html>
```

index.html

CSS Structure

:{) Codaisseur

```
<!DOCTYPE html>
<html>
  <head>
    <title>Taste of Code</title>

    <style>
      h1 {
            color: red;
          }
    </style>

  </head>
  <body>
    <h1>Agenda</h1>
      ...
  </body>
</html>
```

index.html

CSS Structure

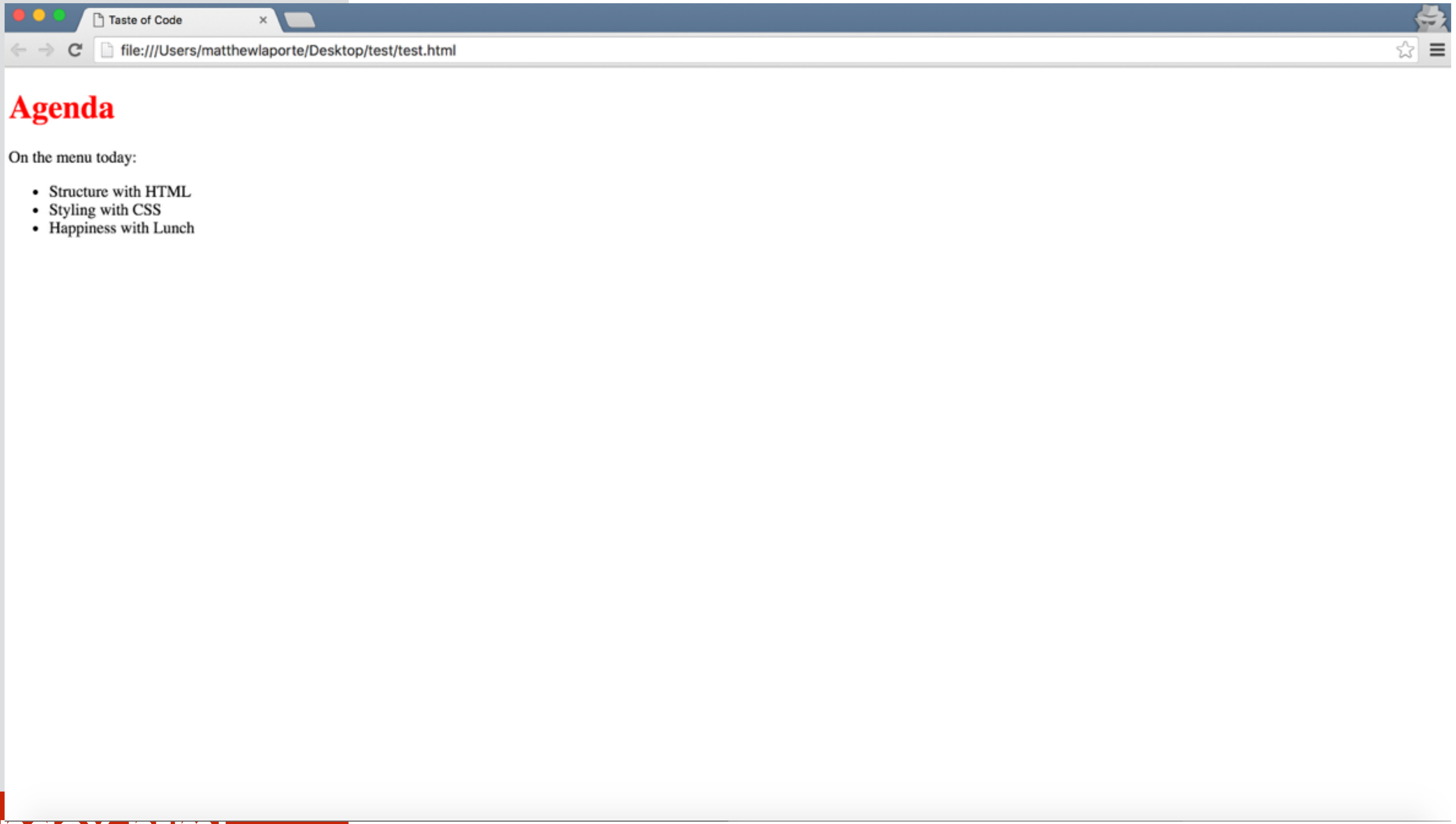
:{) Codaisseur

```
<!DOCTYPE html>
<html>
  <head>
    <title>Taste of Code</title>

    <style>
      h1 {
            color: red;
          }
    </style>

  </head>
  <body>
    <h1>Agenda</h1>
      ...
  </body>
</html>
```

index.html


Taste of Code

file:///Users/matthewlaporte/Desktop/test/test.html

Agenda

On the menu today:

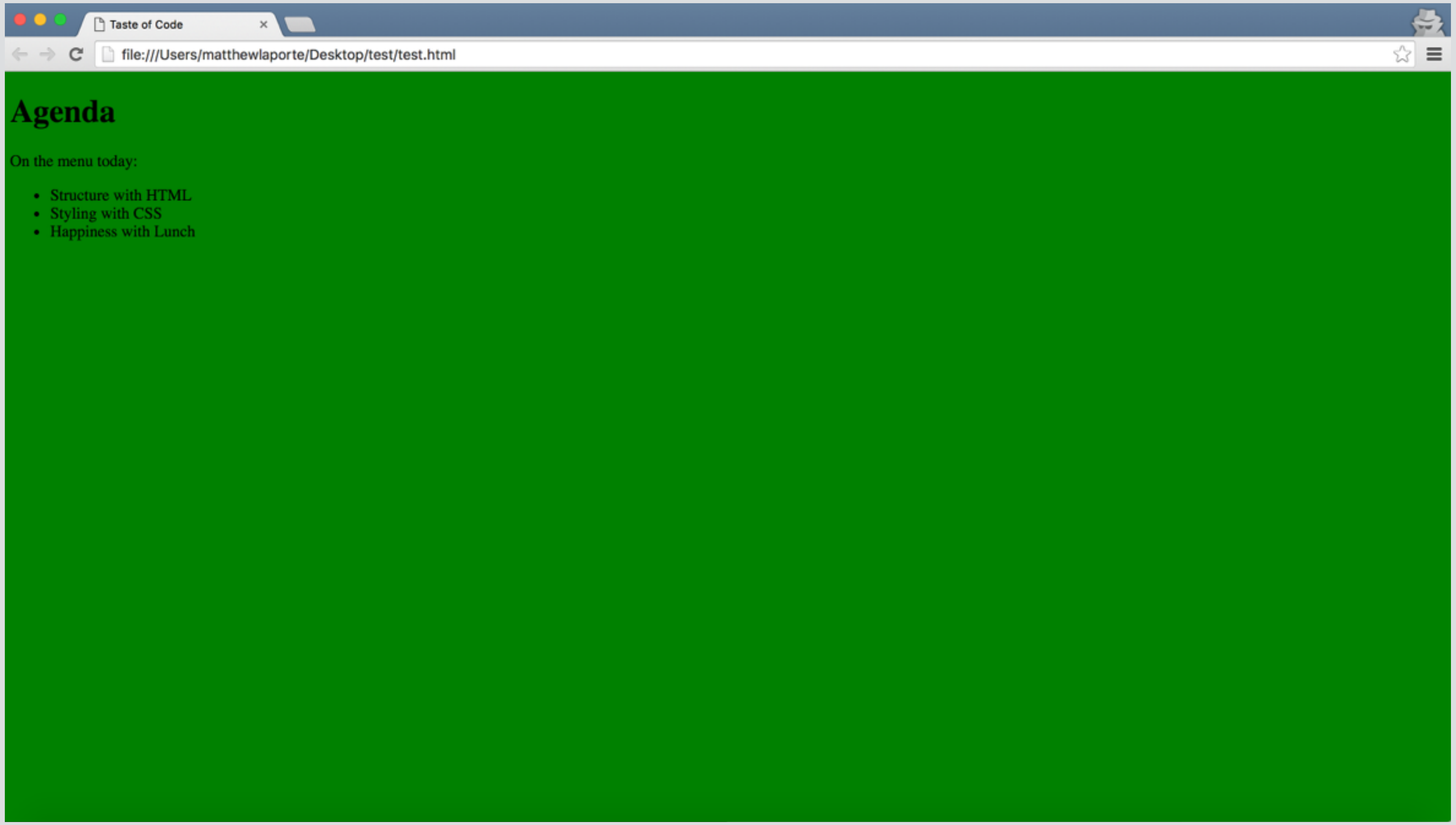
- Structure with HTML
- Styling with CSS
- Happiness with Lunch

✎ Exercise

Change the background color of your document to green.

✎ Exercise

Change the background color of your document to green.



Agenda

On the menu today:

- Structure with HTML
- Styling with CSS
- Happiness with Lunch

CSS Classes

CSS Classes

```
<!DOCTYPE html>
<html>
  <head>
    <title>Taste of Code</title>

    <style>
      .warning {
              color: red;
            }
    </style>

  </head>
  <body>
    <h1 class="warning">Agenda</h1>
       ...
  </body>
</html>
```

index.html

CSS Classes

:{) Codaisseur

Classes allow you to apply the same properties to multiple elements.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Taste of Code</title>

    <style>
      .warning {
              color: red;
            }
    </style>

  </head>
  <body>
    <h1 class="warning">Agenda</h1>
       ...
  </body>
</html>
```

index.html

CSS Classes

:{) Codaisseur

Classes allow you to apply the same properties to multiple elements.

When selecting a class to style, a period (“.”) precedes the name of the class.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Taste of Code</title>

    <style>
      .warning {
               color: red;
             }
    </style>

  </head>
  <body>
    <h1 class=“warning”>Agenda</h1>
       …
  </body>
</html>
```

index.html

CSS Classes

Classes allow you to apply the same properties to multiple elements.

When selecting a class to style, a period (".") precedes the name of the class.

You add the class to the element's first tag in this format: **class="class-name"**.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Taste of Code</title>

    <style>
      .warning {
              color: red;
          }
    </style>

  </head>
  <body>
    <h1 class="warning">Agenda</h1>
       ...
  </body>
</html>
```

index.html

CSS Classes

:{) Codaisseur

Classes allow you to apply the same properties to multiple elements.

When selecting a class to style, a period (".") precedes the name of the class.

You add the class to the element's first tag in this format: **class="class-name"**.

The name of your class can be anything.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Taste of Code</title>

    <style>
      .warning {
              color: red;
            }
    </style>

  </head>
  <body>
    <h1 class="warning">Agenda</h1>
        ...
  </body>
</html>
```

index.html

CSS Classes

:{) Codaisseur

```
<!DOCTYPE html>
<html>
  <head>
    <title>Taste of Code</title>

    <style>
      .warning {
              color: red;
            }
    </style>

  </head>
  <body>
    <h1 class="warning">Agenda</h1>
        ...
  </body>
</html>
```

index.html

CSS Classes

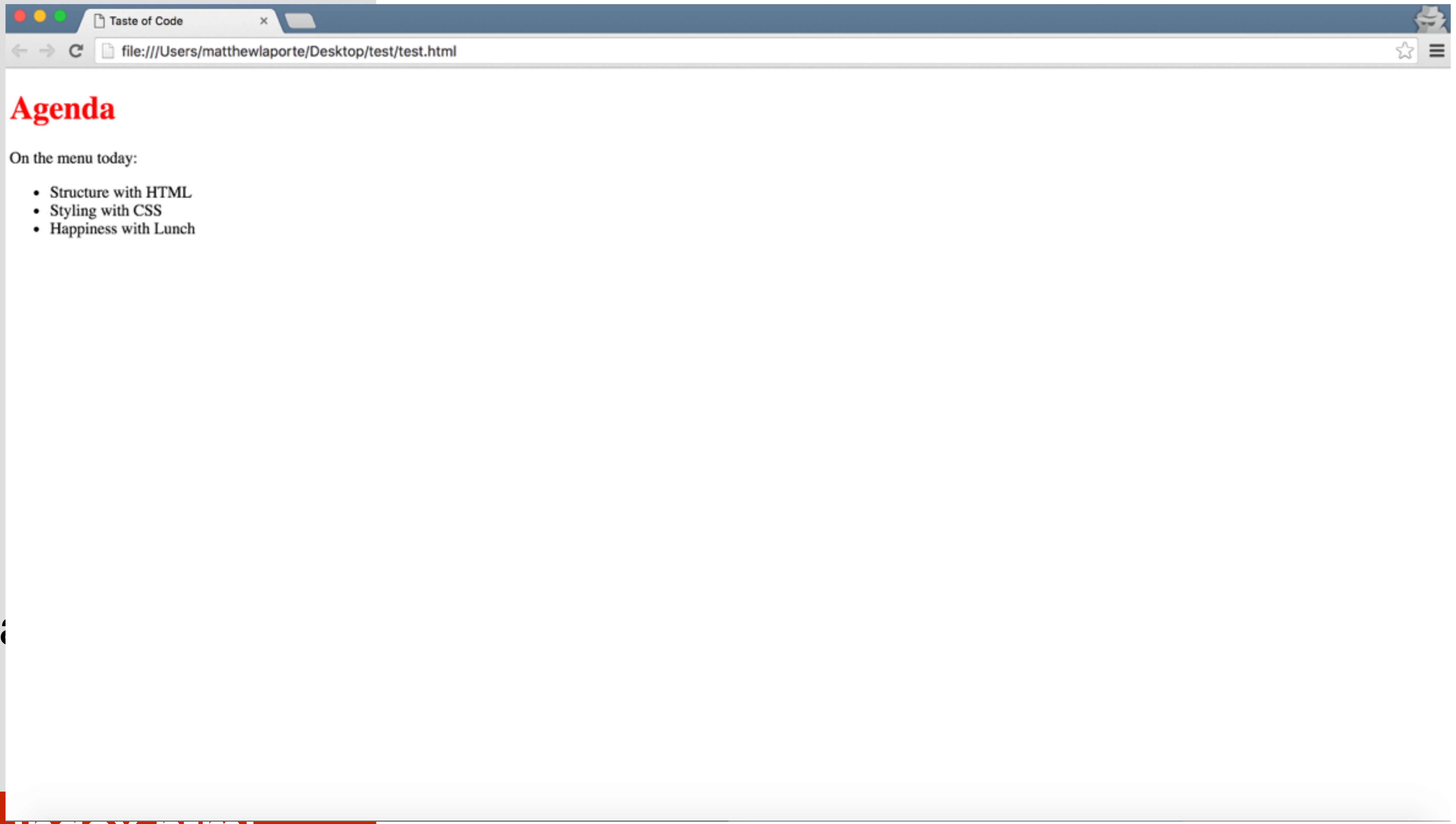
:{) Codaisseur

```
<!DOCTYPE html>
<html>
  <head>
    <title>Taste of Code</title>

    <style>
     .warning {
              color: red;
            }
    </style>

  </head>
  <body>
    <h1 class=“warning”>Agend
      ...
  </body>
</html>
```

index.html


Taste of Code

file:///Users/matthewlaporte/Desktop/test/test.html

Agenda

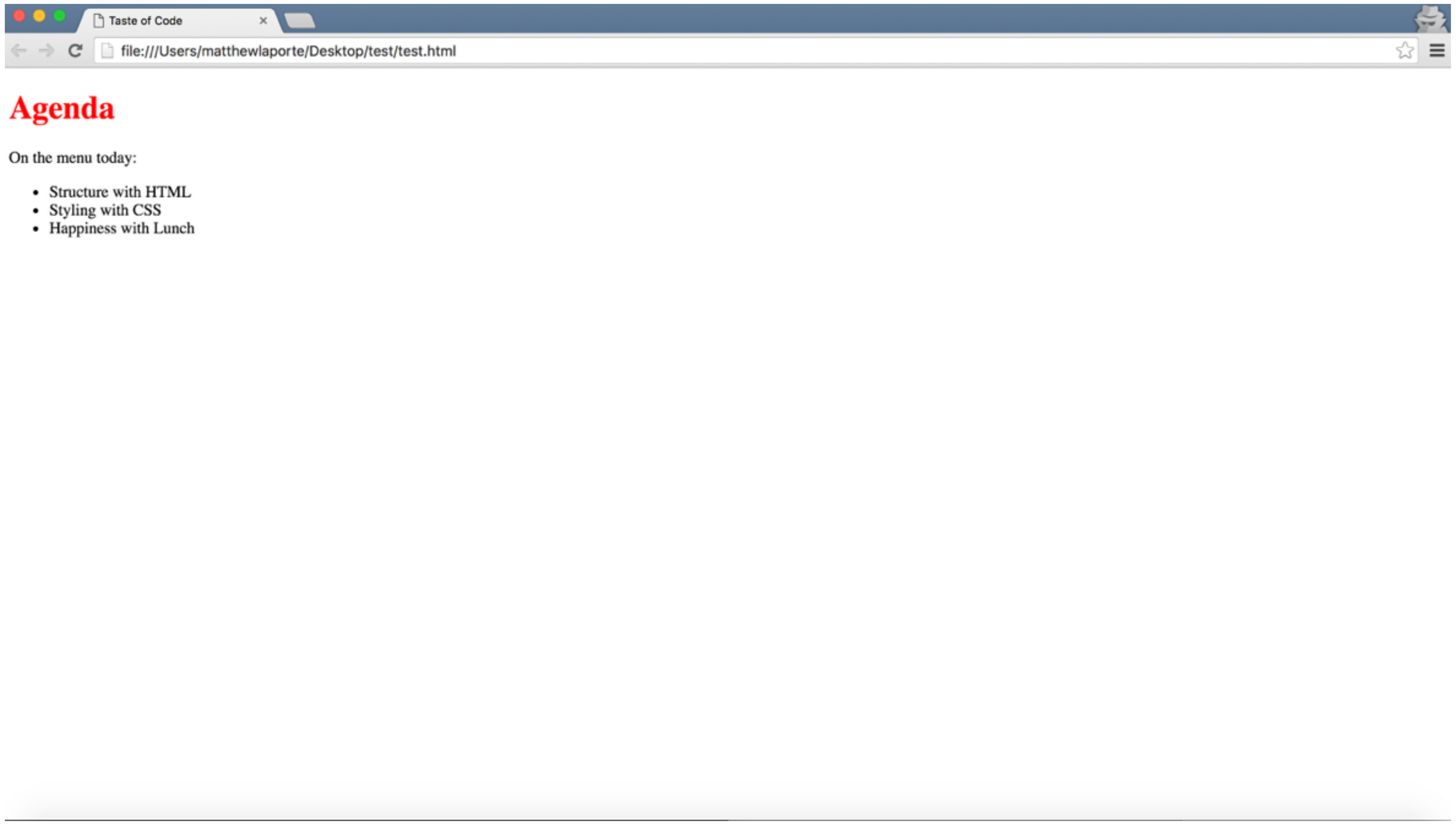
On the menu today:

- Structure with HTML
- Styling with CSS
- Happiness with Lunch

CSS Classes

CSS Classes

<h1 class=“warning”>Agenda</h1>


Taste of Code

file:///Users/matthewlaporte/Desktop/test/test.html

Agenda

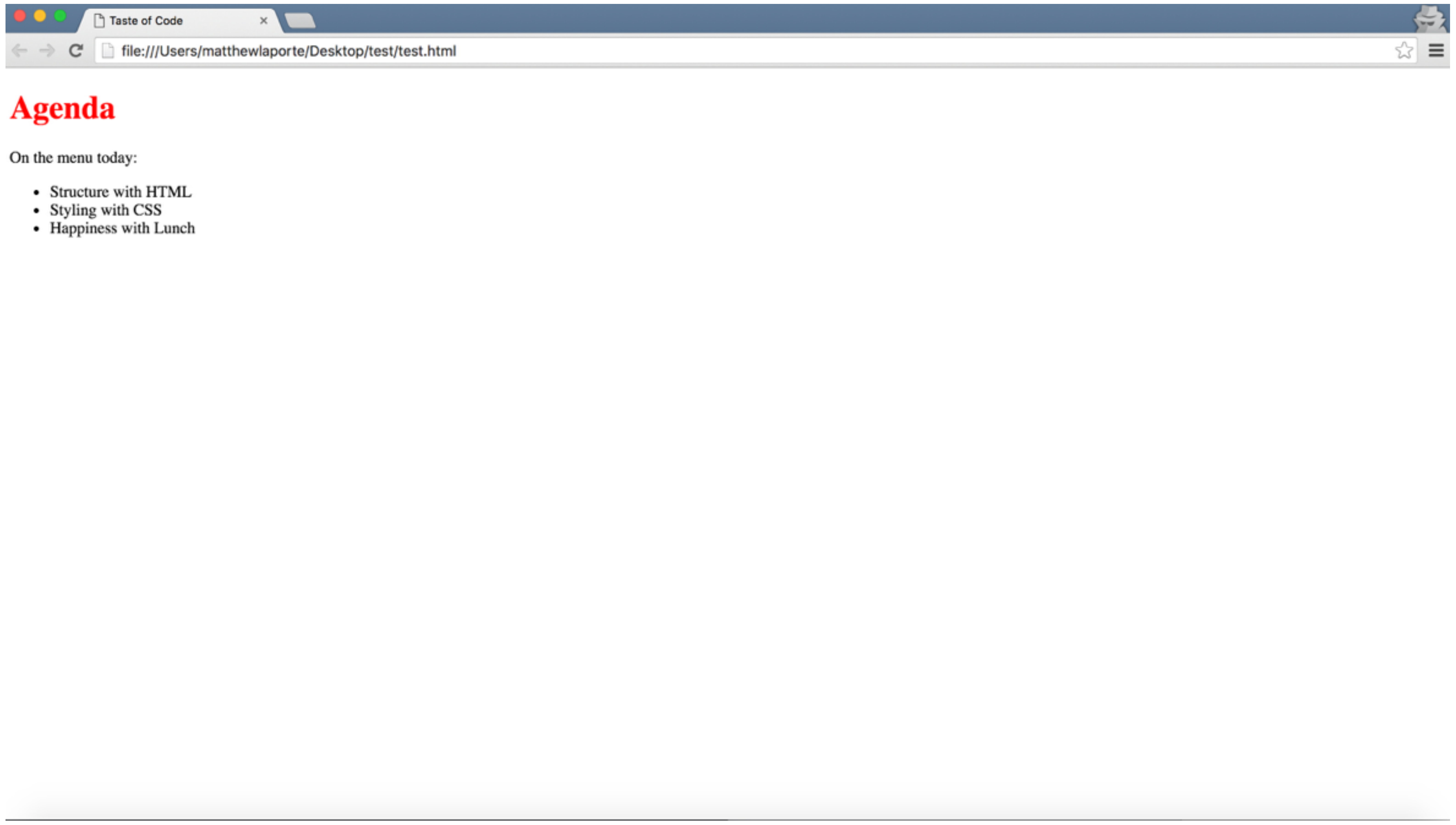
On the menu today:

- Structure with HTML
- Styling with CSS
- Happiness with Lunch

CSS Classes

<h1 class="warning">Agenda</h1>

<p class="warning">On the menu today:</p>



Agenda

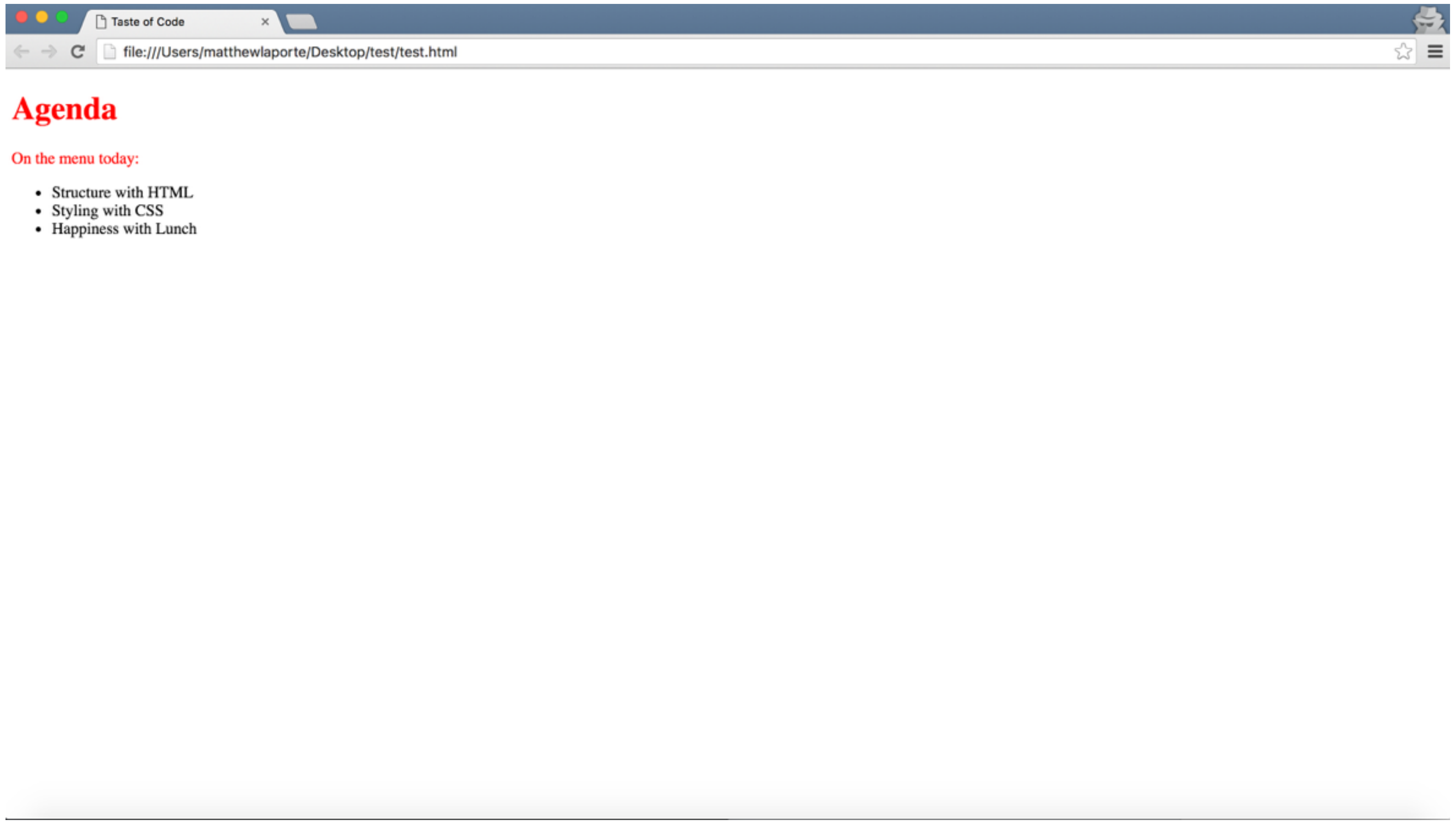
On the menu today:

- Structure with HTML
- Styling with CSS
- Happiness with Lunch

CSS Classes

```
<h1 class=“warning”>Agenda</h1>

<p class=“warning”>On the menu today:</p>
```



Taste of Code

file:///Users/matthewlaporte/Desktop/test/test.html

Agenda

On the menu today:

- Structure with HTML
- Styling with CSS
- Happiness with Lunch

✎ Exercise

✎ Exercise

Warning

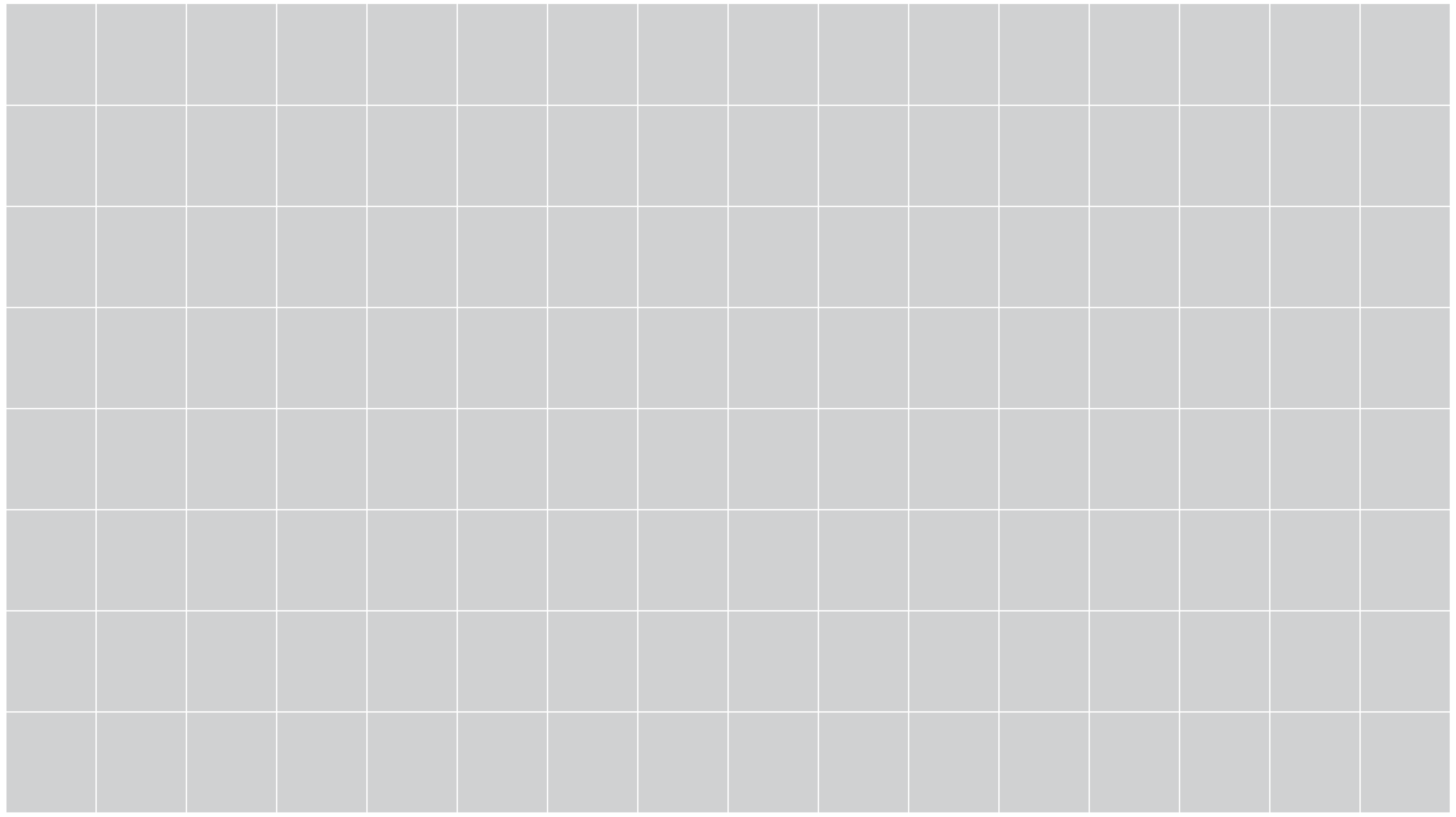
Apply the warning class to the main heading in your web page and change its color.

Padding & Margin in Pixels

Padding & Margin in Pixels

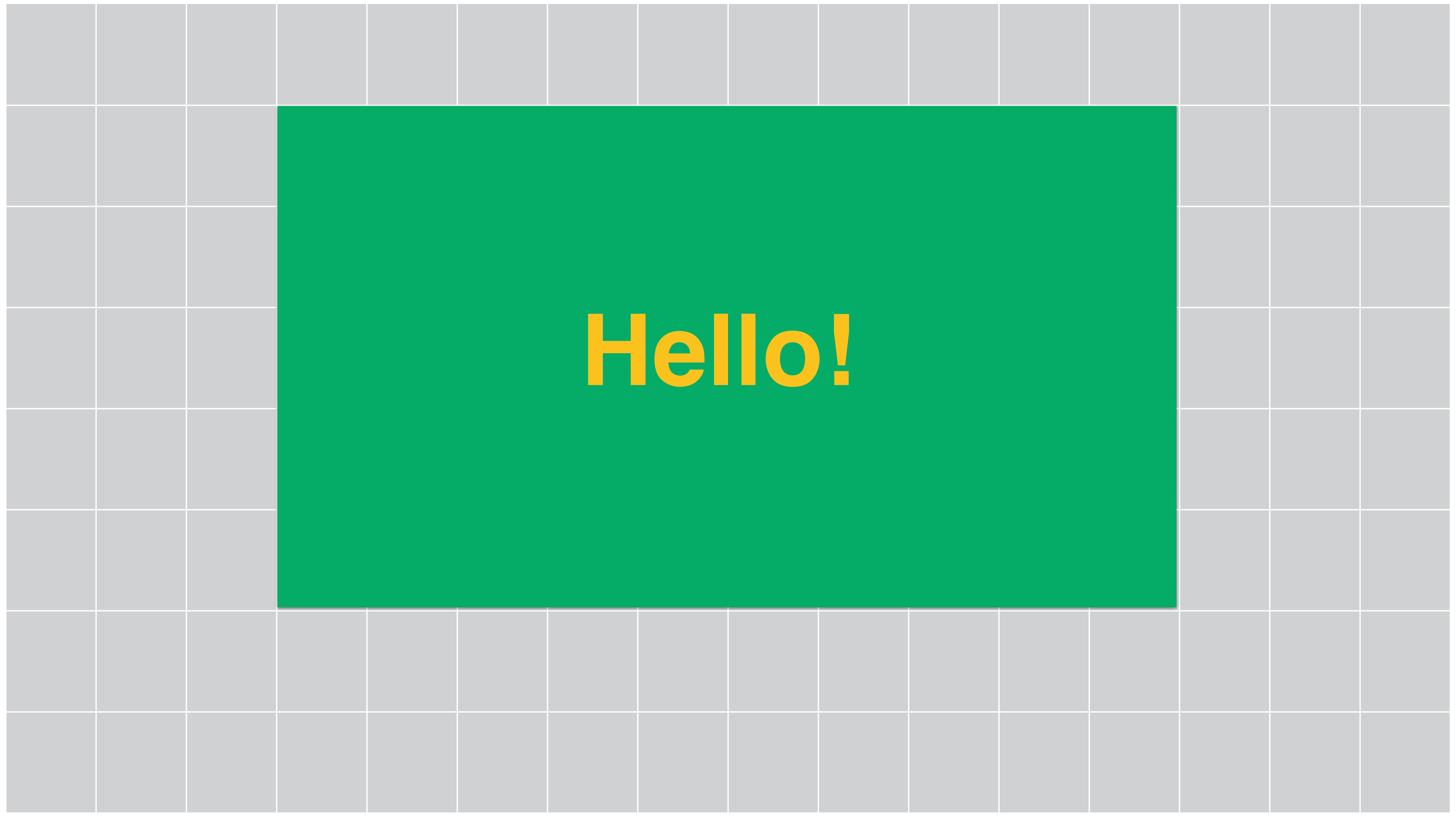
:{) Codaisseur

A screen consists of pixels



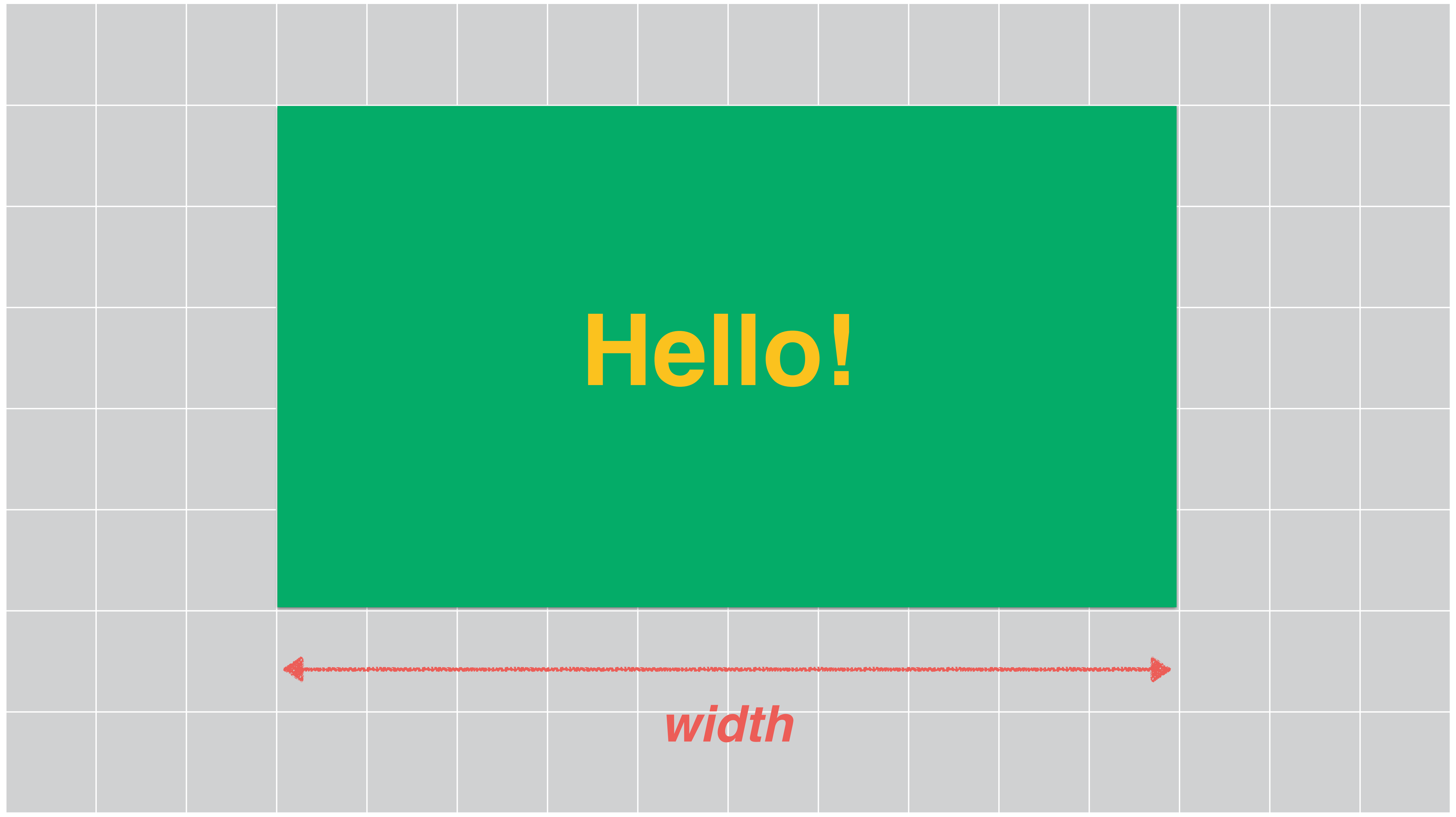
Padding & Margin in Pixels

A screen consists of pixels



Padding & Margin in Pixels

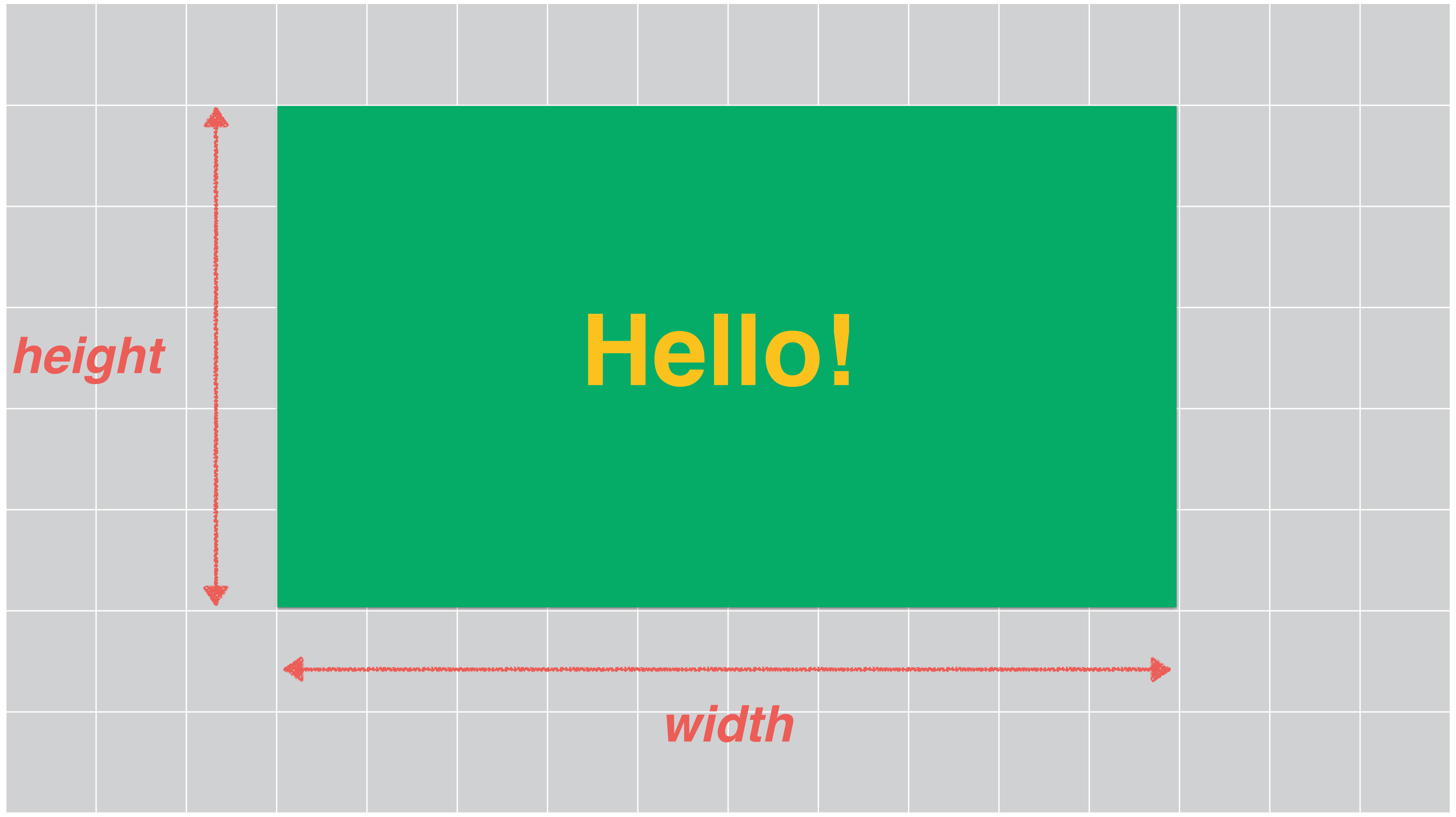
A screen consists of pixels



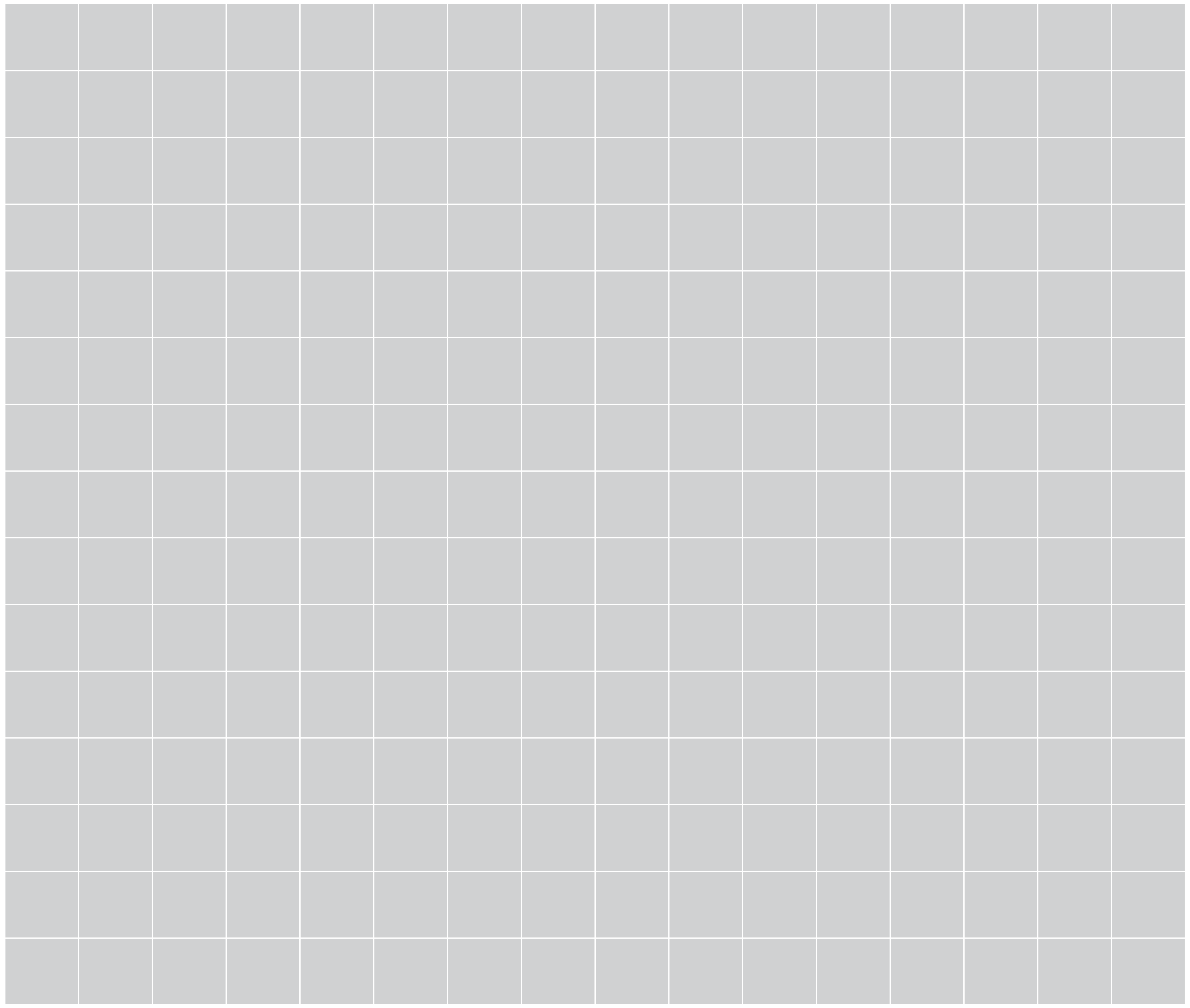
Padding & Margin in Pixels

:{} Codaisseur

A screen consists of pixels



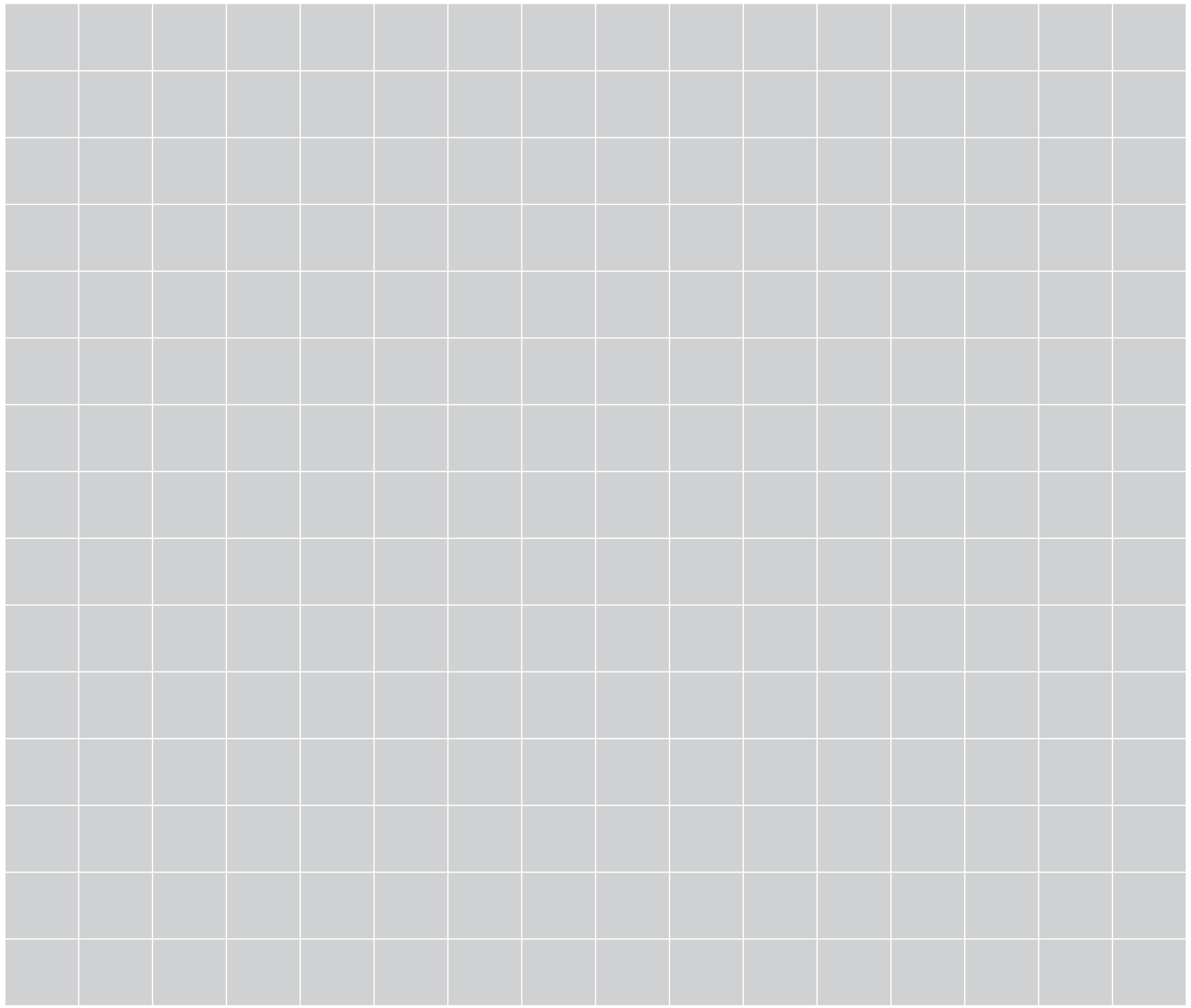
Padding



Padding

:{) Codaisseur

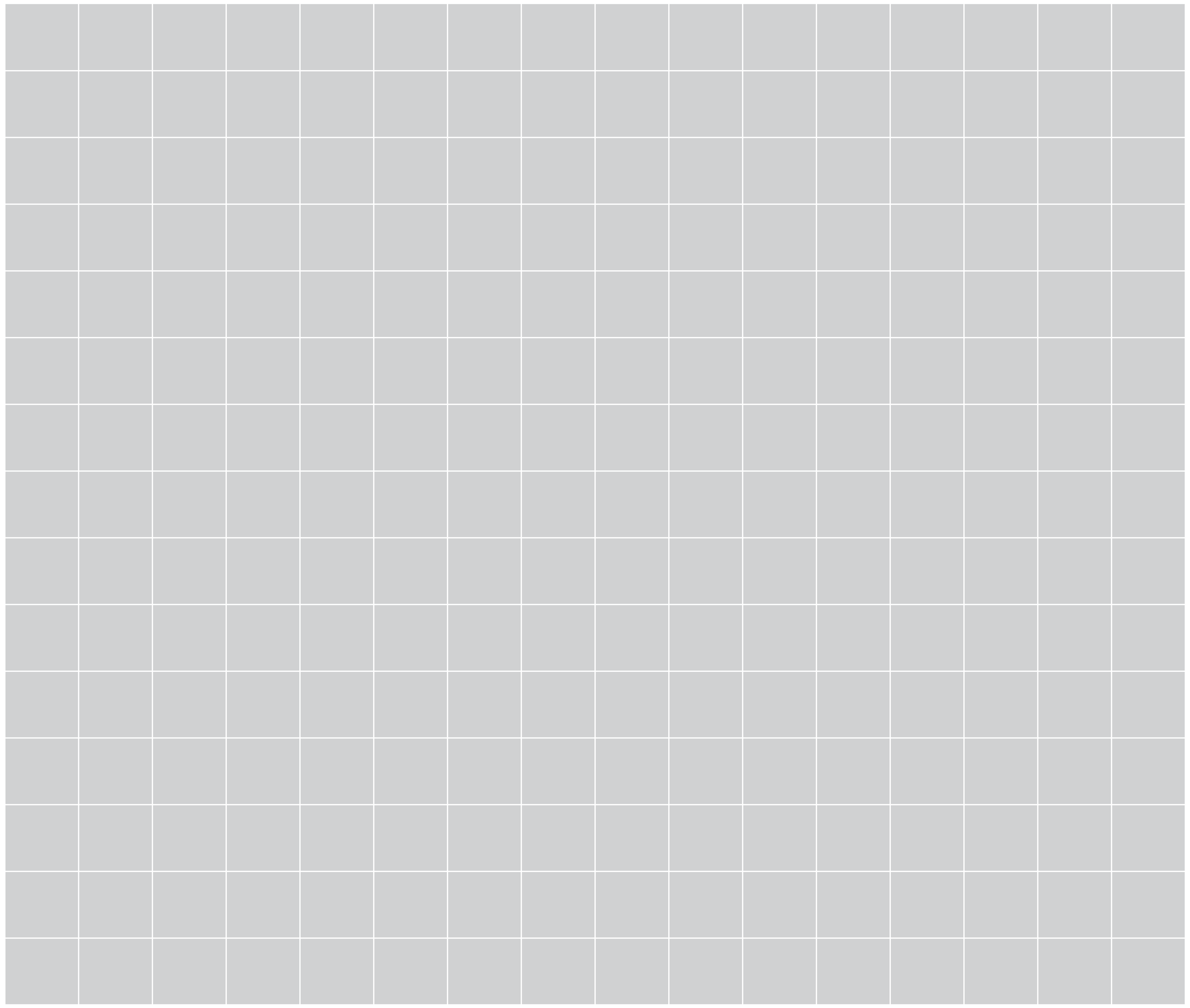
Generate space around content.



Padding

Generate space around content.

Sets the size of white space between the element content and element border.



Padding

Generate space around content.

Sets the size of white space between the element content and element border.

<h1 class="yellow"**>**Agenda**</h1>**

Padding

:{) Codaisseur

Generate space around content.

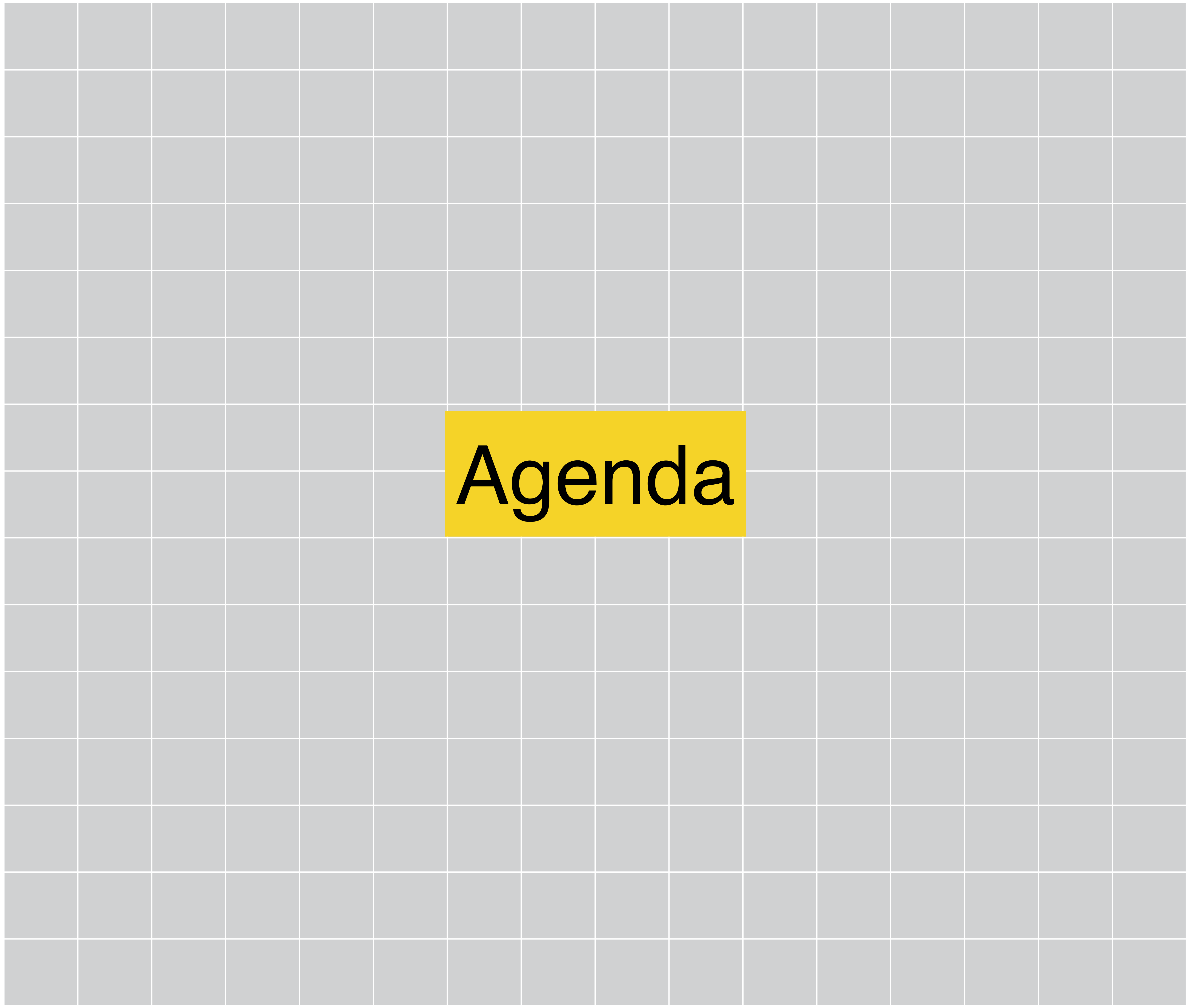
Sets the size of white space between the element content and element border.

<h1 class="yellow"**>**Agenda**</h1>**

```
<style>
 .yellow {
  background-color: yellow;



  }
</style>
```



Padding

:{) Codaisseur

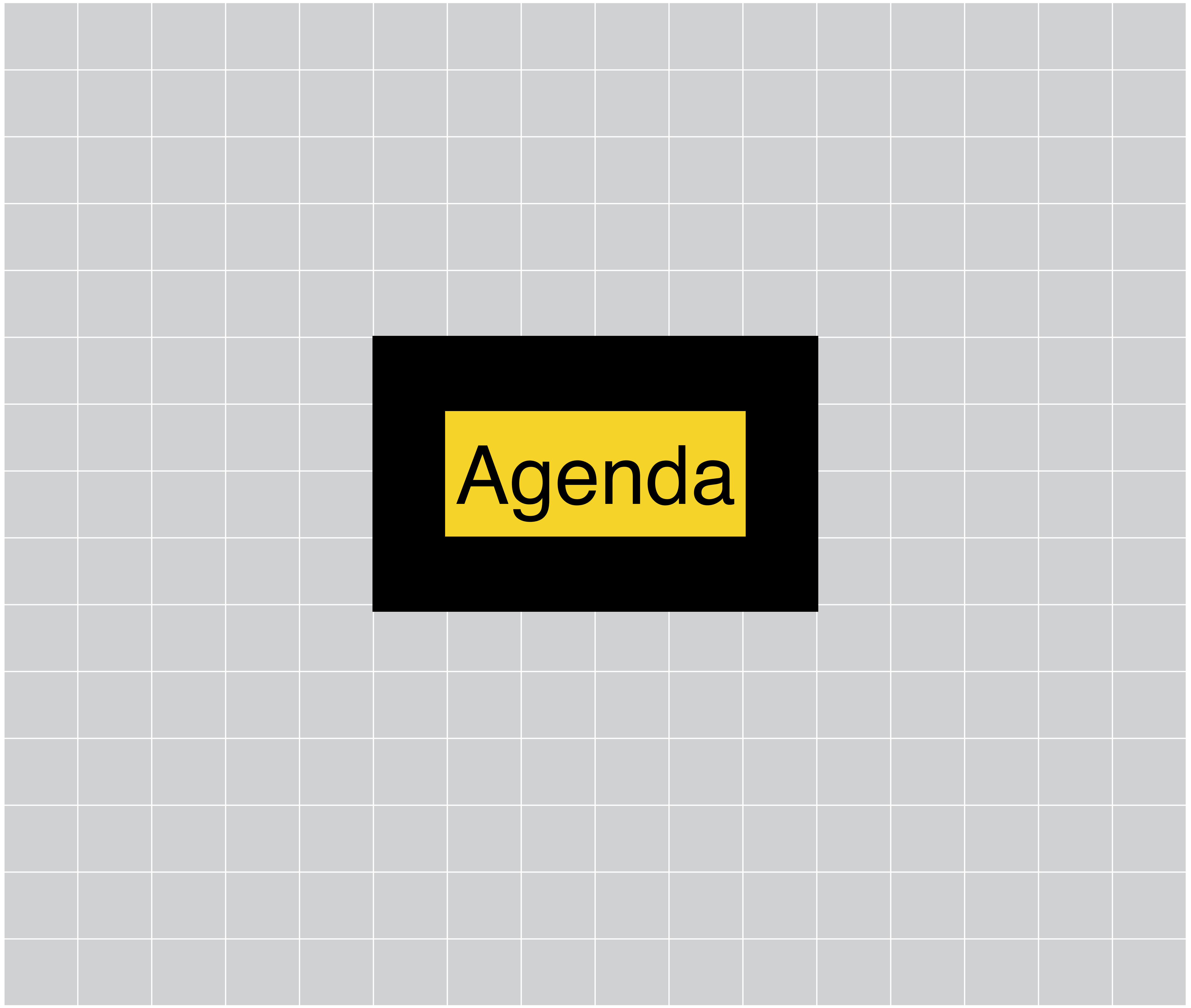
Generate space around content.

Sets the size of white space between the element content and element border.

<h1 class="yellow"**>**Agenda**</h1>**

```
<style>
 .yellow {
  background-color: yellow;
  border: 1px solid;

  }
</style>
```



Padding

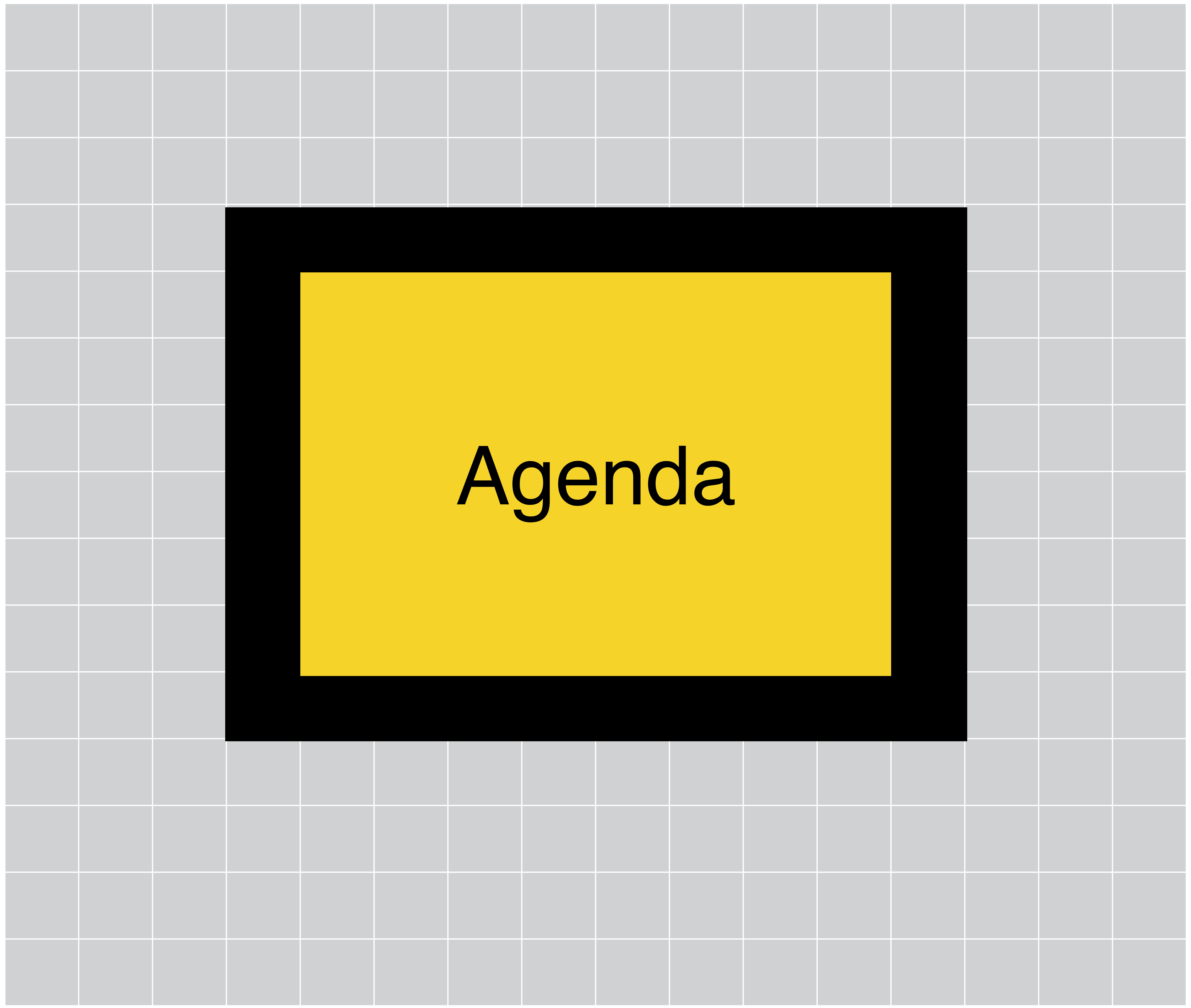
:{) Codaisseur

Generate space around content.

Sets the size of white space between the element content and element border.

<h1 class="yellow"**>**Agenda**</h1>**

```
<style>
 .yellow {
   background-color: yellow;
   border: 1px solid;
   padding: 2px;
   }
</style>
```



Padding

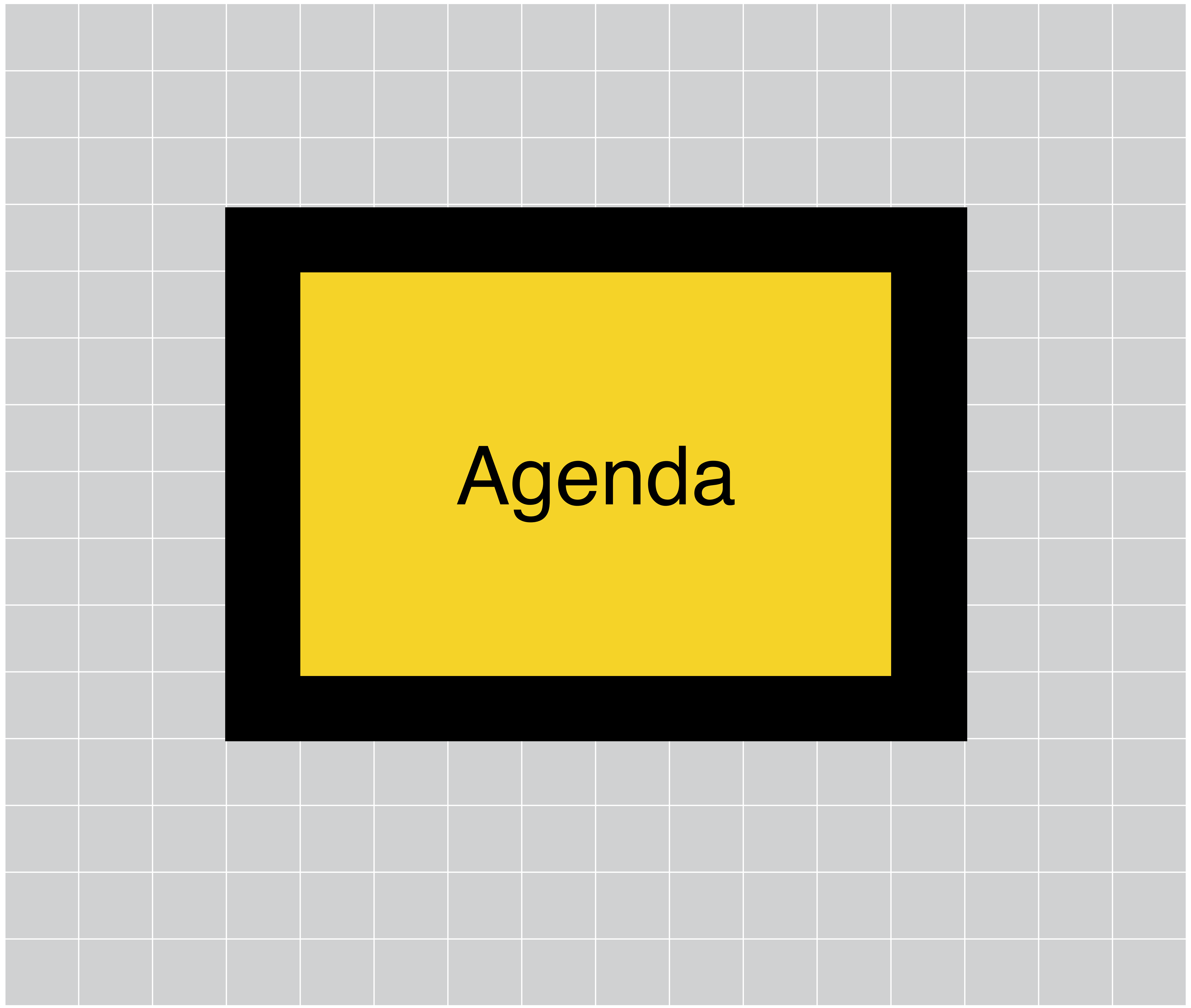
Generate space around content.

Sets the size of white space between the element content and element border.

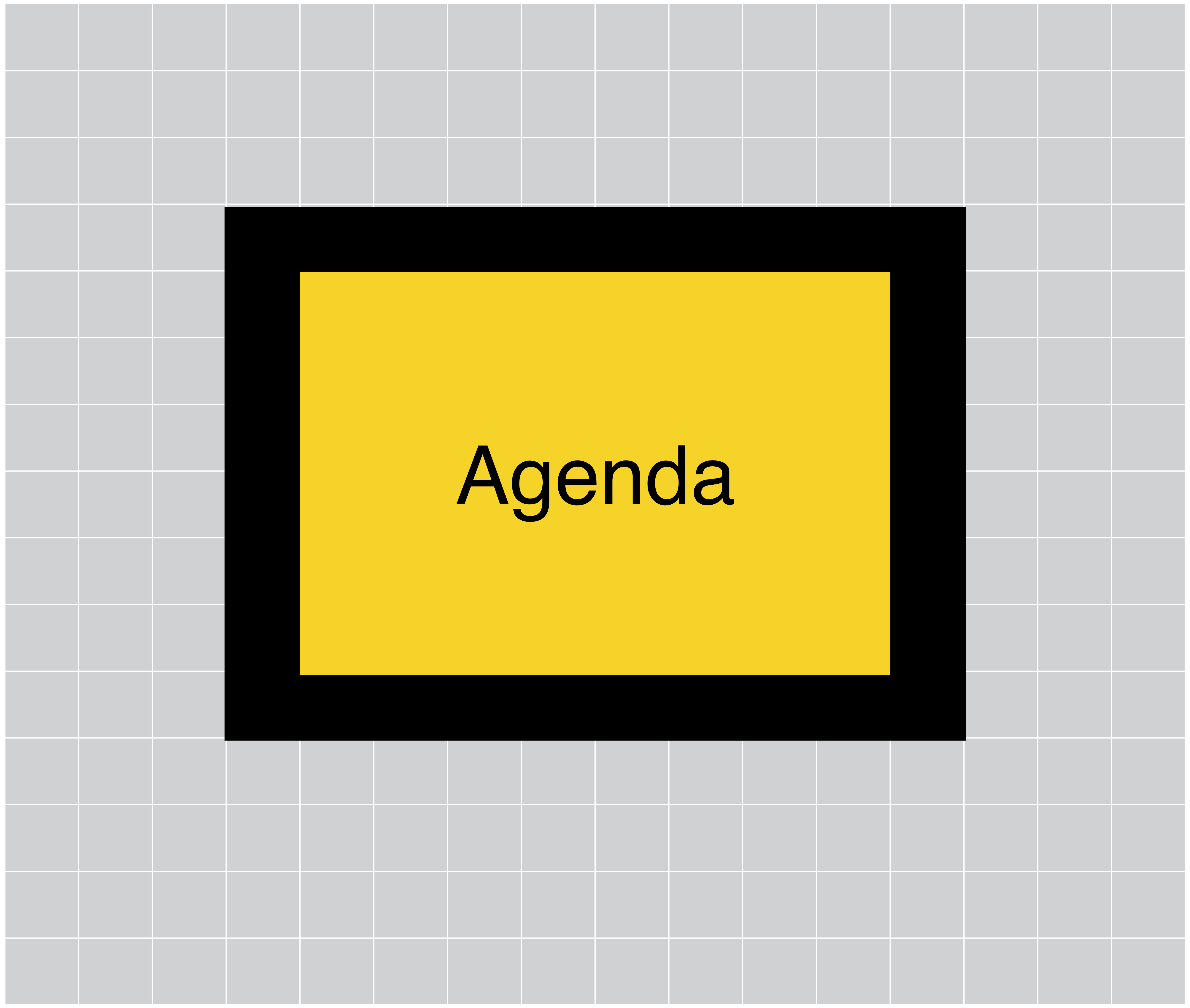
<h1 class="yellow"**>**Agenda**</h1>**

```
<style>
 .yellow {
  background-color: yellow;
  border: 1px solid;
  padding: 2px;
  }
</style>
```

padding-left, padding-right
padding-top, padding-bottom

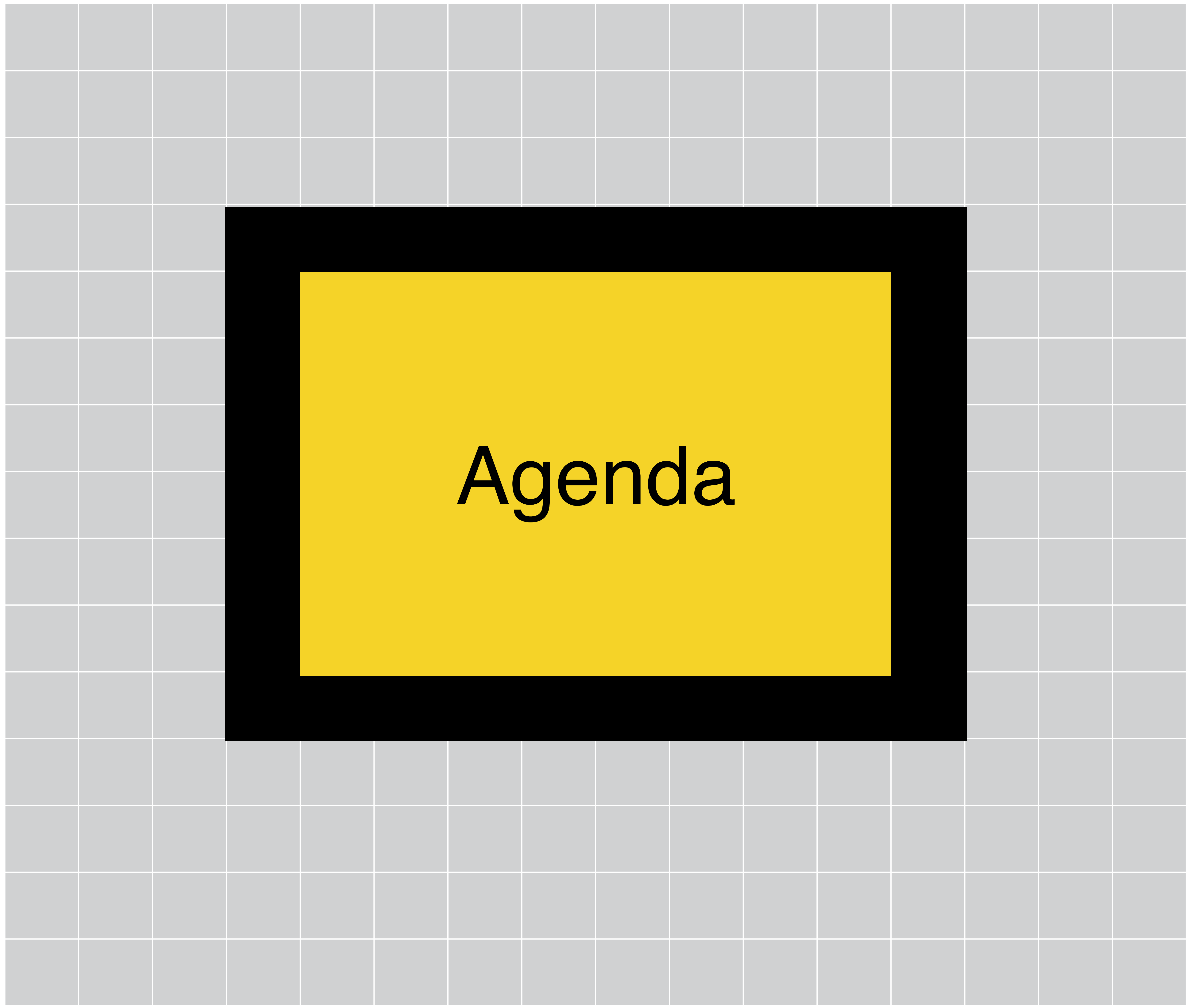


Margins



Margins

Set the size of white space outside of the element border.



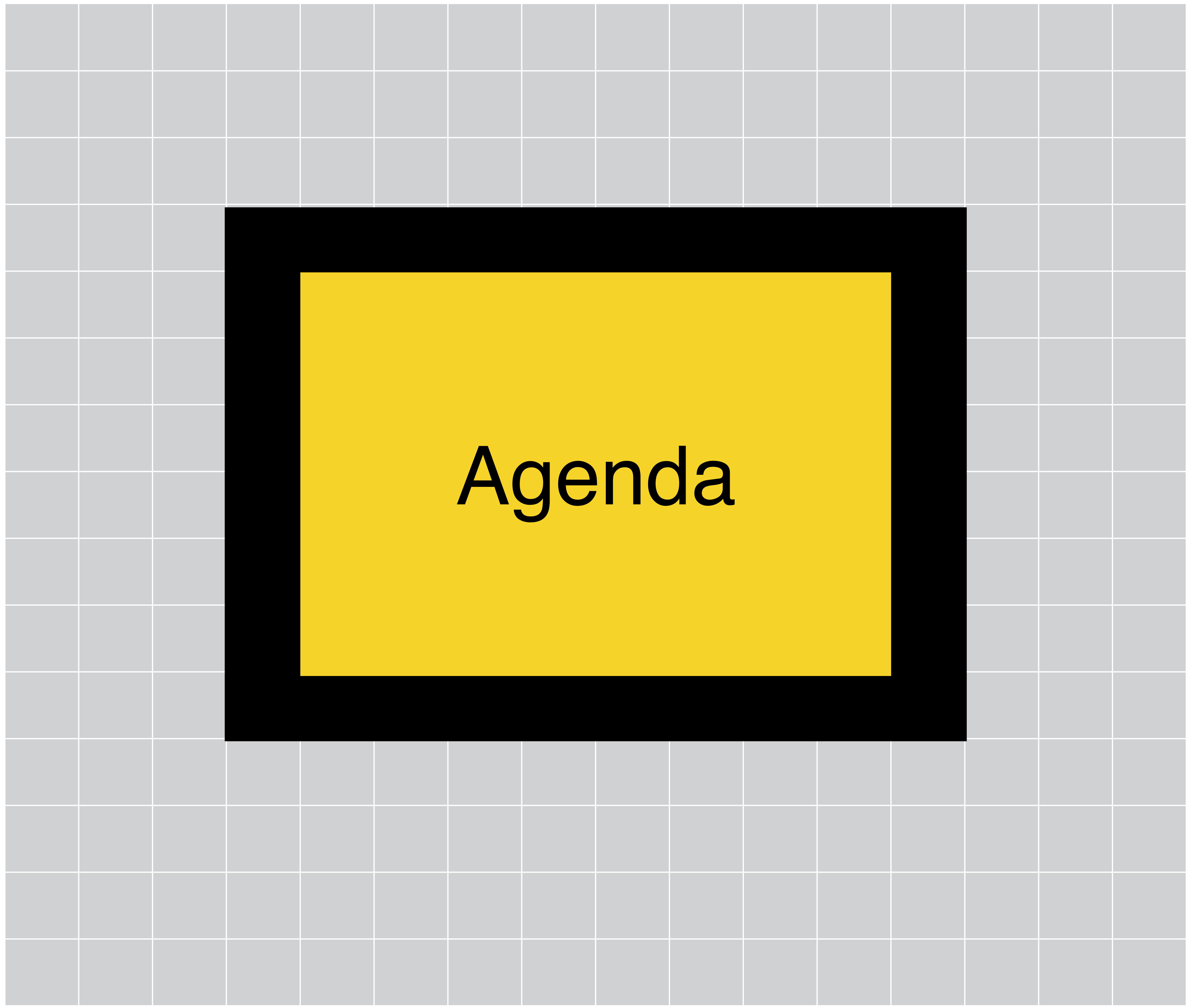
Margins

Set the size of white space outside of the element border.

<h1 class=“yellow”**>**Agenda**</h1>**

```
<style>
 .yellow {
  background-color: yellow;
  border: 1px solid:
  padding: 2px;

  }
</style>
```

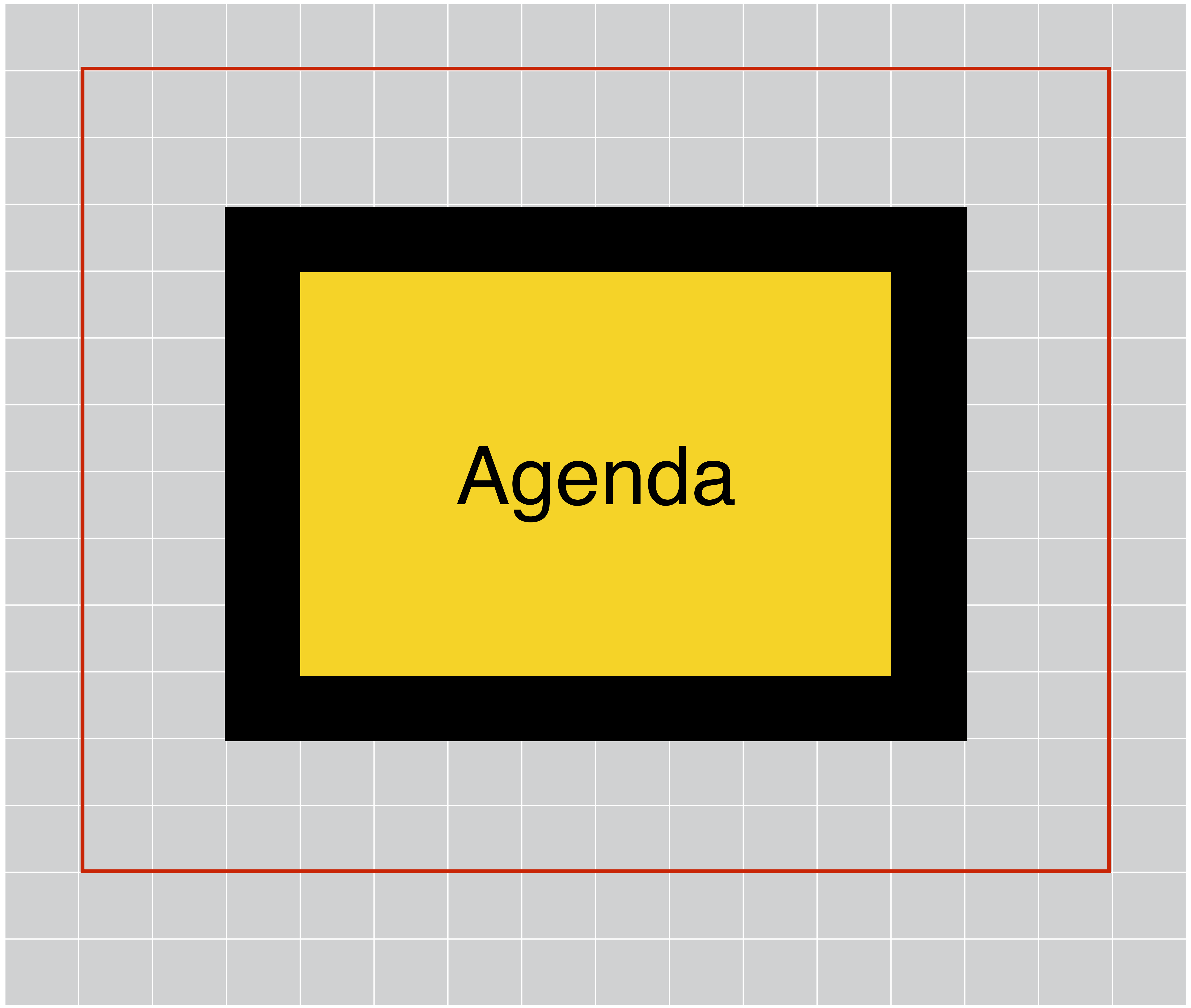


Margins

Set the size of white space outside of the element border.

<h1 class="yellow"**>**Agenda**</h1>**

```
<style>
 .yellow {
  background-color: yellow;
  border: 1px solid;
  padding: 2px;
  margin: 2px;
  }
</style>
```



Margins

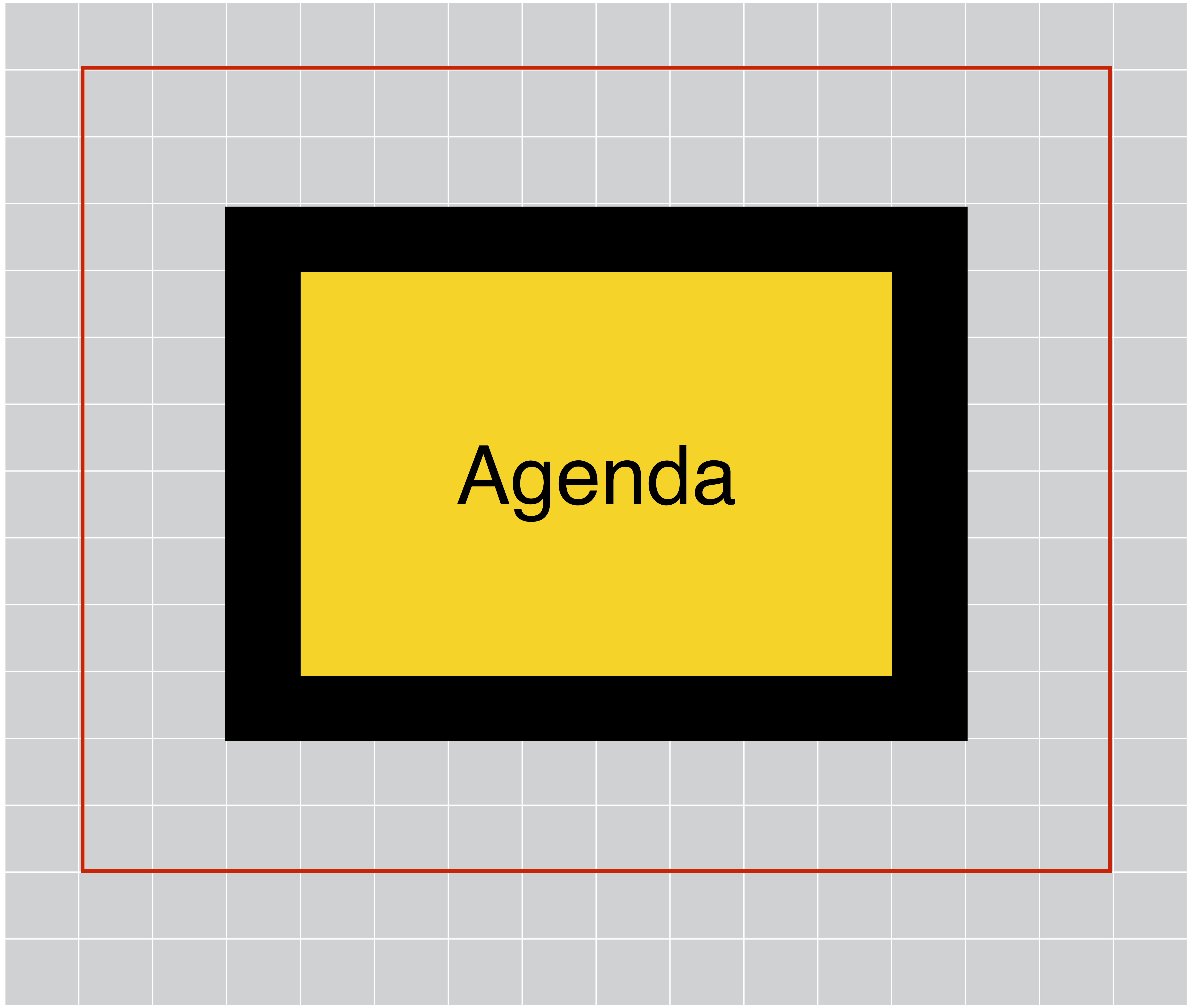
:{) Codaisseur

Set the size of white space outside of the element border.

<h1 class=“yellow”**>**Agenda**</h1>**

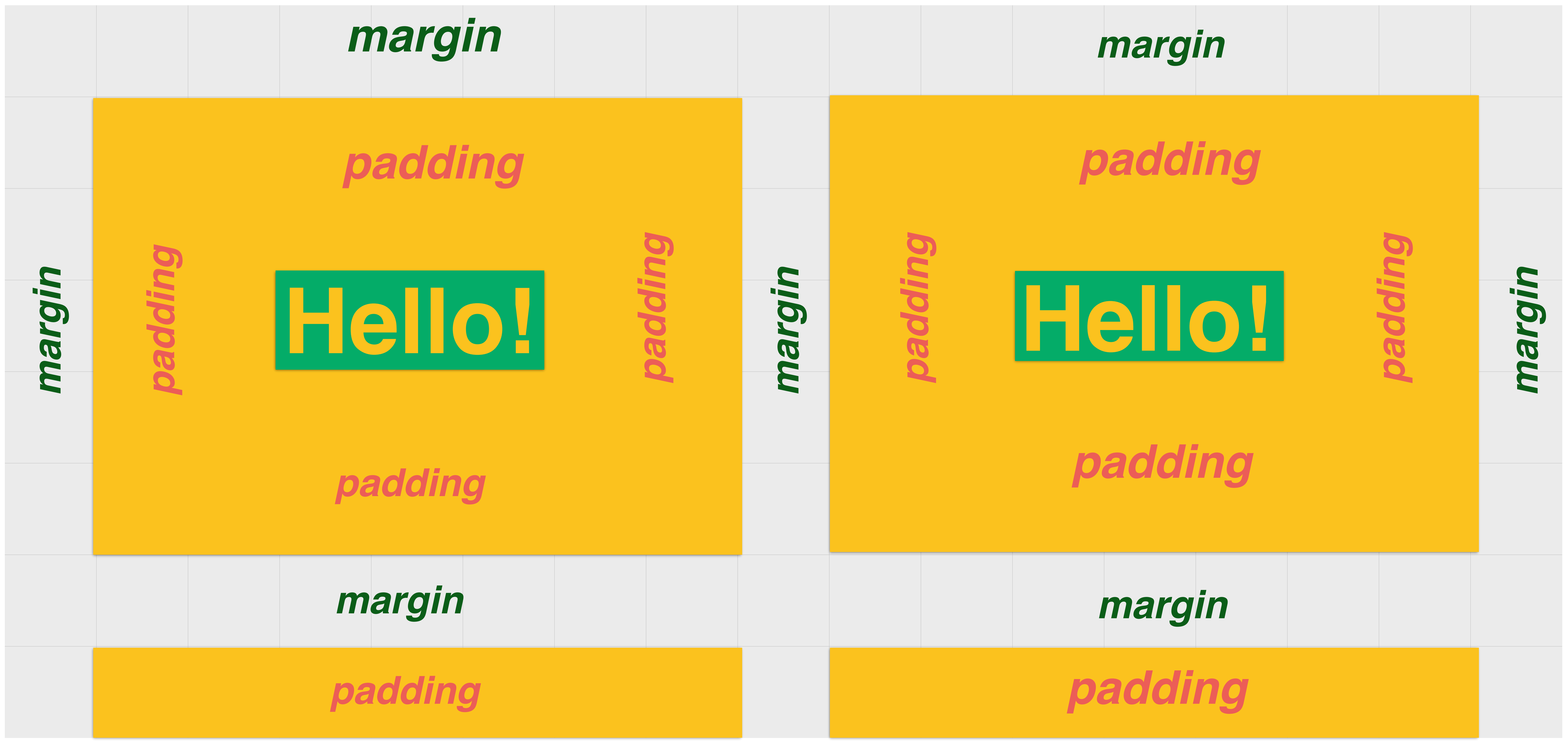
```
<style>
 .yellow {
  background-color: yellow;
  border: 1px solid;
  padding: 2px;
  margin: 2px;
  }
</style>
```

margin-left, margin-right
margin-top, margin-bottom



Padding & Margin in Pixels

:{) Codaisseur



✎ Exercise

Styling

Apply some margins and paddings to
the elements in your HTML document.

It’s time to start over!

It’s time to start over!

```
<!DOCTYPE html>
<html>
  <head>
    <title></title>
  </head>
  <body>
  </body>
</html>
```

Div’s

Div’s

<div>
defines a division or section in an HTML document.

Div’s

<div>
defines a division or section in an HTML document.

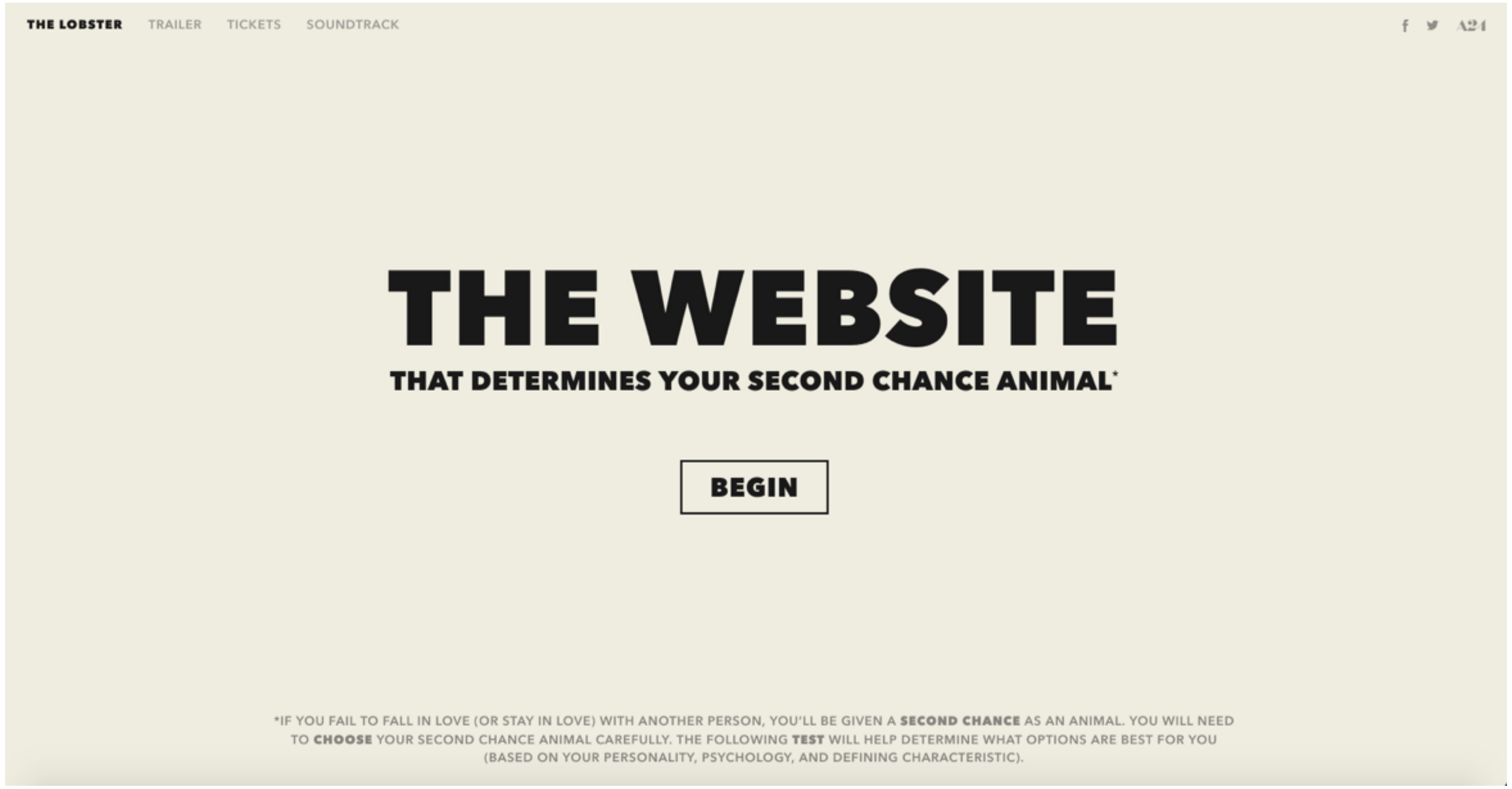
Used to group elements together for styling and/or scripting reasons (ie. animation).

Div’s

:{) Codaisseur

<div>
defines a division or section in an HTML document.

Used to group elements together for styling and/or scripting reasons (ie. animation).



THE WEBSITE

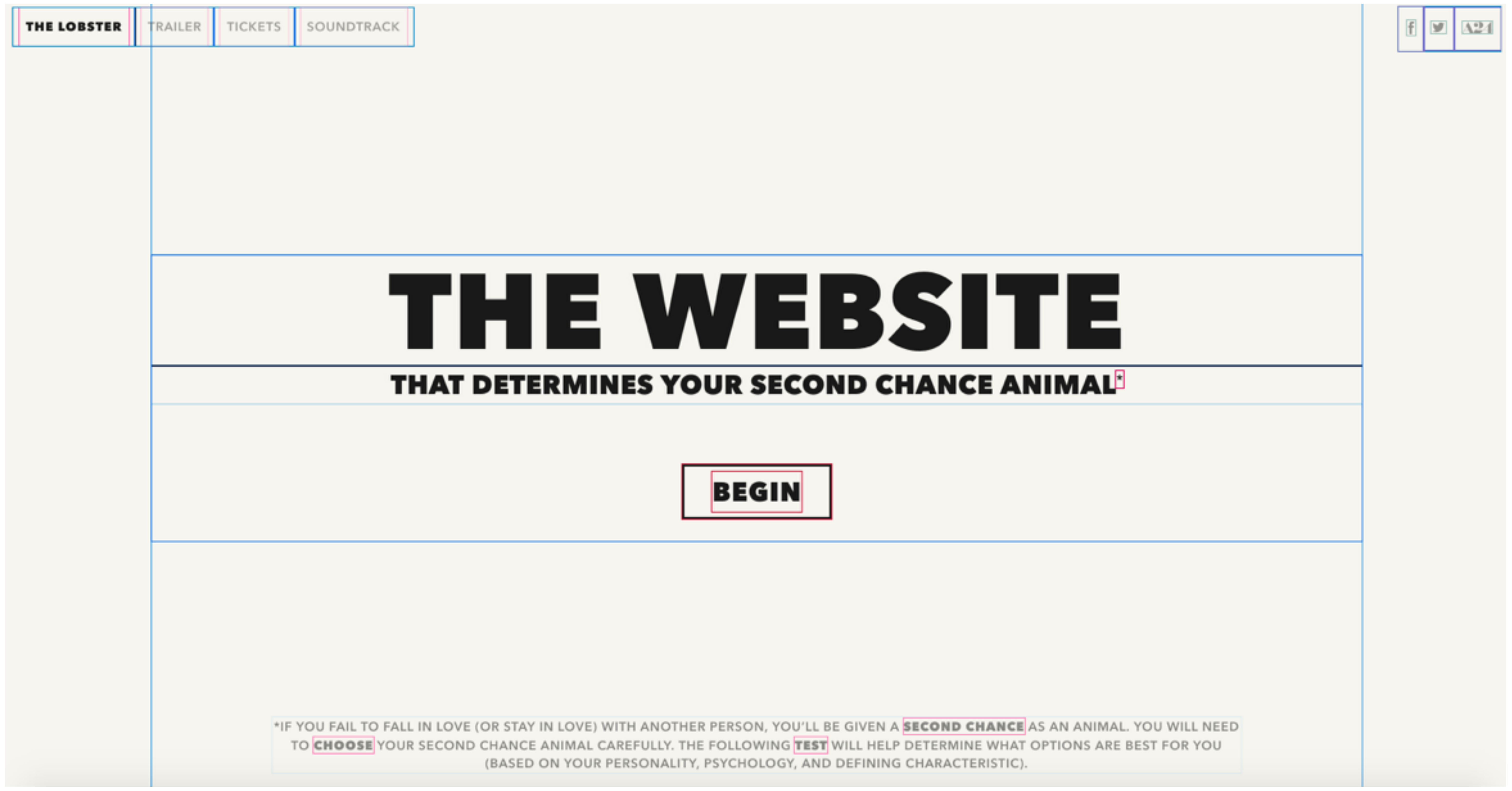
THAT DETERMINES YOUR SECOND CHANCE ANIMAL*

*IF YOU FAIL TO FALL IN LOVE (OR STAY IN LOVE) WITH ANOTHER PERSON, YOU’LL BE GIVEN A **SECOND CHANCE** AS AN ANIMAL. YOU WILL NEED TO **CHOOSE** YOUR SECOND CHANCE ANIMAL CAREFULLY. THE FOLLOWING **TEST** WILL HELP DETERMINE WHAT OPTIONS ARE BEST FOR YOU (BASED ON YOUR PERSONALITY, PSYCHOLOGY, AND DEFINING CHARACTERISTIC).

Div's

<div>
defines a division or section in an HTML document.

Used to group elements together for styling and/or scripting reasons (ie. animation).



THE LOBSTER | TRAILER | TICKETS | SOUNDTRACK

THE WEBSITE

THAT DETERMINES YOUR SECOND CHANCE ANIMAL*

BEGIN

*IF YOU FAIL TO FALL IN LOVE (OR STAY IN LOVE) WITH ANOTHER PERSON, YOU'LL BE GIVEN A **SECOND CHANCE** AS AN ANIMAL. YOU WILL NEED TO **CHOOSE** YOUR SECOND CHANCE ANIMAL CAREFULLY. THE FOLLOWING **TEST** WILL HELP DETERMINE WHAT OPTIONS ARE BEST FOR YOU (BASED ON YOUR PERSONALITY, PSYCHOLOGY, AND DEFINING CHARACTERISTIC).

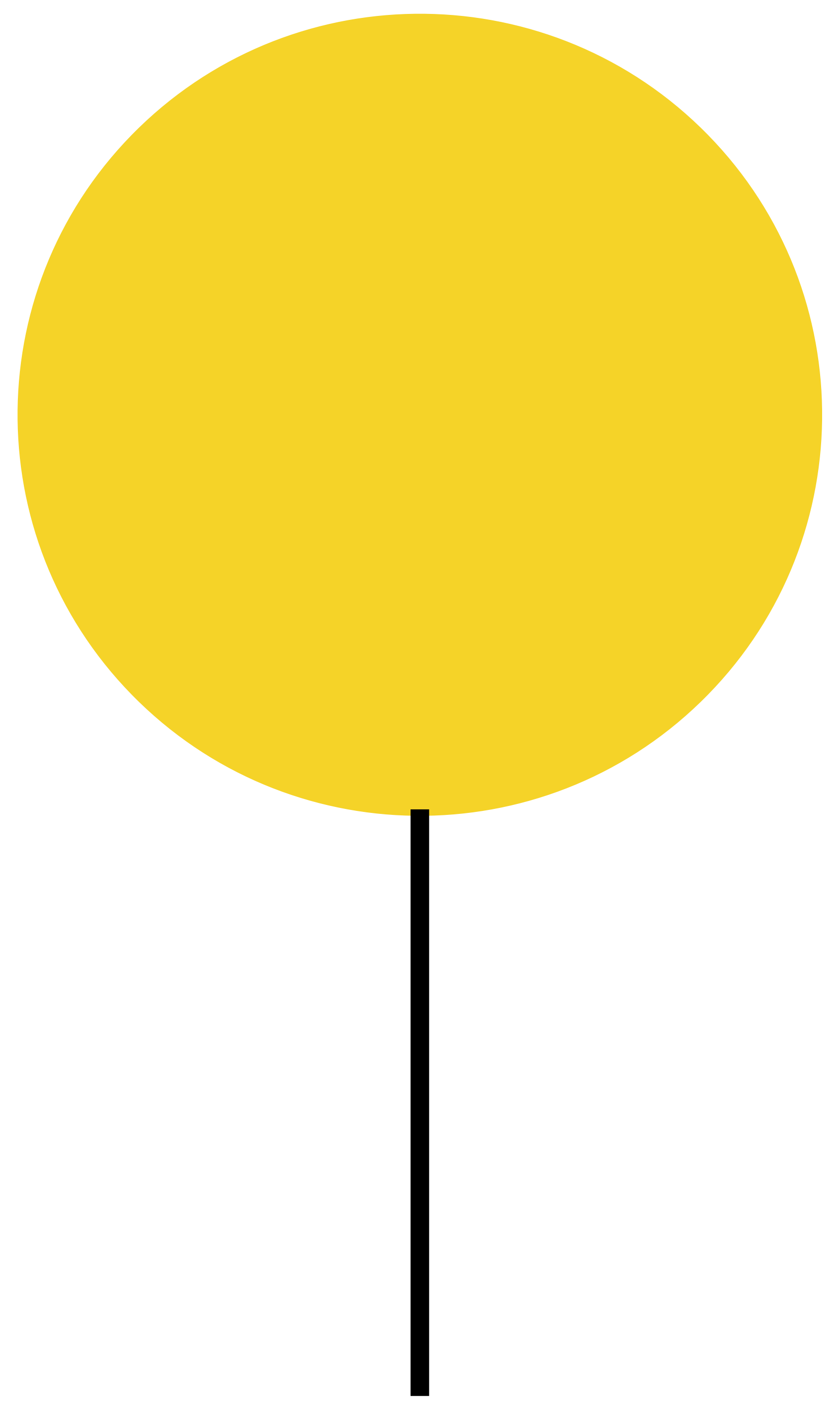
Div’s

Div’s

“Div”ing a balloon

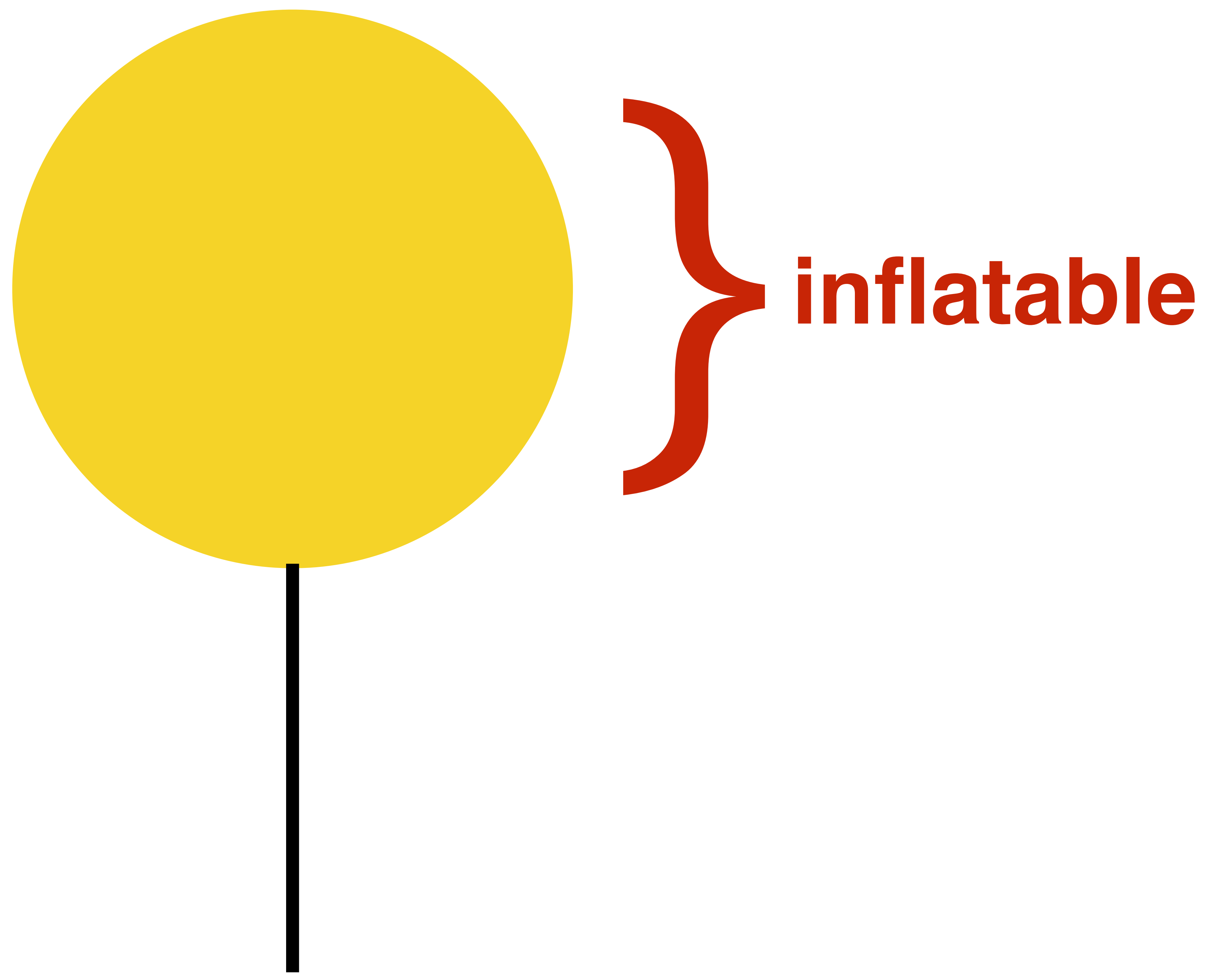
Div's

"Div"ing a balloon



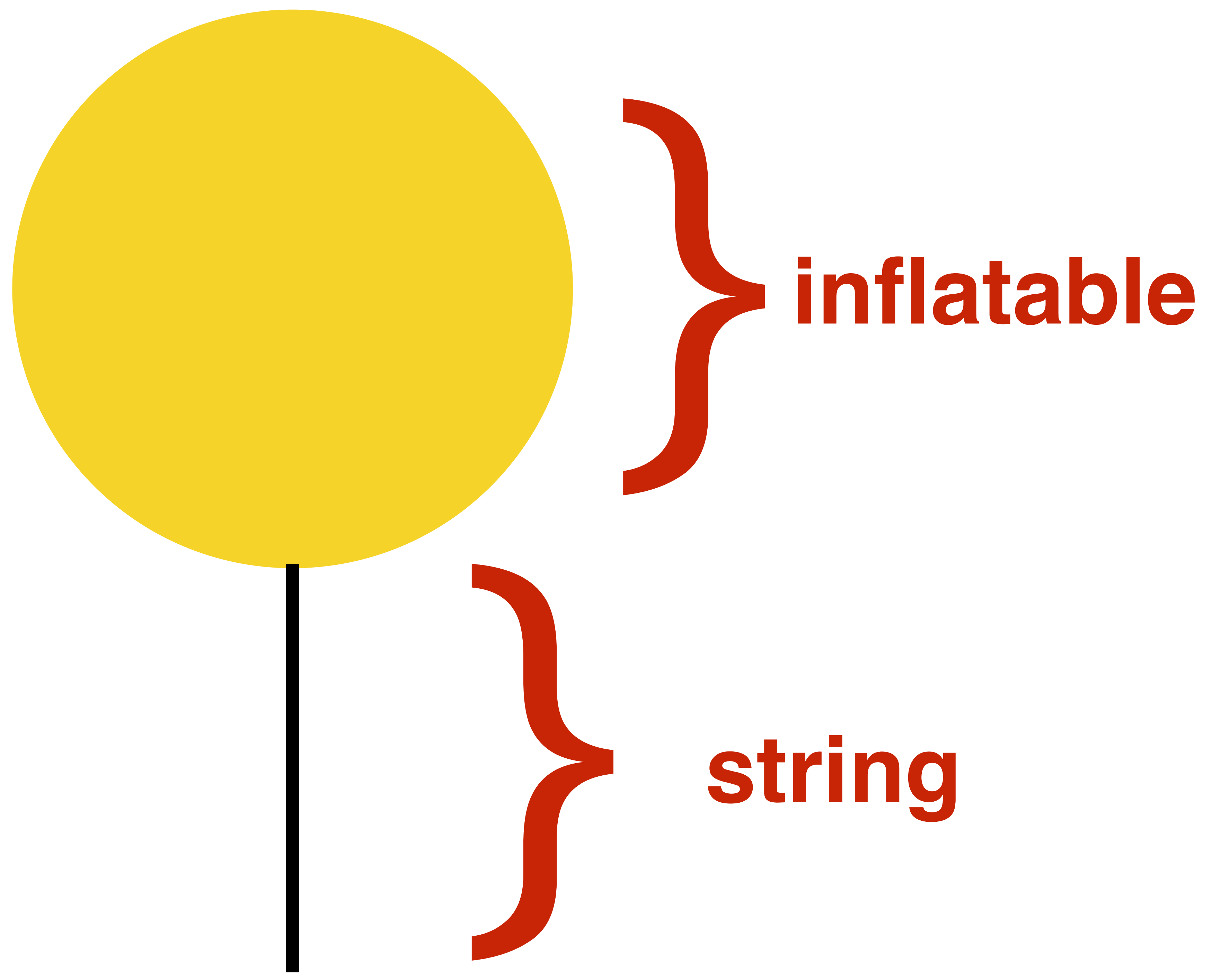
Div’s

“Div”ing a balloon



Div’s

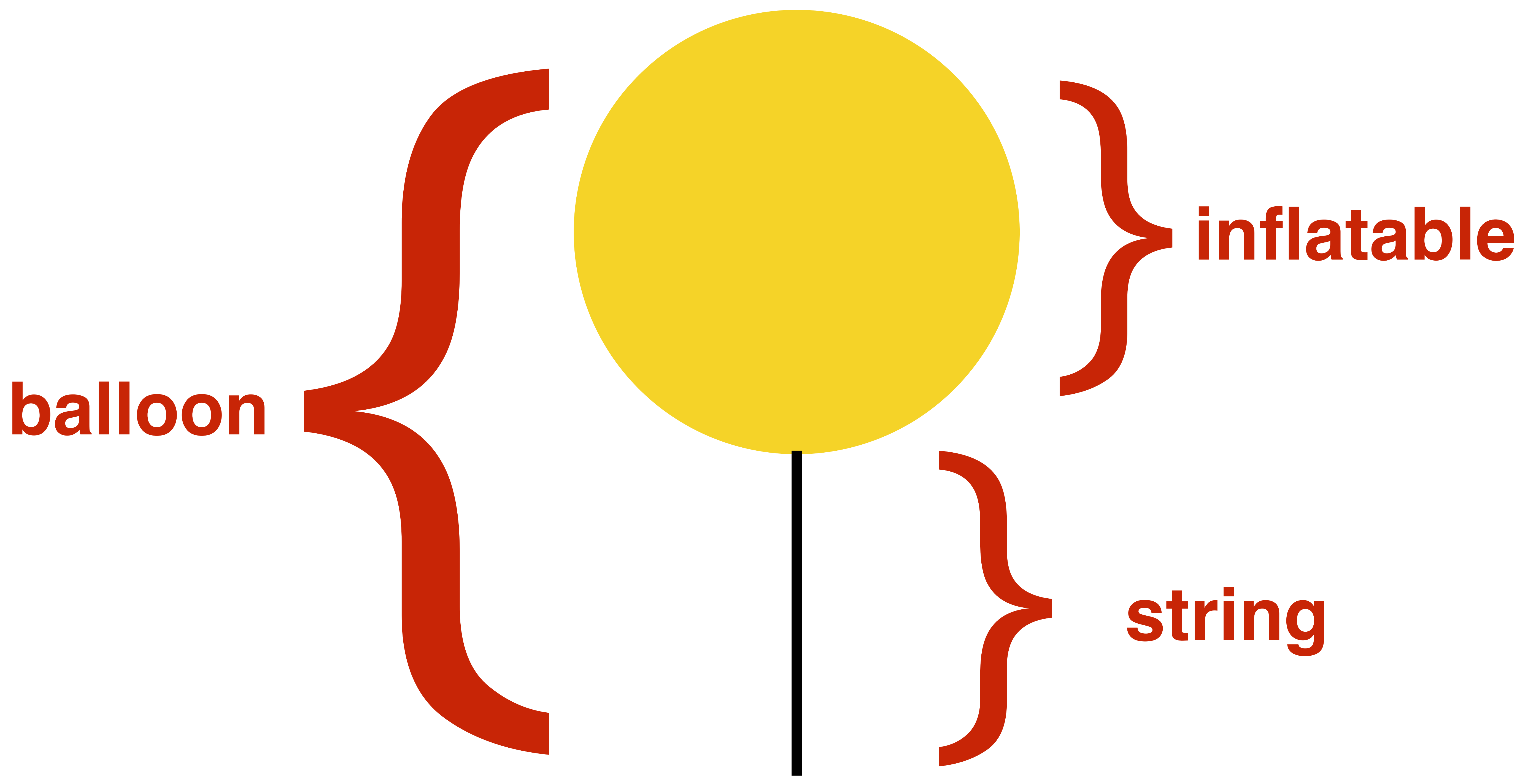
“Div”ing a balloon



Div’s

:{) Codaisseur

“Div”ing a balloon

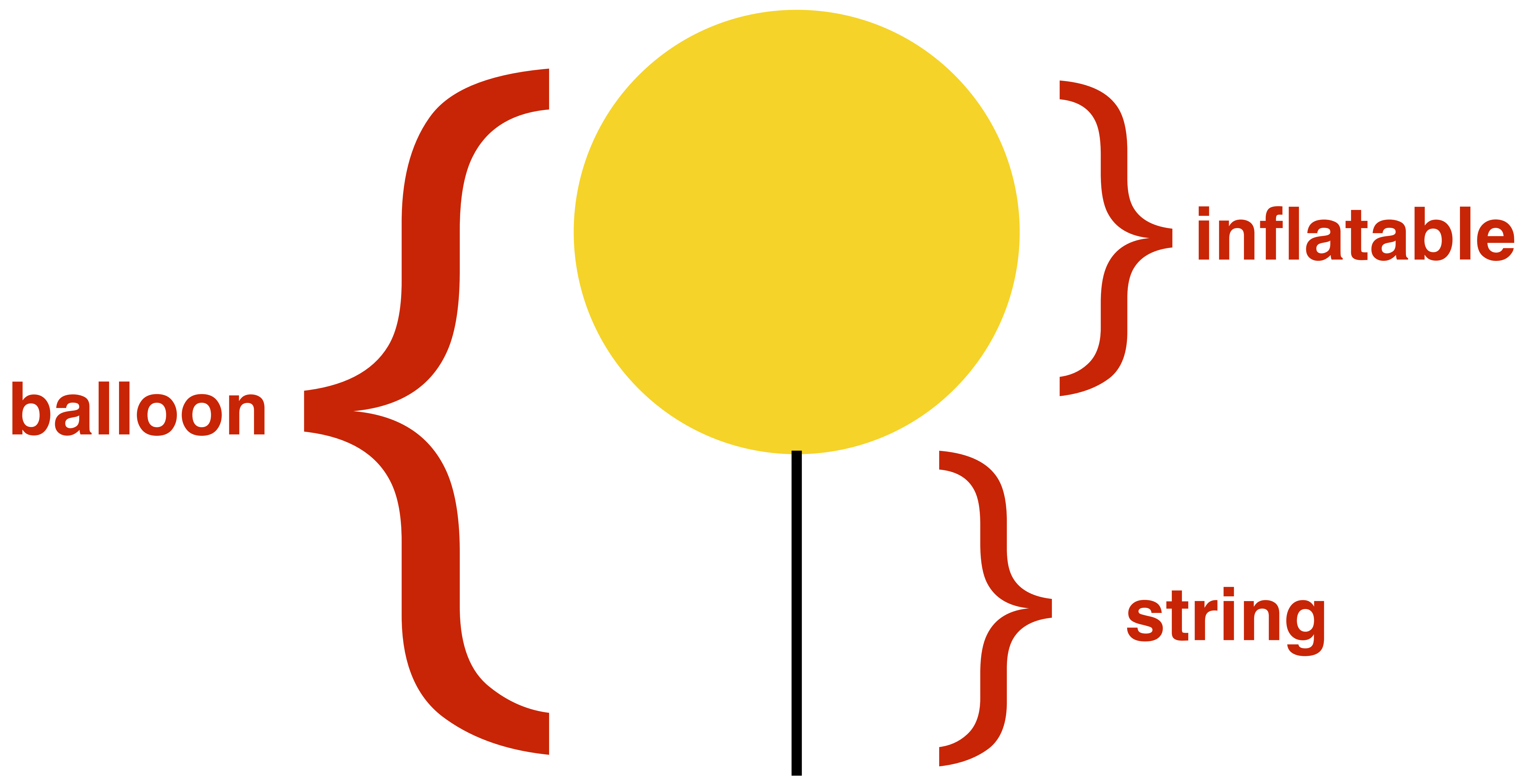


Div’s

:{) Codaisseur

```
<div class=“balloon”>
 <div class=“inflatable”>
  </div>
  <div class=“string”>
  </div>
</div>
```

“Div”ing a balloon



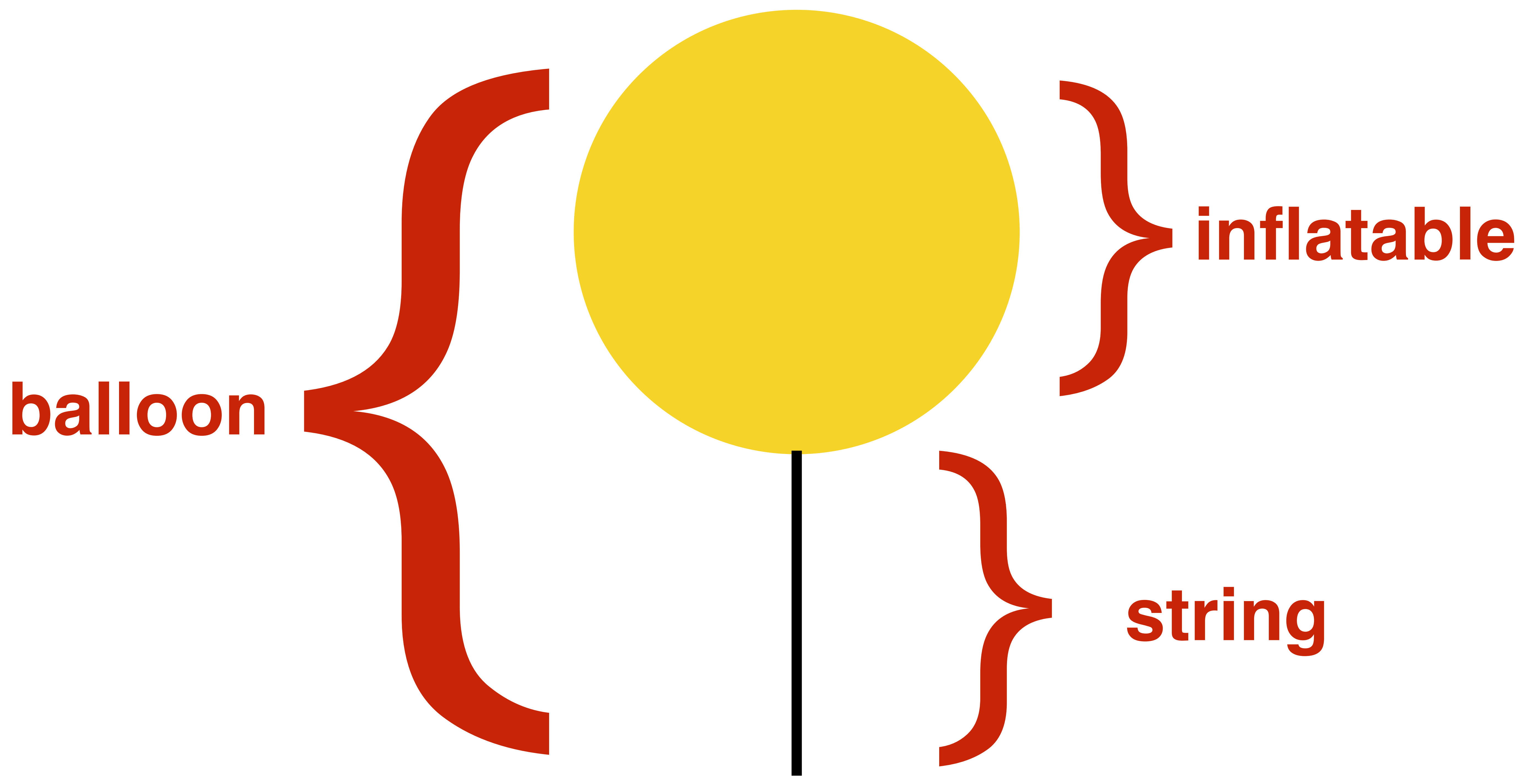
Div’s

:{) Codaisseur

“Div”ing a balloon

```
<div class=“balloon”>
 <div class=“inflatable”>
 </div>
 <div class=“string”>
 </div>
</div>
```

In atom, place this code between your body tags

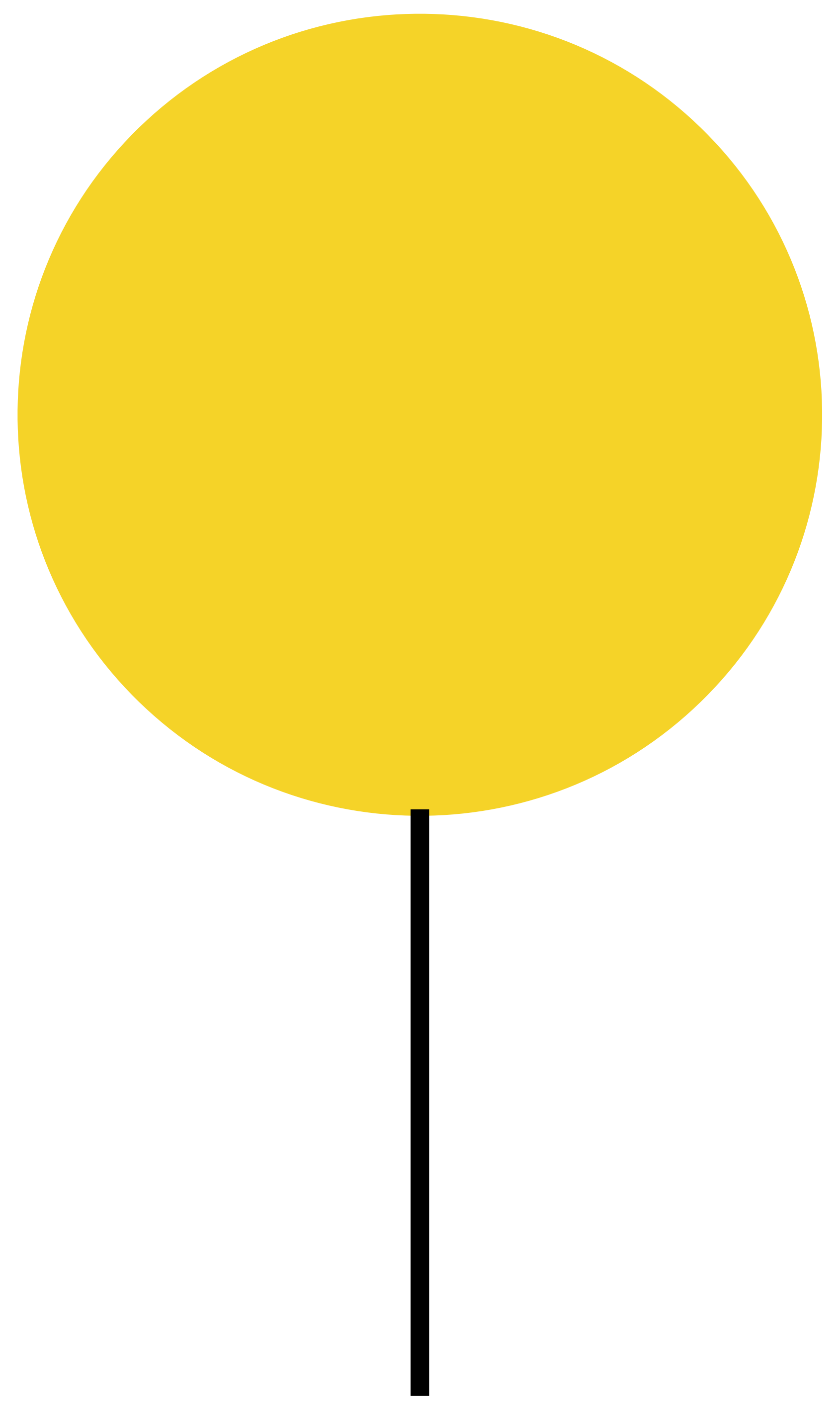


Div’s

Div’s

Styling the balloon

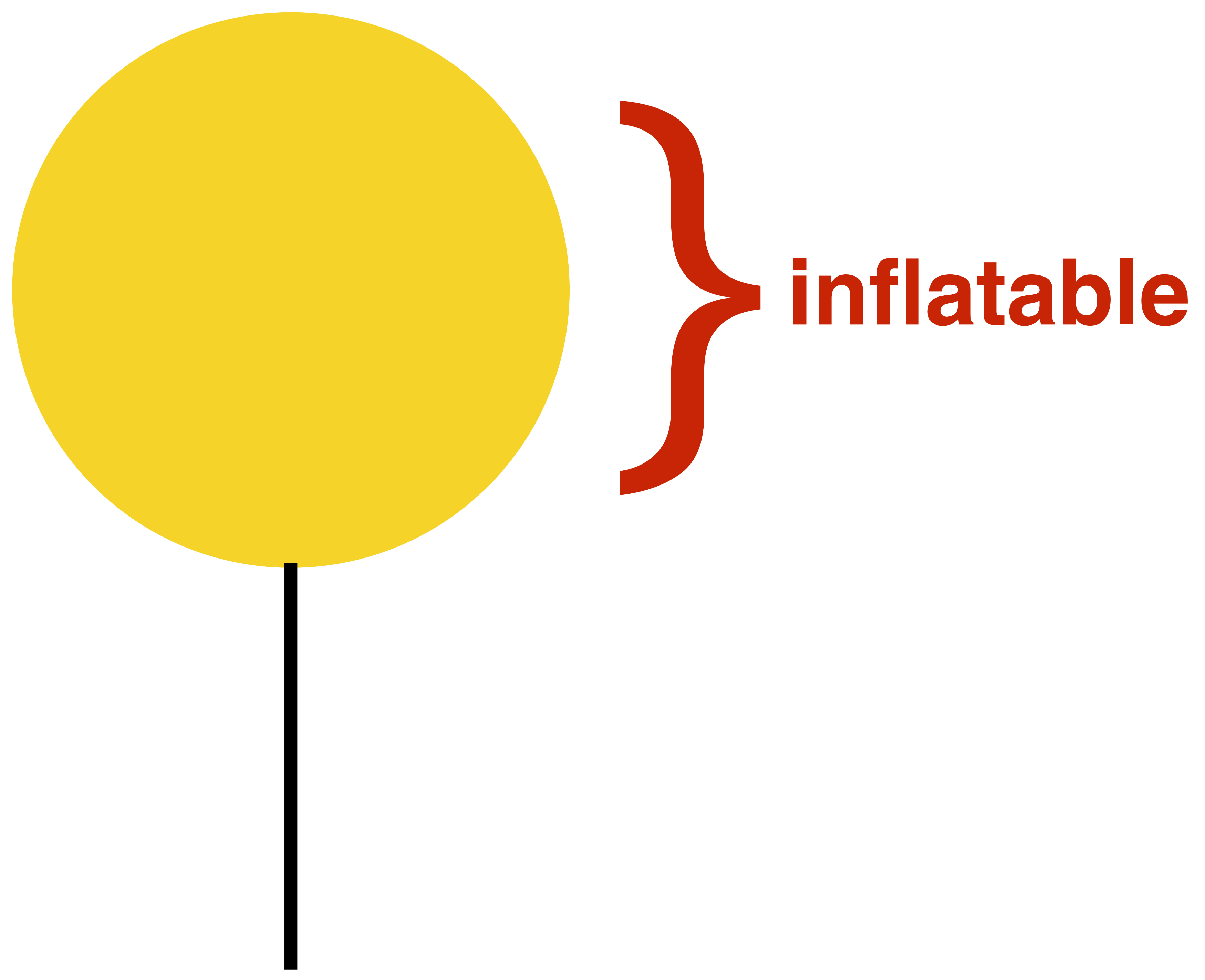
Styling the balloon



Div's

Codaisseur

Styling the balloon

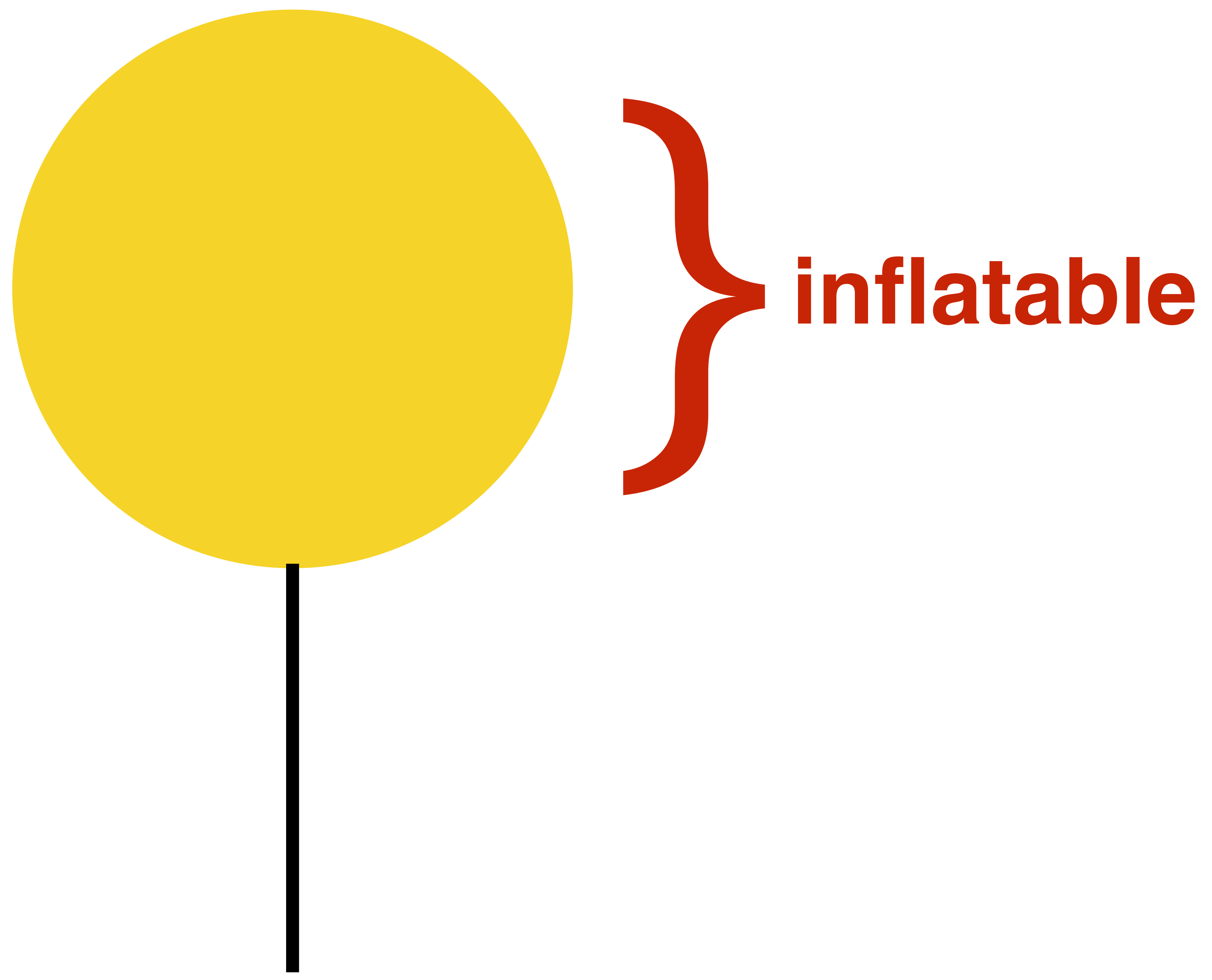


Div’s

:{) Codaisseur

```
.inflatable {
 width: 180px;
 height: 200px;
 background-color: yellow;
 border-radius: 50%;
 }
```

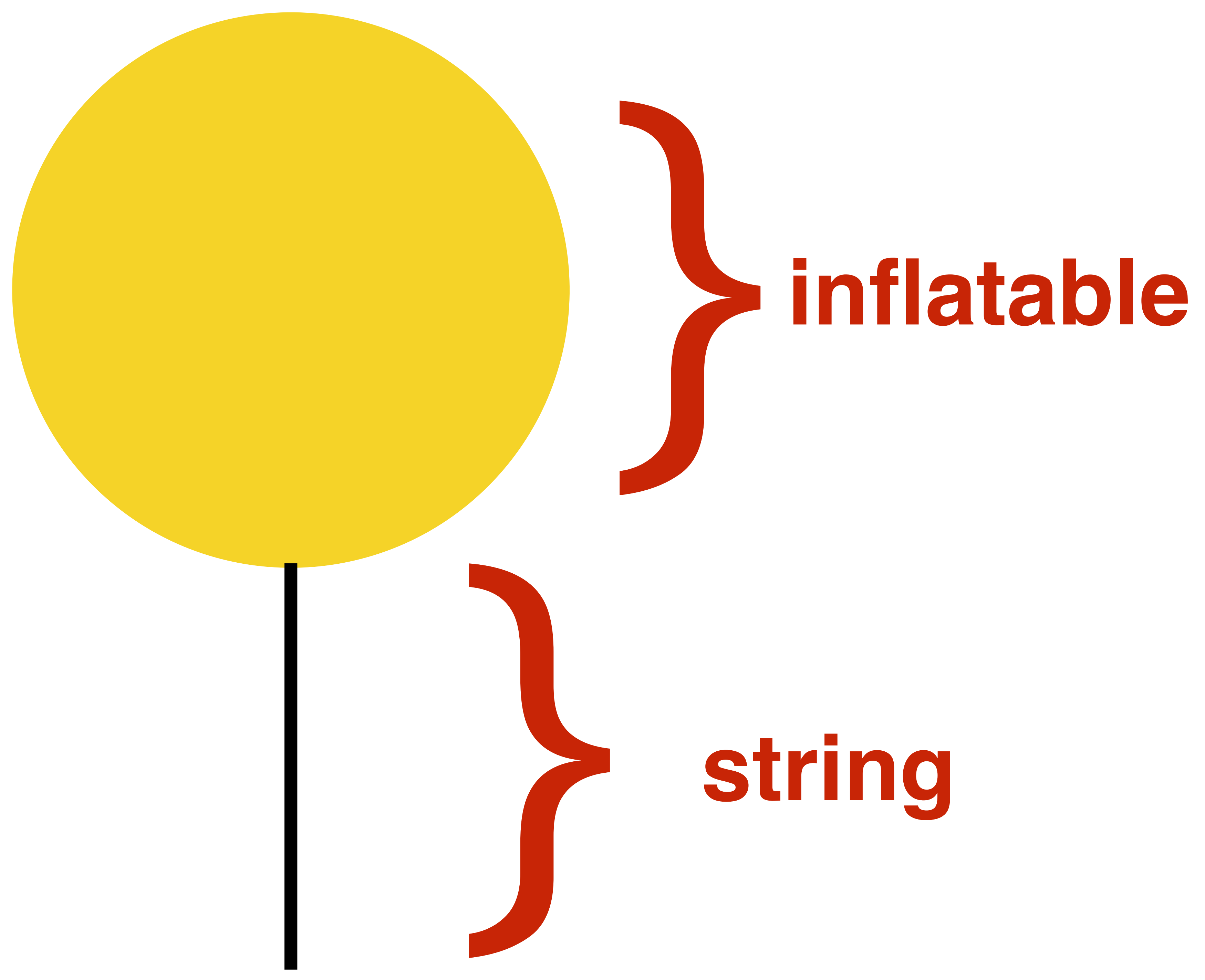
Styling the balloon



Div’s

```
.inflatable {
 width: 180px;
 height: 200px;
 background-color: yellow;
 border-radius: 50%;
 }
```

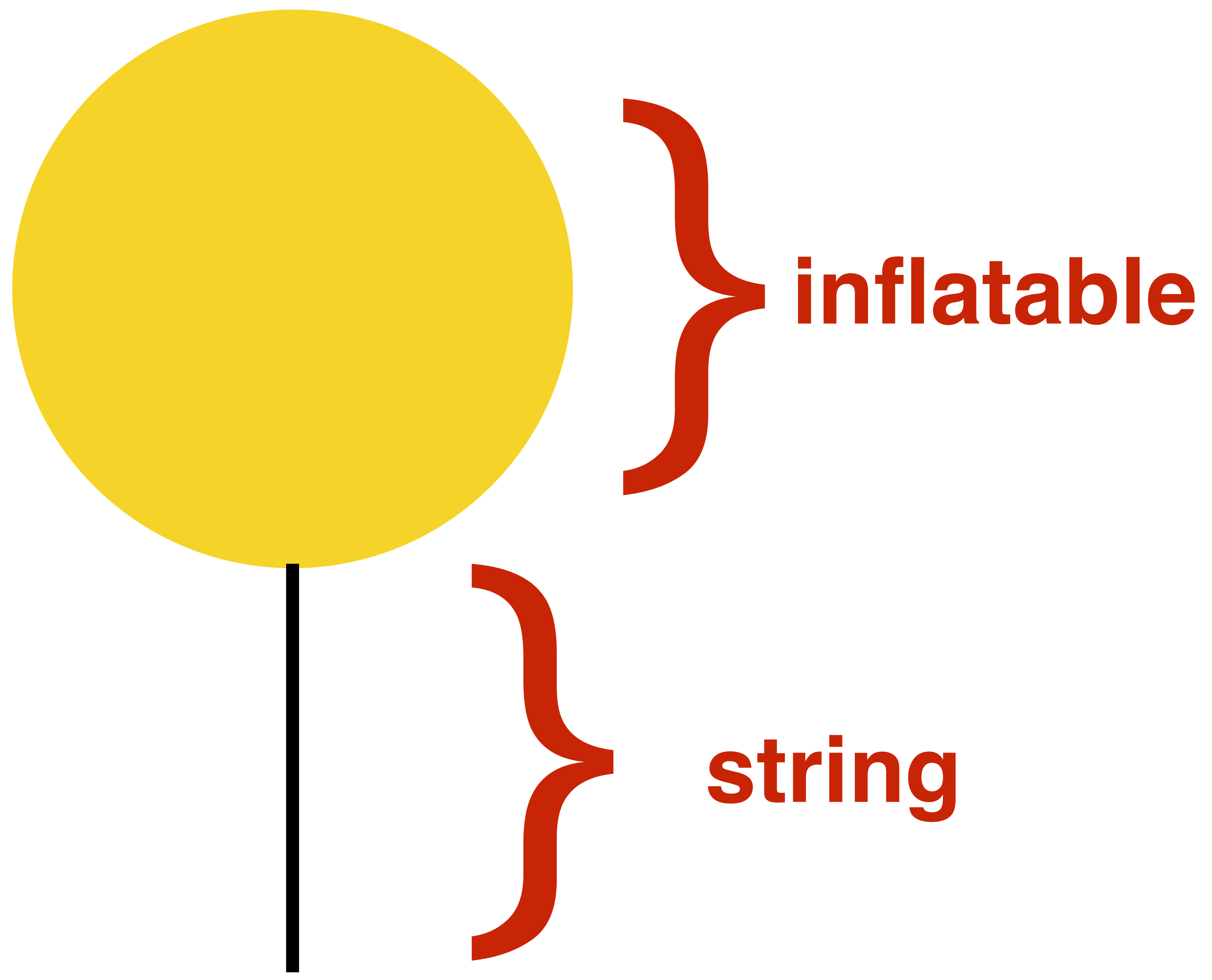
Styling the balloon



Div's

```
.inflatable {
  width: 180px;
  height: 200px;
  background-color: yellow;
  border-radius: 50%;
  }
.string {
  width: 1px;
  height: 100px;
  background-color: black;
  margin-left: 90px;
  }
```

Styling the balloon

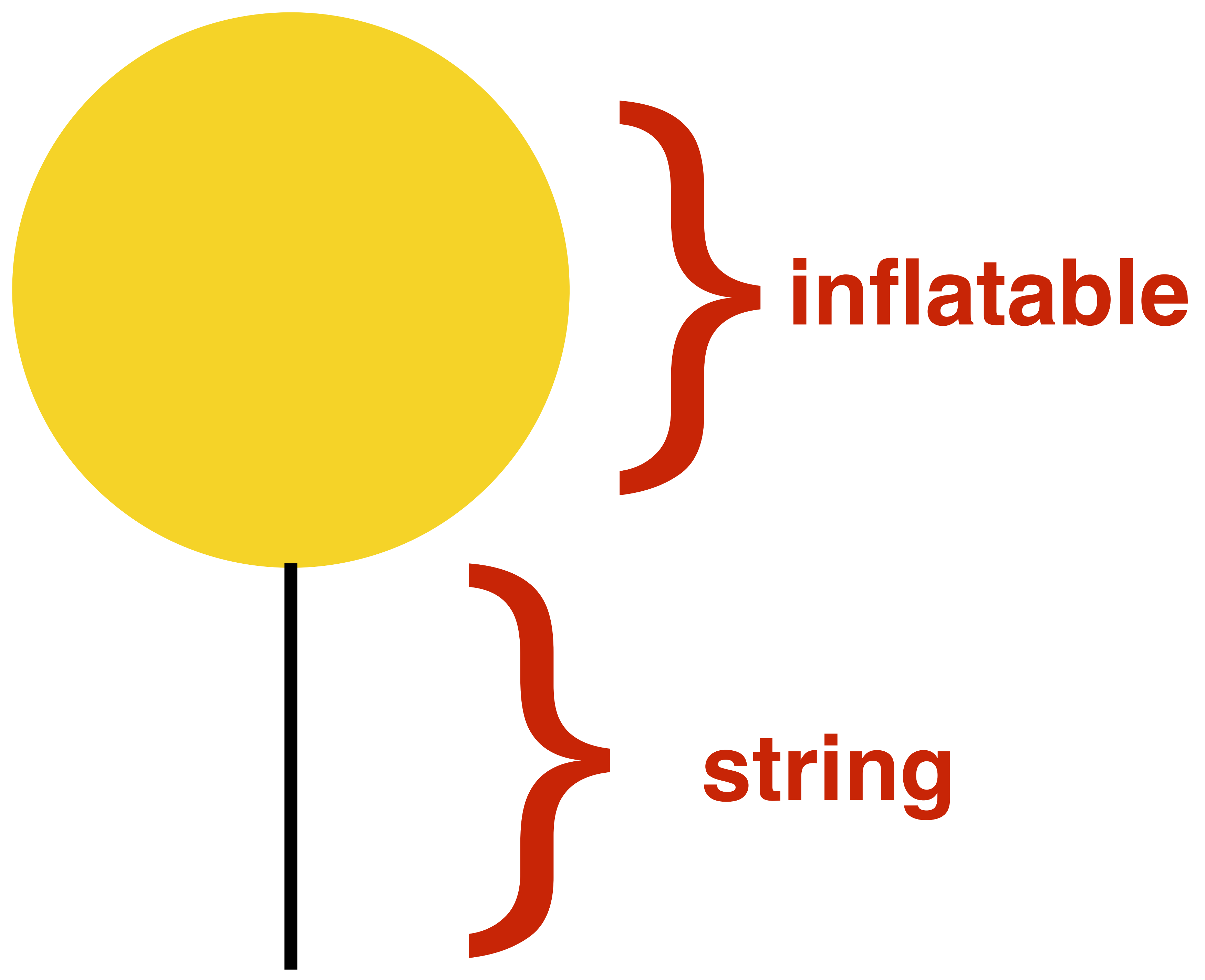


Div’s

:{) Codaisseur

```
.inflatable {
  width: 180px;
  height: 200px;
  background-color: yellow;
  border-radius: 50%;
  }
.string {
  width: 1px;
  height: 100px;
  background-color: black;
  margin-left: 90px;
  }
```

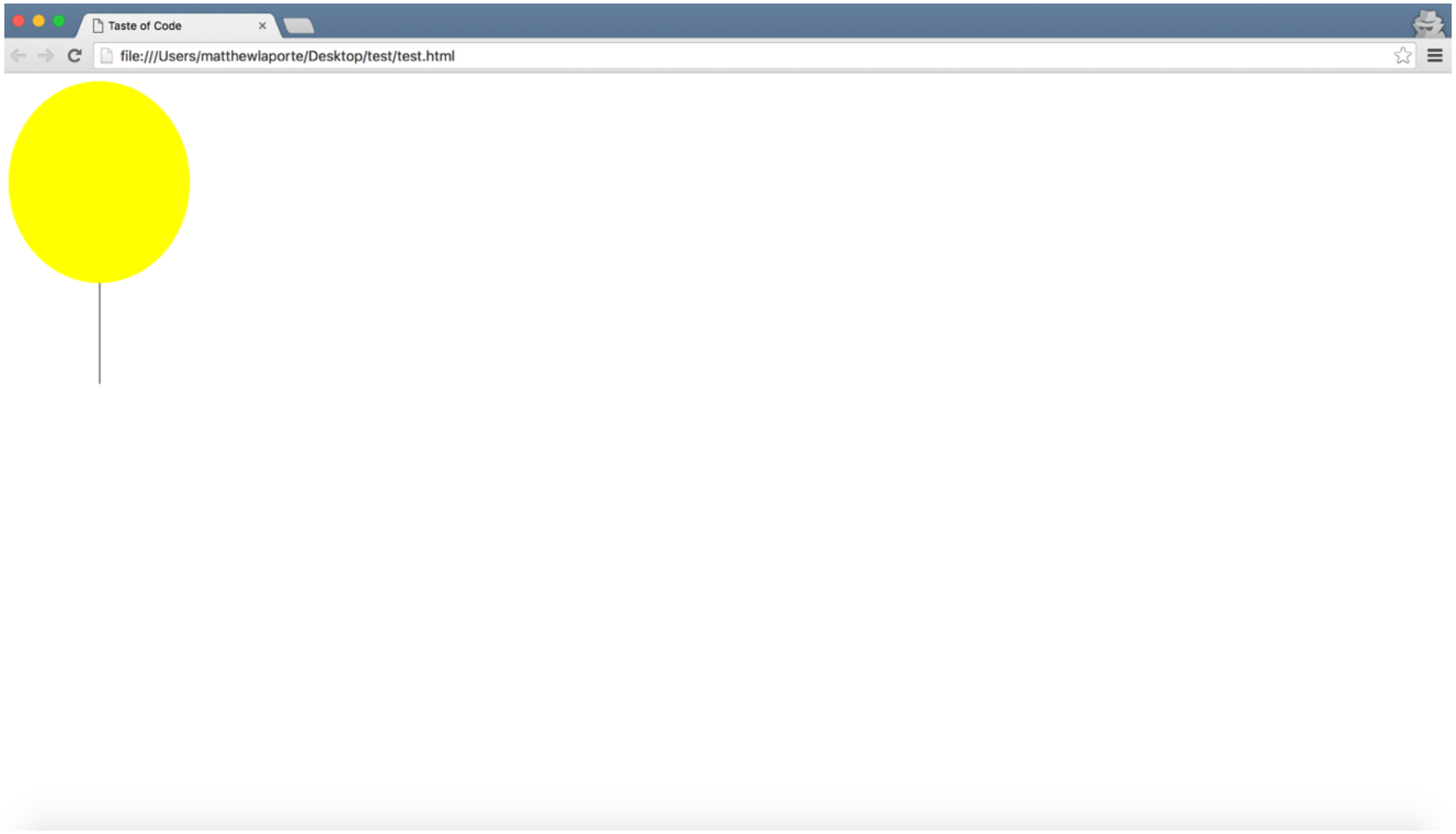
Styling the balloon



In atom, place this code between style tags

Div’s

:{) Codaisseur



LUNCH

JavaScript

Behaviour layer

This is where we can change **how the page behaves,** adding **interactivity**.

What is Javascript?

What is Javascript?

It’s a programming language that
deals with how your document behaves

What is Javascript?

:{) Codaisseur

It’s a programming language that
deals with how your document behaves

It is one of the three core technologies
of the World Wide Web

What is Javascript?

:{) Codaisseur

It's a programming language that
deals with how your document behaves

It is one of the three core technologies
of the World Wide Web

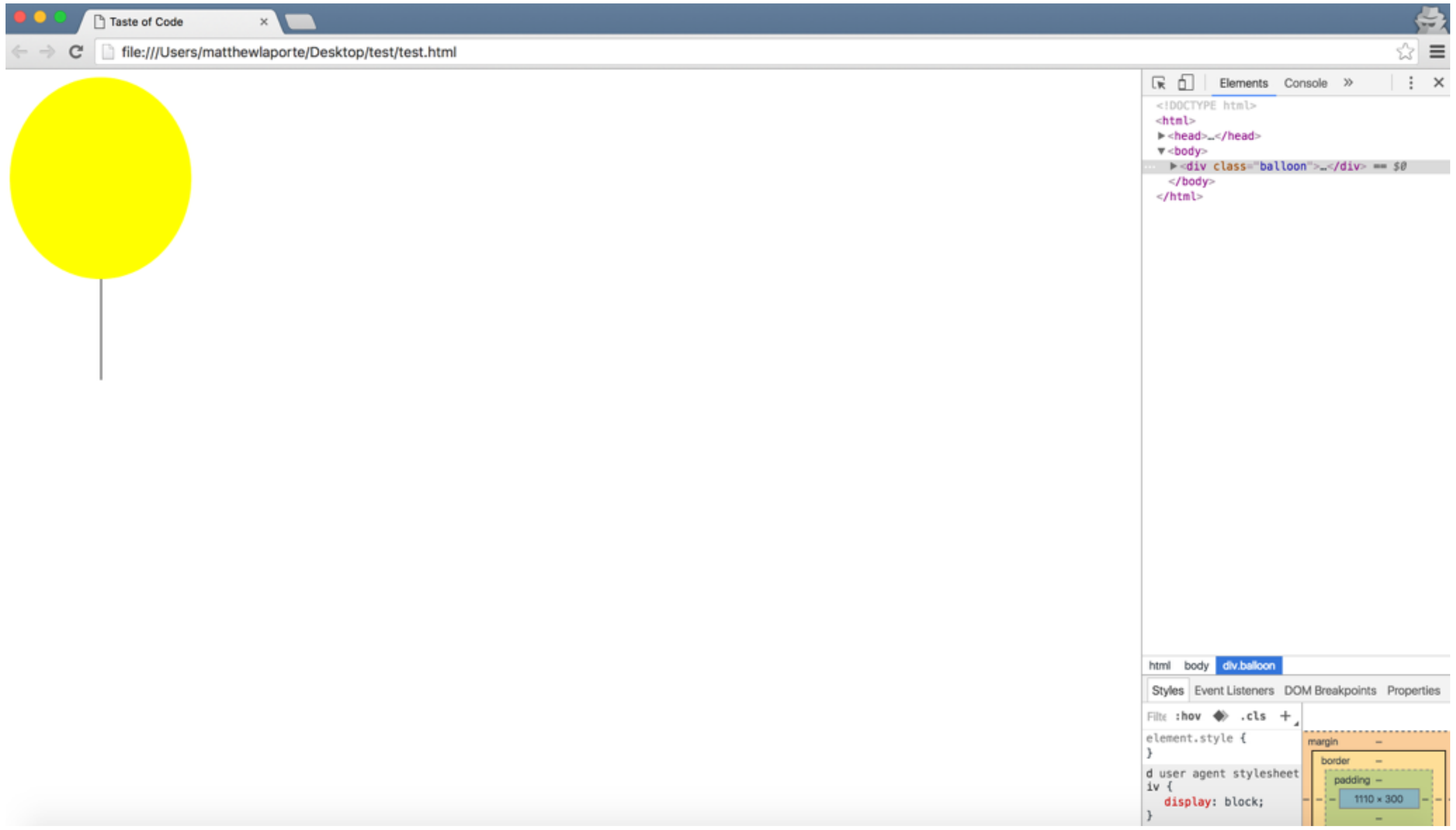
The majority of websites employ it and is
supported by all modern web browsers

JavaScript lives in your browser

:{) Codaisseur

JavaScript lives in your browser

:{) Codaisseur

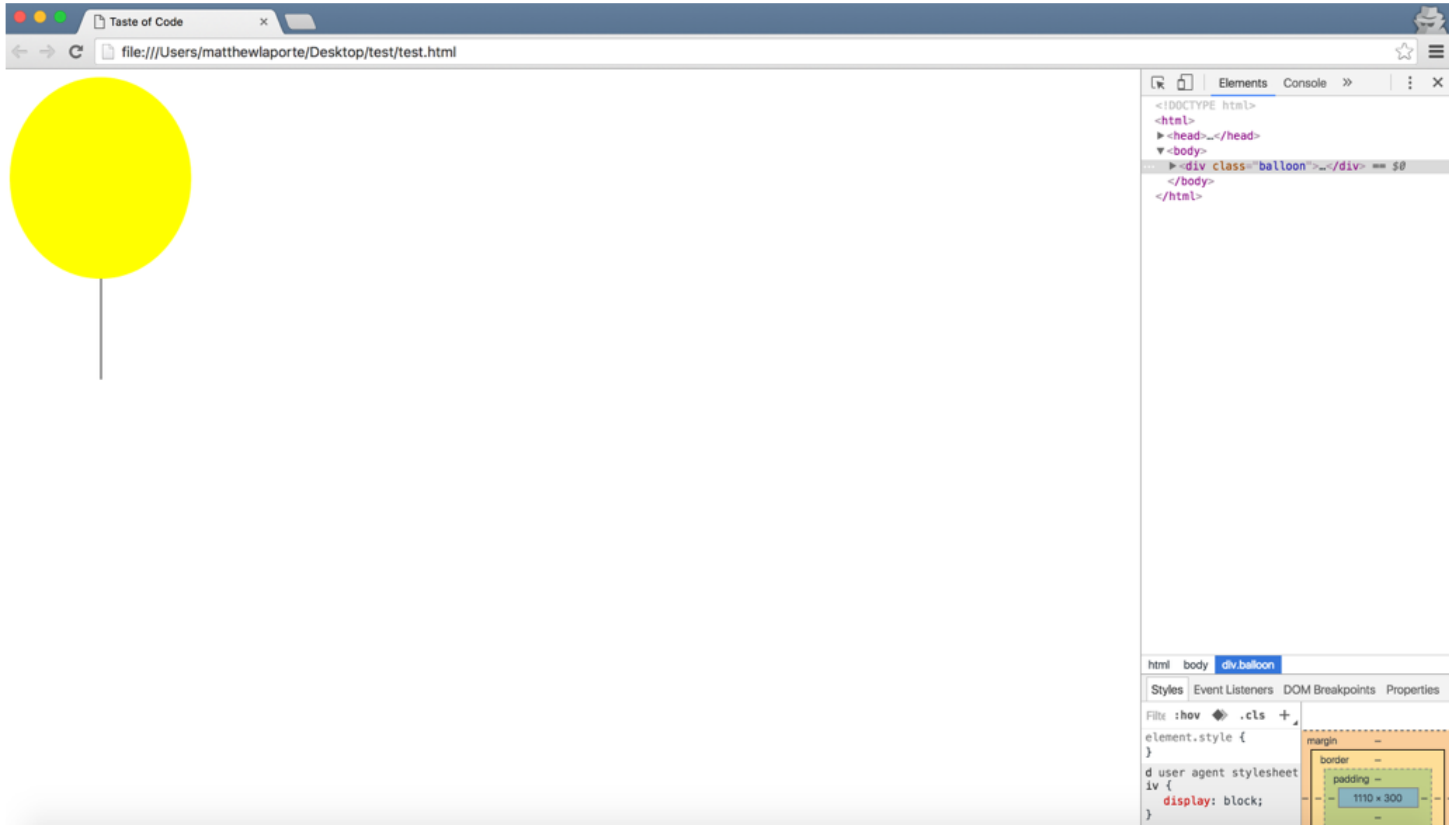


```
<!DOCTYPE html>
<html>
 ▶<head>…</head>
 ▼<body>
   ▶<div class="balloon">…</div> == $0
   </body>
</html>
```

```
element.style {
}
d user agent stylesheet
iv {
   display: block;
}
```

JavaScript lives in your browser

:{} Codaisseur



```
<!DOCTYPE html>
<html>
 ▶<head>…</head>
 ▼<body>
   ▶<div class="balloon">…</div> == $0
   </body>
</html>
```

```
element.style {
}
d user agent stylesheet
iv {
   display: block;
}
```

Open JavaScript Console

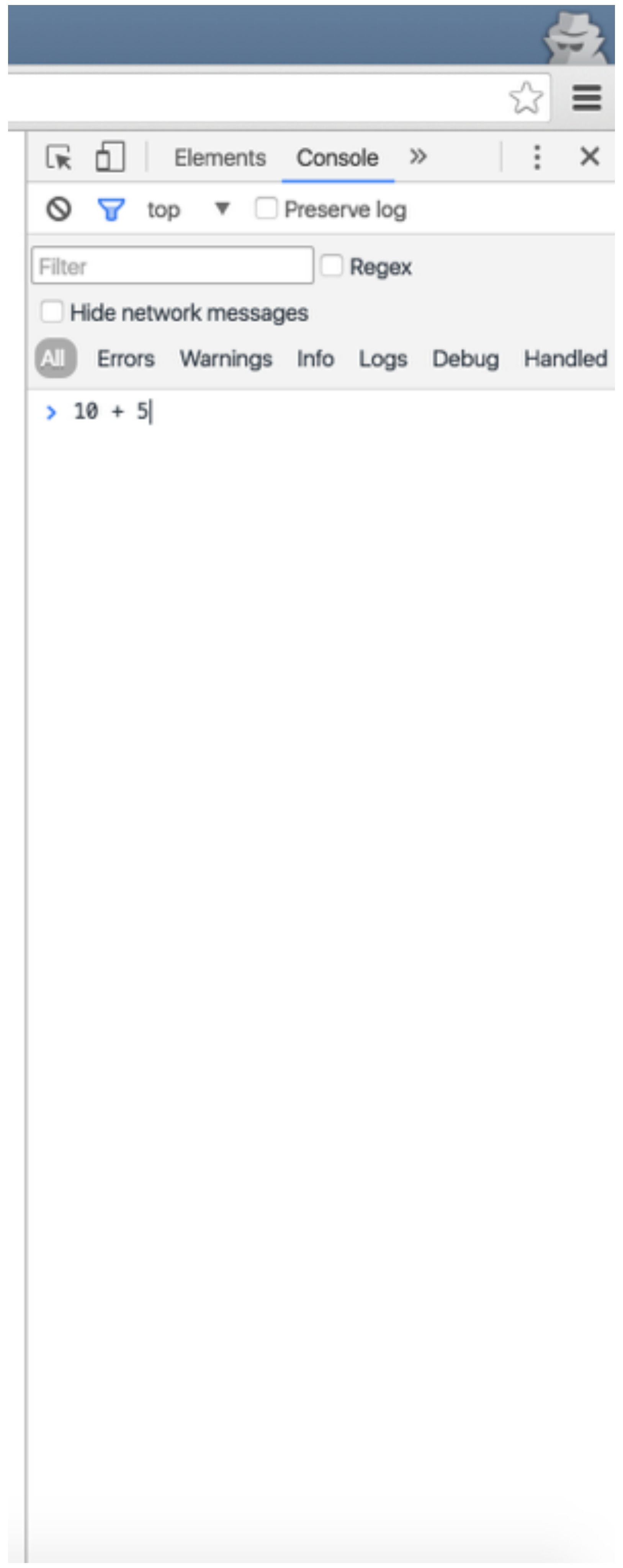
CMD + ALT + J (Mac)

Control + Shift + J (Windows/Linux)

JavaScript lives in your browser

JavaScript lives in your browser

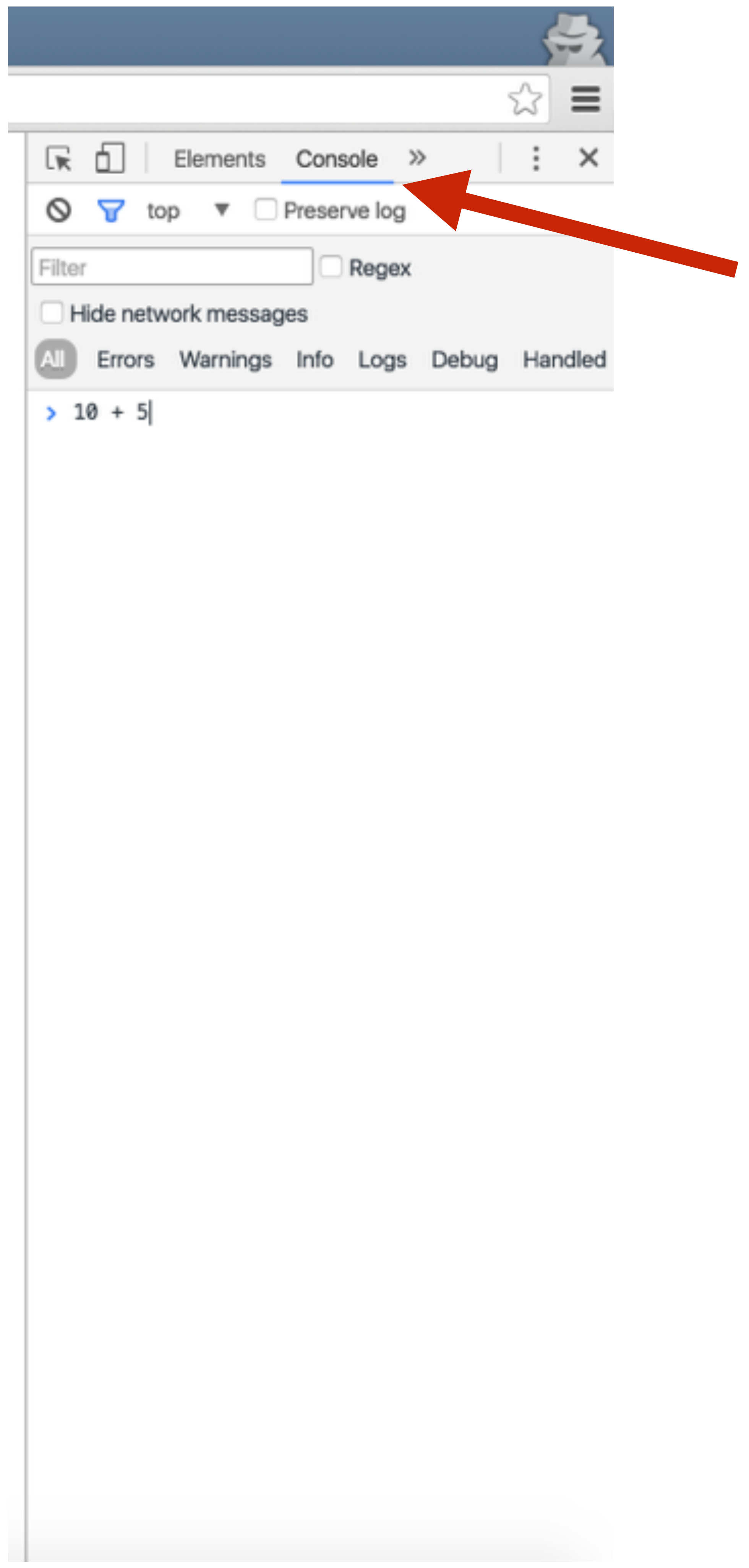
:{} Codaisseur



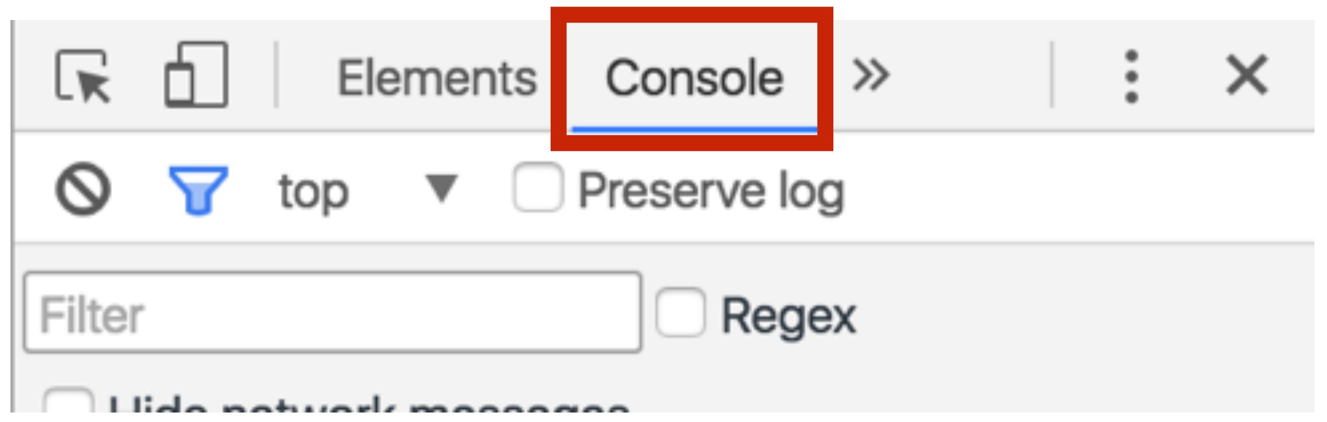
```
> 10 + 5
```

JavaScript lives in your browser

:{) Codaisseur

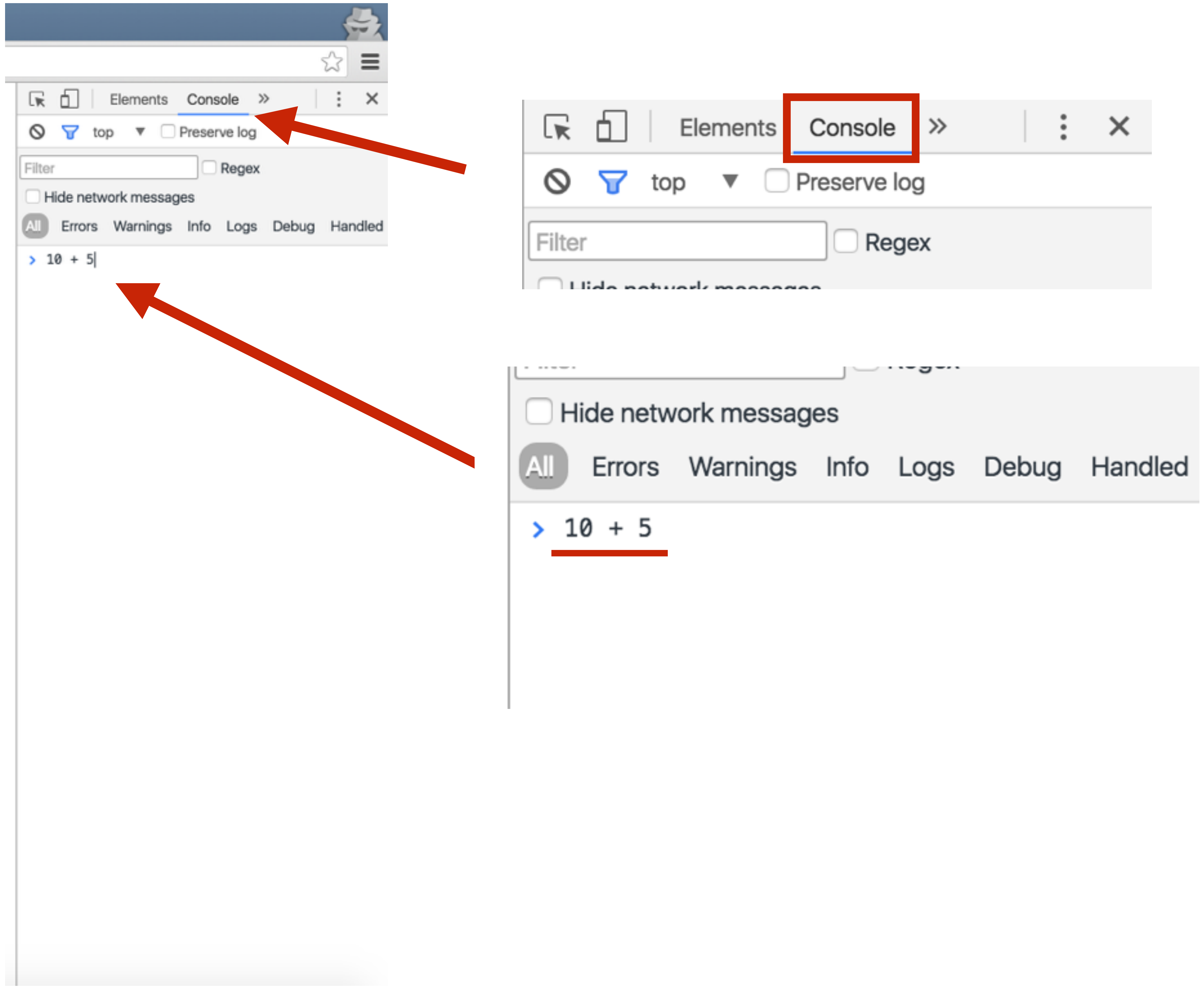


```
> 10 + 5
```



JavaScript lives in your browser

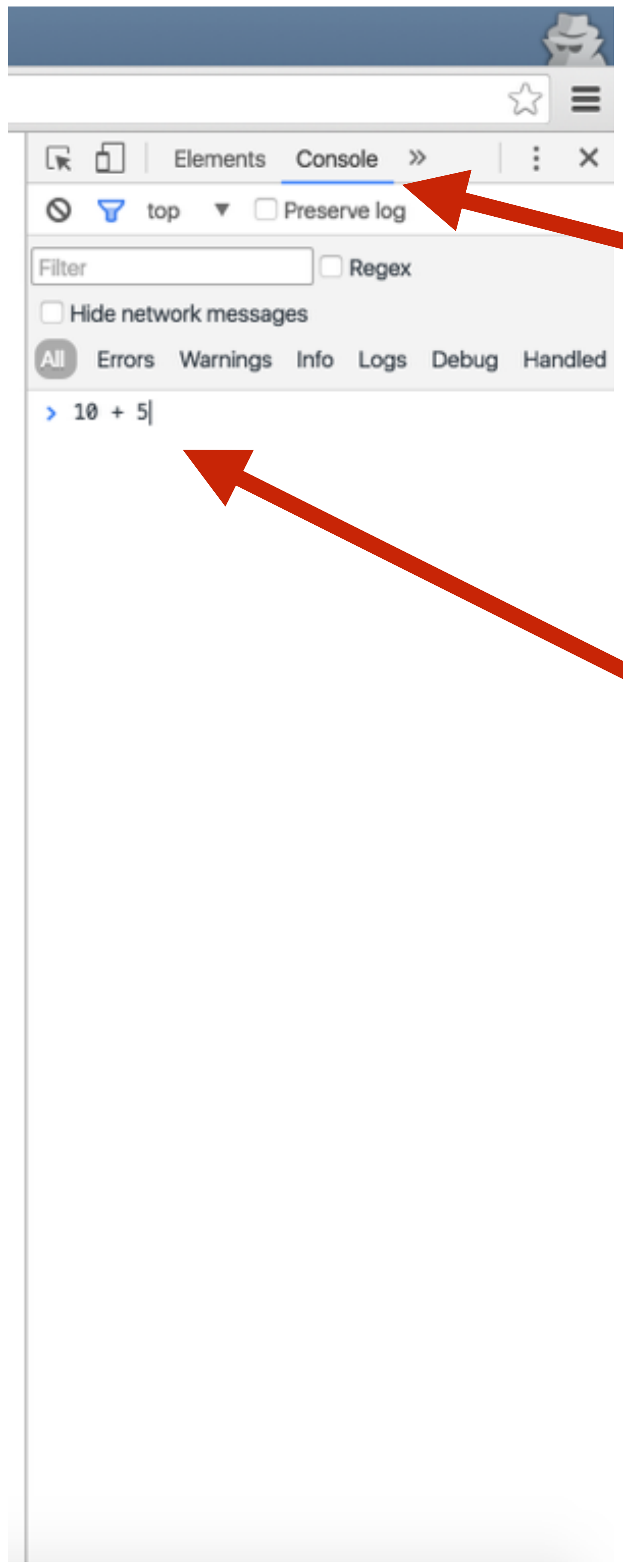
:{) Codaisseur



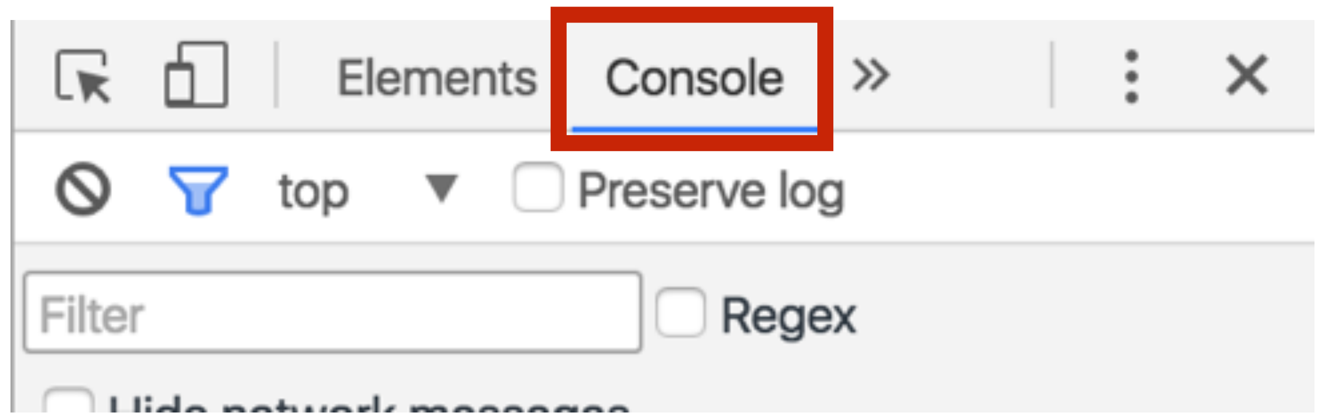
```
> 10 + 5
```

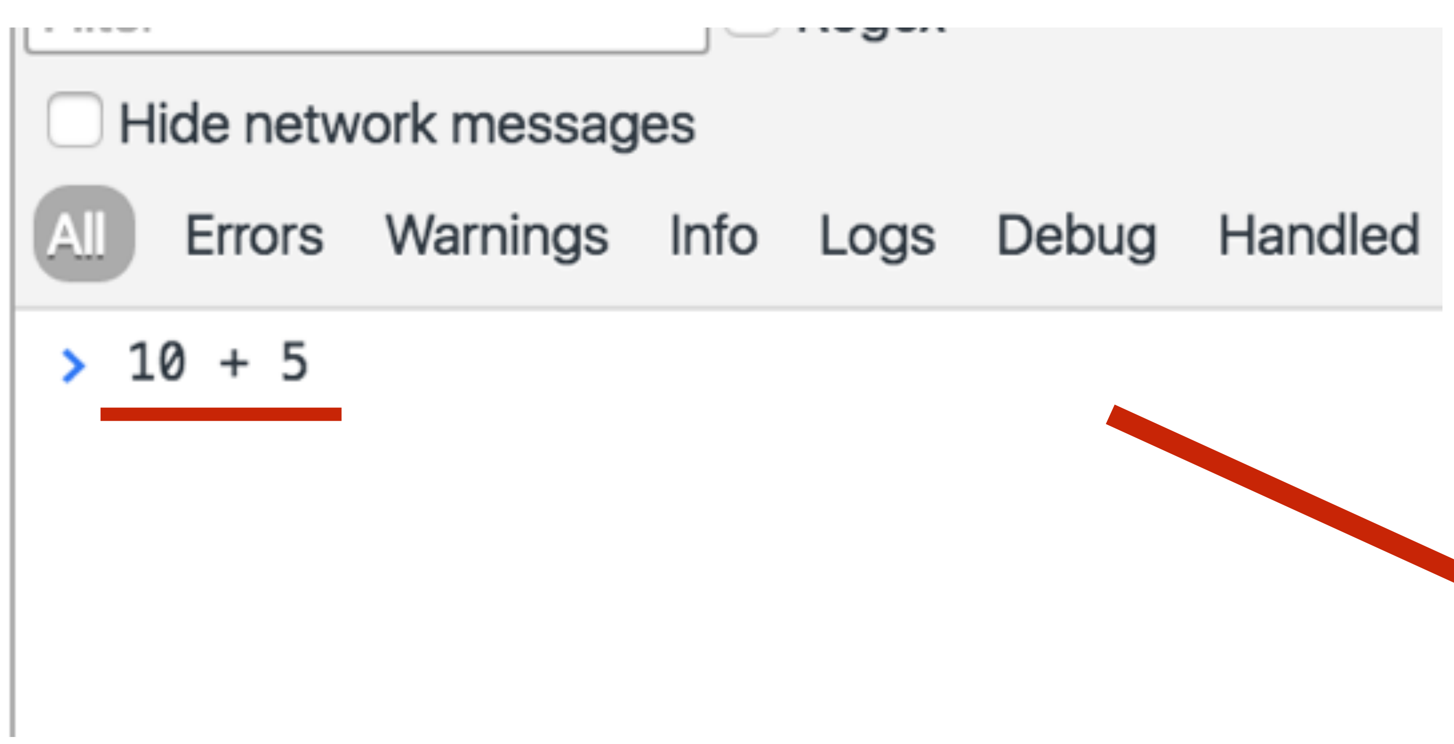
JavaScript lives in your browser

:{) Codaisseur



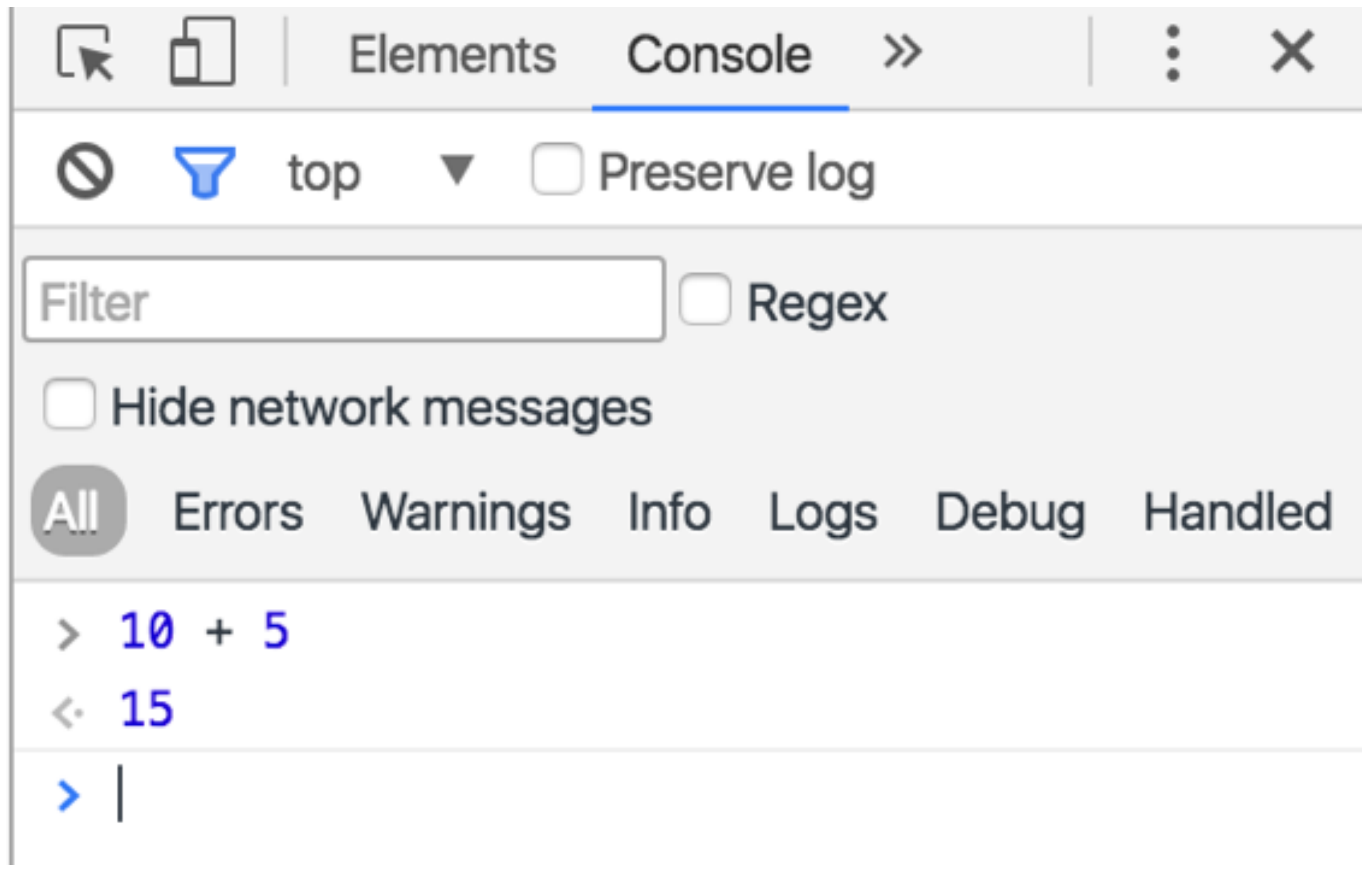
```
> 10 + 5
```





```
> 10 + 5
```

Result



```
> 10 + 5
<· 15
> |
```

✎ **Exercise**

JavaScript Equations

Open the JavaScript console in your browser and use it to do some calculations.

add **+**
subtract **-**
divide **/**
multiply *****

Interactive Javascript

Access content

select elements that have a `warning` class

Interactive Javascript

Codaisseur

Access content

select elements that have a `warning` class

Modify content

add a paragraph of text after the first `<h1>` element

Interactive Javascript

Access content

select elements that have a `warning` class

Modify content

add a paragraph of text after the first `<h1>` element

Program Rules or Instructions

write a script that writes some content depending on the day time

Interactive Javascript

:{) Codaisseur

Access content
select elements that have a `warning` class

Modify content
add a paragraph of text after the first `<h1>` element

Program Rules or Instructions
write a script that writes some content depending on the day time

React to events
specify that a script should be run when a button is clicked

Objects, properties & methods

Objects, properties & methods

In computer programming,
each thing in the world
can be represented as an object.

Objects, properties & methods

In computer programming,
each thing in the world
can be represented as an object.

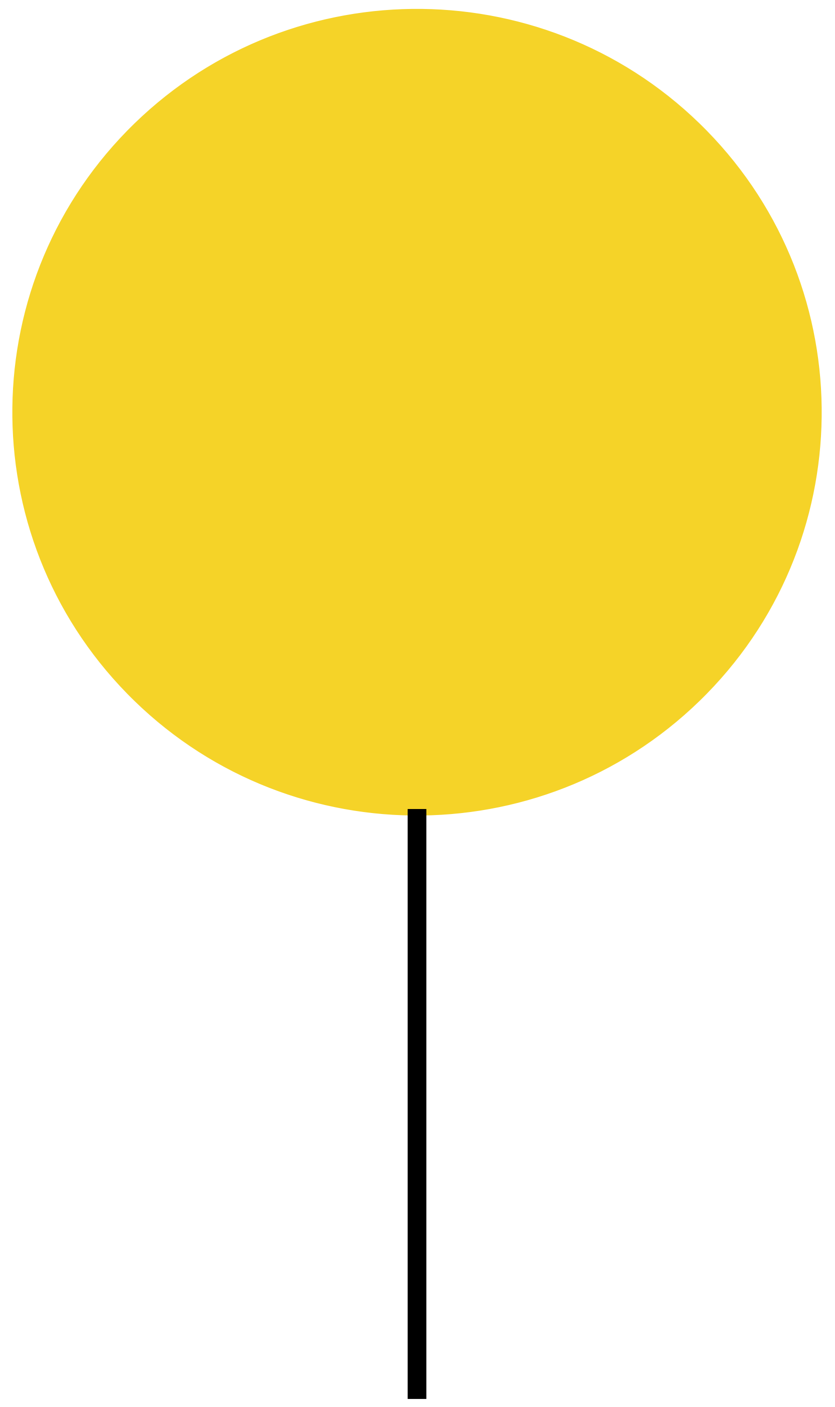
Each object can have
its own **properties** and **methods**.

Objects, properties & methods

:{) Codaisseur

In computer programming,
each thing in the world
can be represented as an object.

Each object can have
its own **properties** and **methods**.

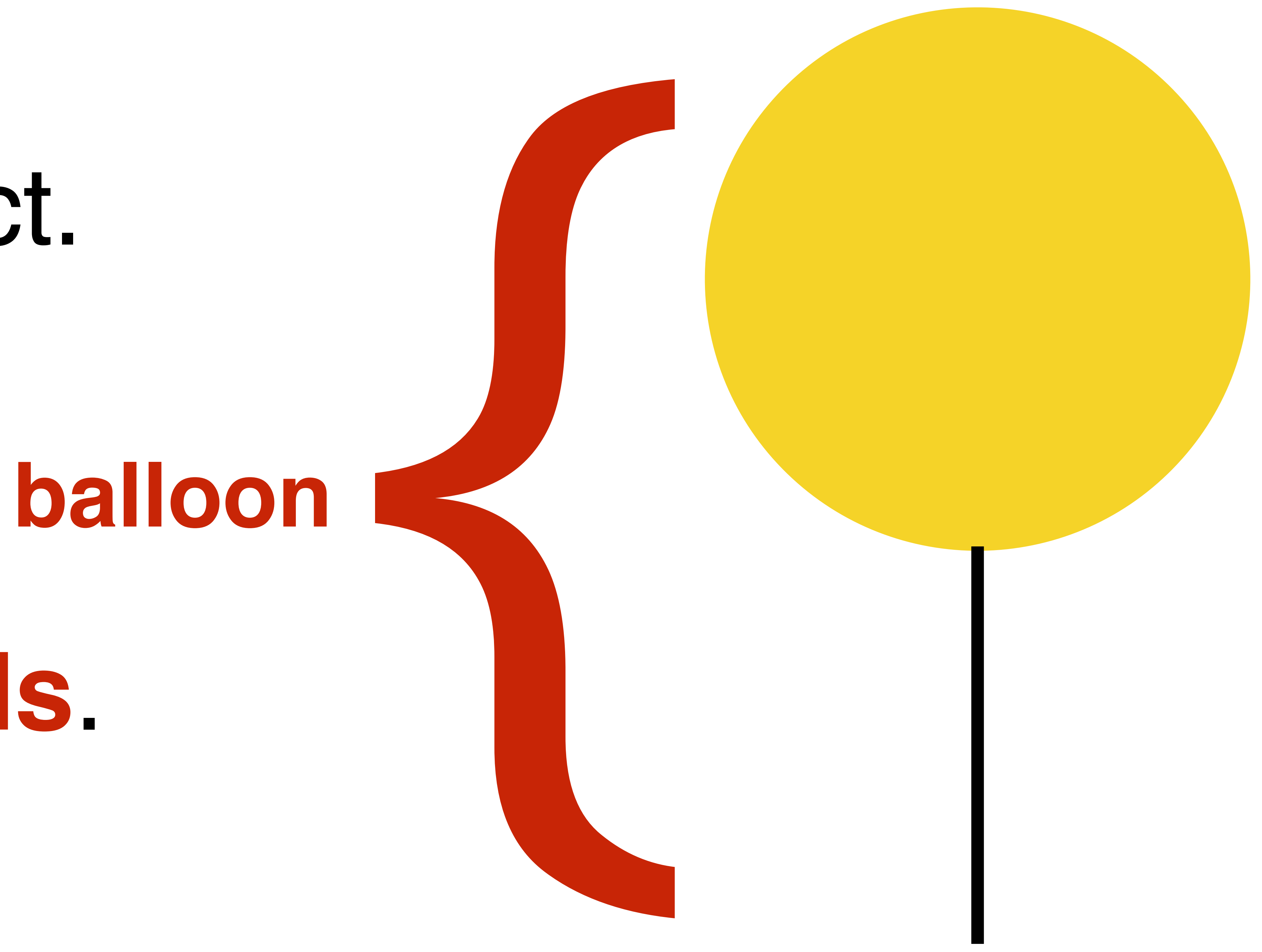


Objects, properties & methods

:{) Codaisseur

In computer programming,
each thing in the world
can be represented as an object.

Each object can have
its own **properties** and **methods**.

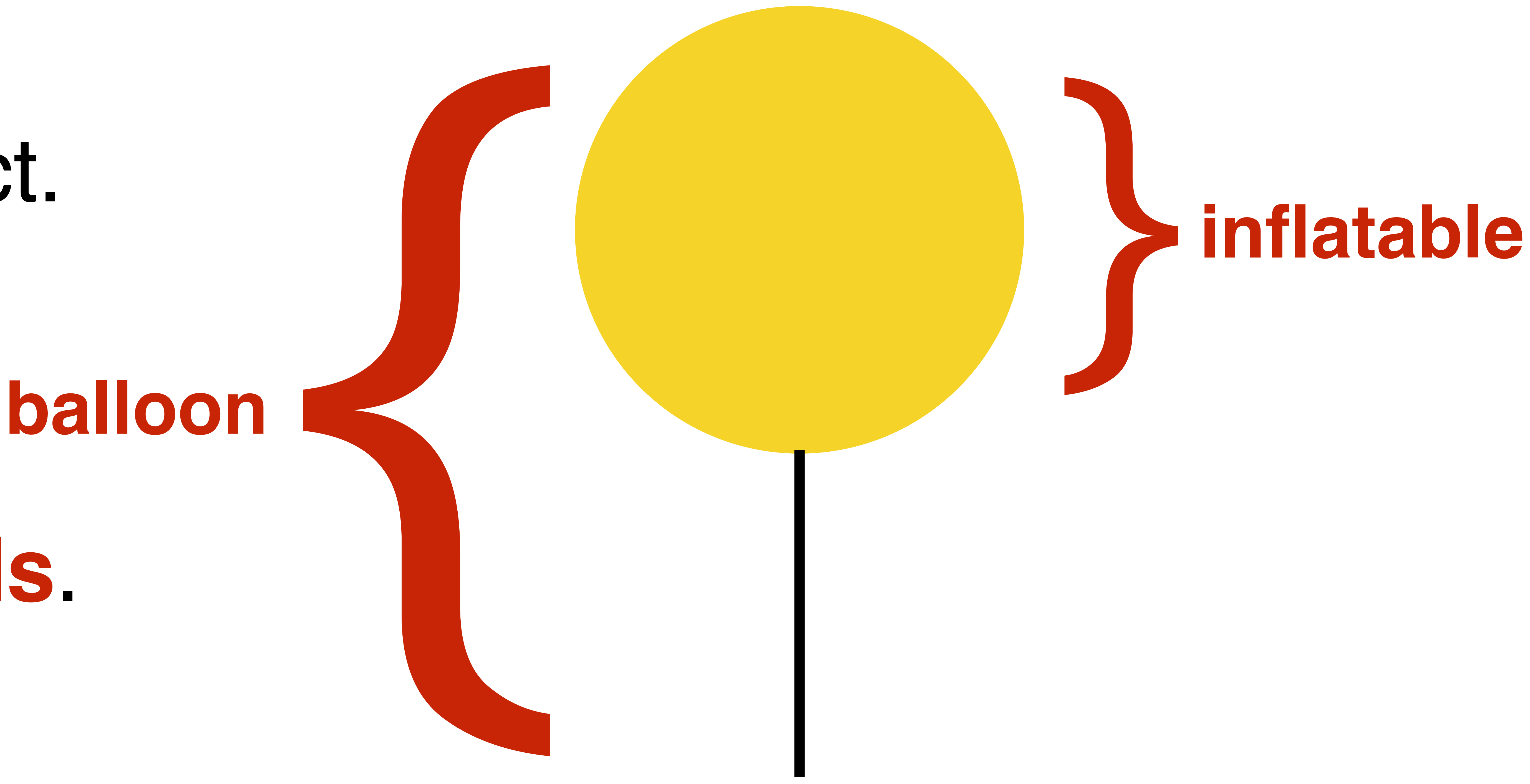


Objects, properties & methods

:{) Codaisseur

In computer programming,
each thing in the world
can be represented as an object.

Each object can have
its own **properties** and **methods**.

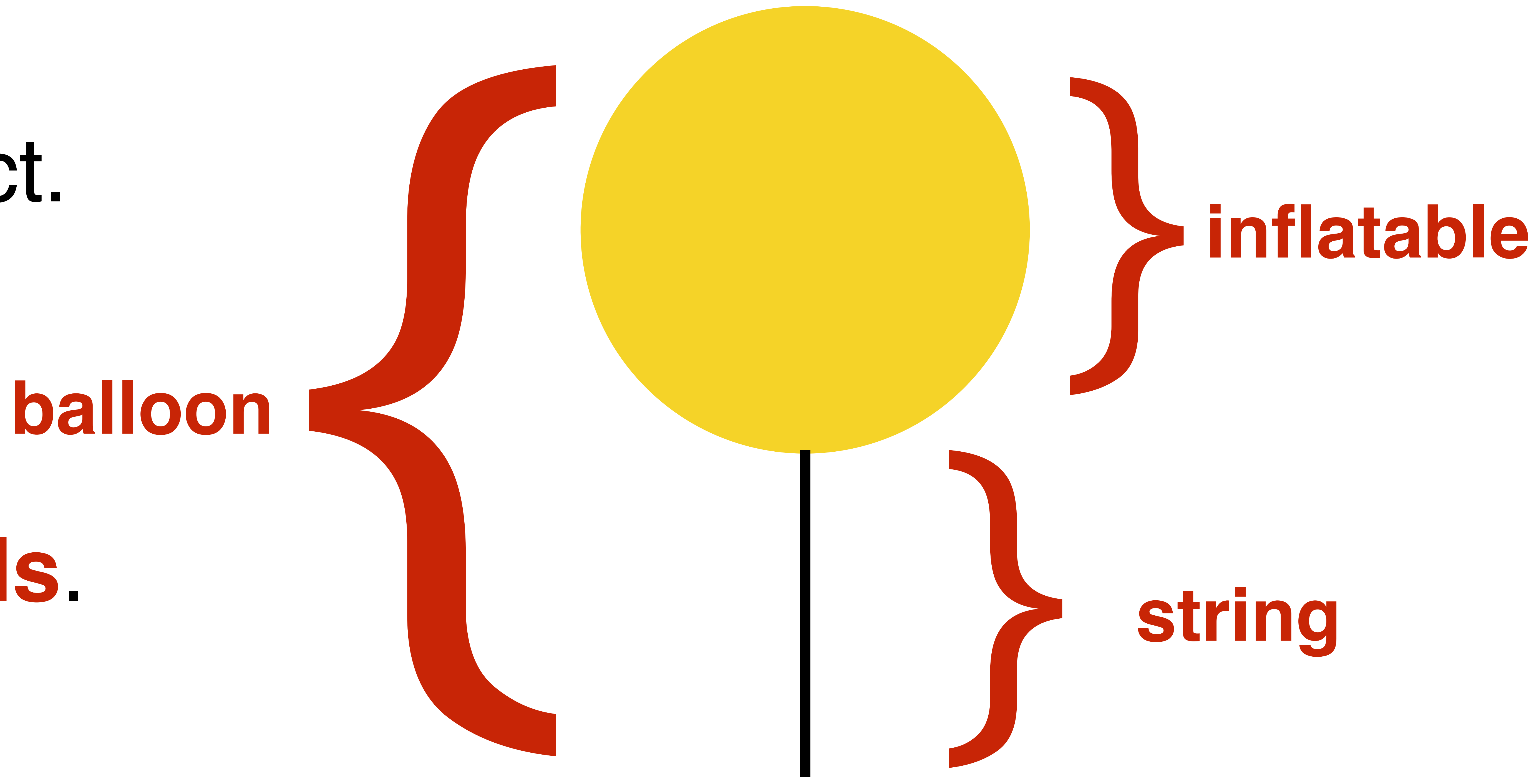


Objects, properties & methods

:{) Codaisseur

In computer programming,
each thing in the world
can be represented as an object.

Each object can have
its own **properties** and **methods**.



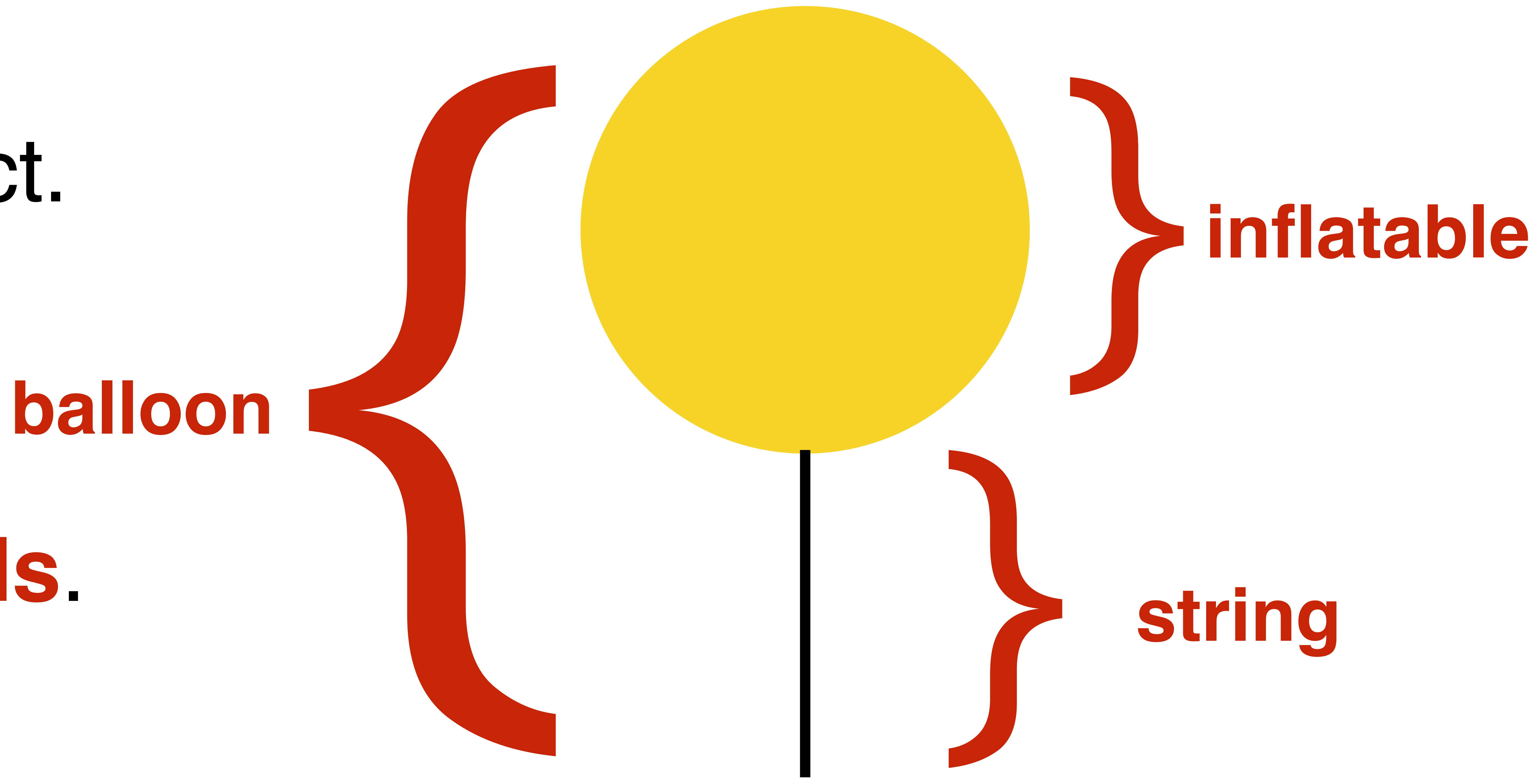
Objects, properties & methods

:{) Codaisseur

In computer programming,
each thing in the world
can be represented as an object.

Each object can have
its own **properties** and **methods**.

A **method** is an action that can
be performed on an object.



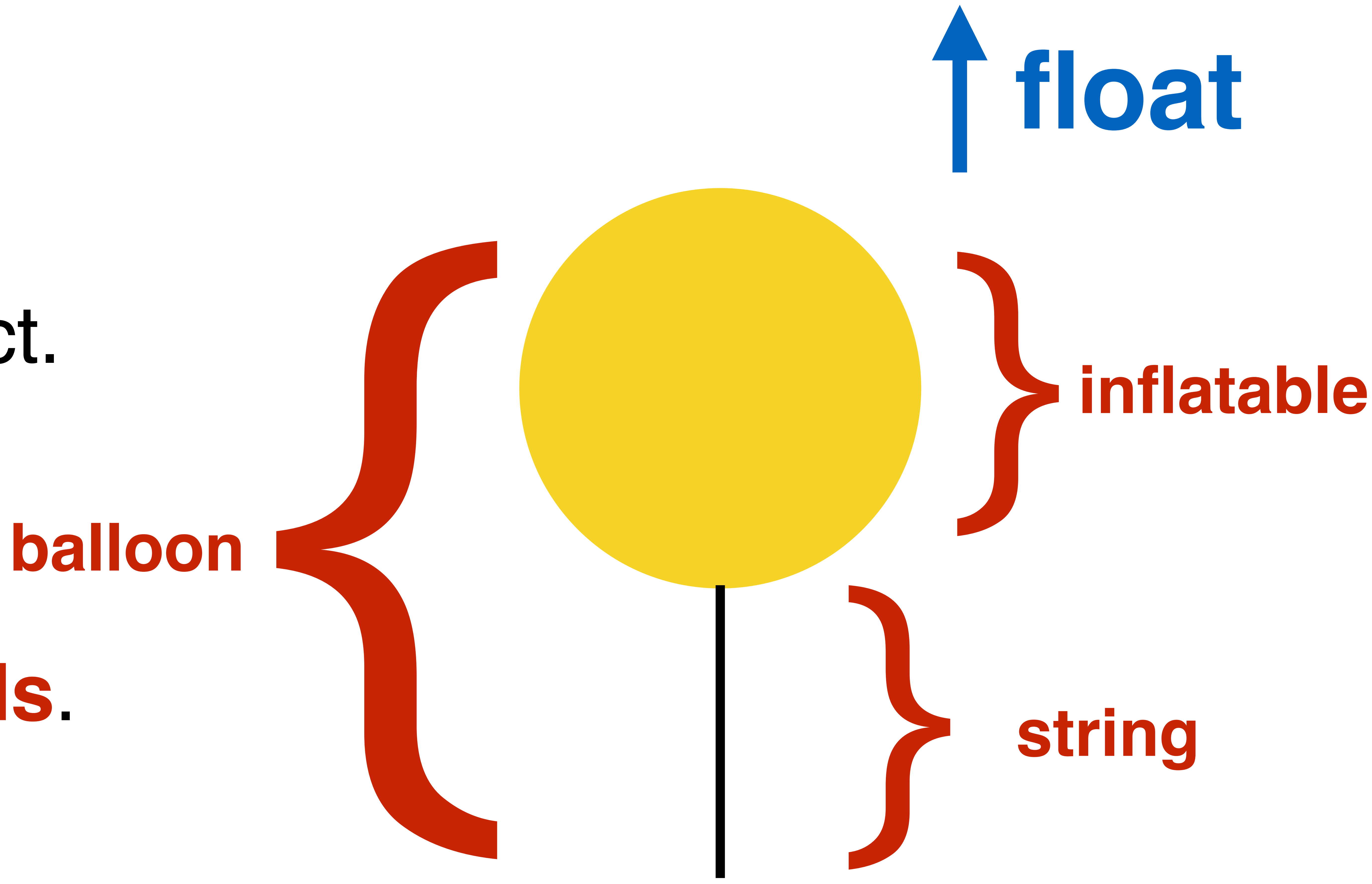
Objects, properties & methods

:{) Codaisseur

In computer programming,
each thing in the world
can be represented as an object.

Each object can have
its own **properties** and **methods**.

A **method** is an action that can
be performed on an object.



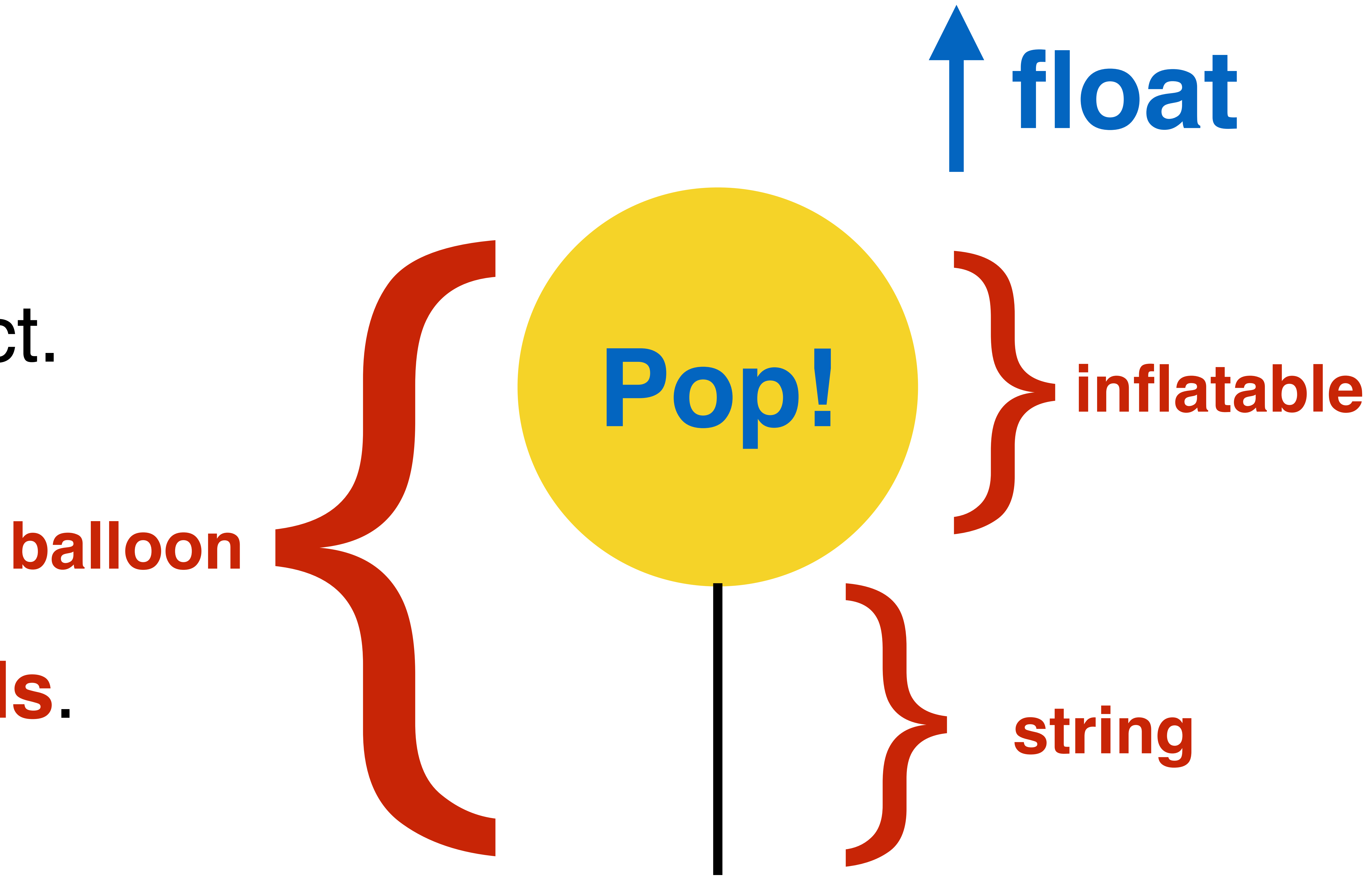
Objects, properties & methods

:{) Codaisseur

In computer programming,
each thing in the world
can be represented as an object.

Each object can have
its own **properties** and **methods**.

A **method** is an action that can
be performed on an object.



Objects, properties & methods

:{) Codaisseur

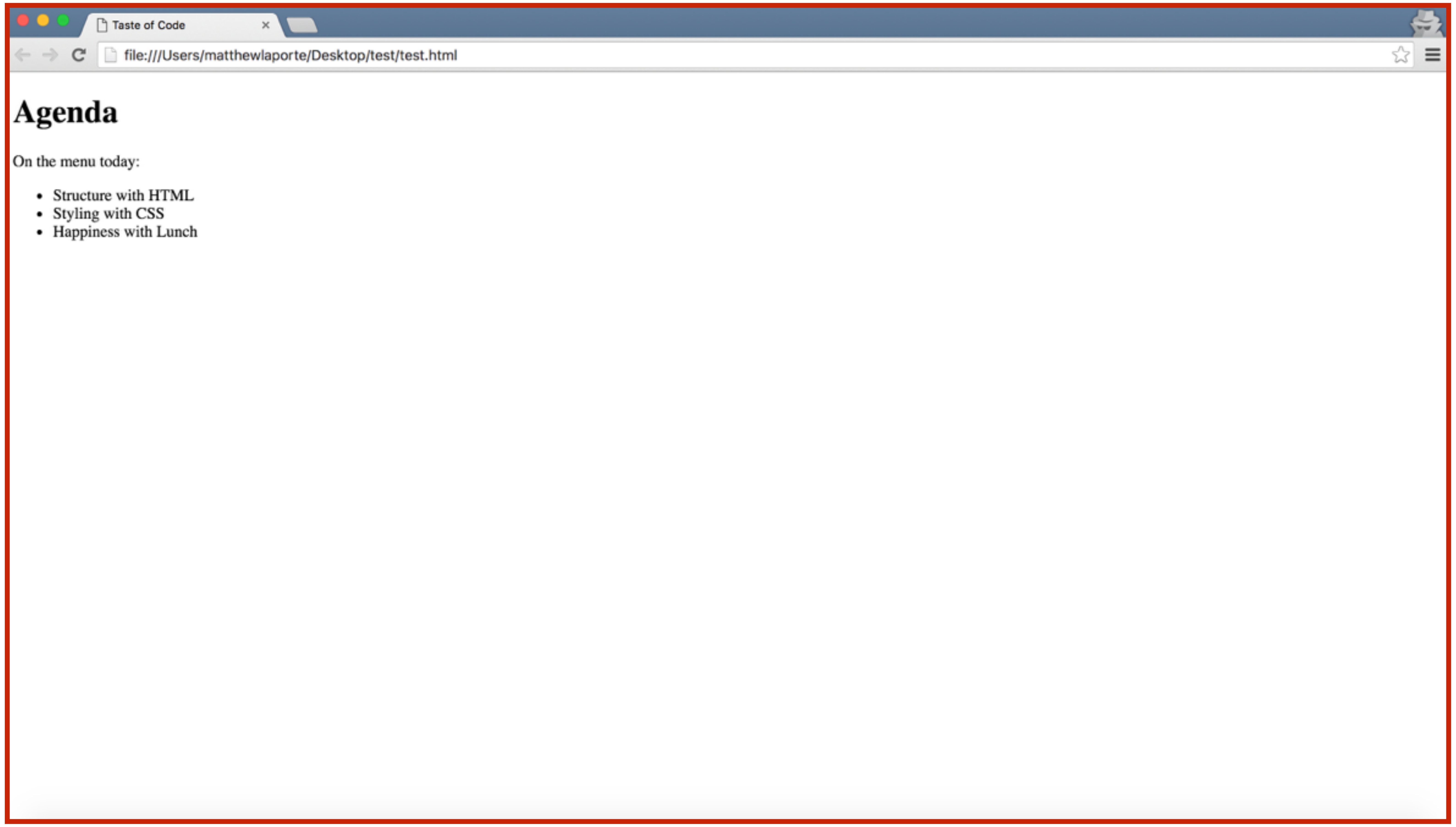
Objects, properties & methods

The browser represents each window or tab using a **window** object.

Objects, properties & methods

:{) Codaisseur

The browser represents each window or tab using a **window** object.


Taste of Code

file:///Users/matthewlaporte/Desktop/test/test.html

Agenda

On the menu today:

- Structure with HTML
- Styling with CSS
- Happiness with Lunch

✎ Exercise

Interact with the window

Carry out the actions below in your javascript console.

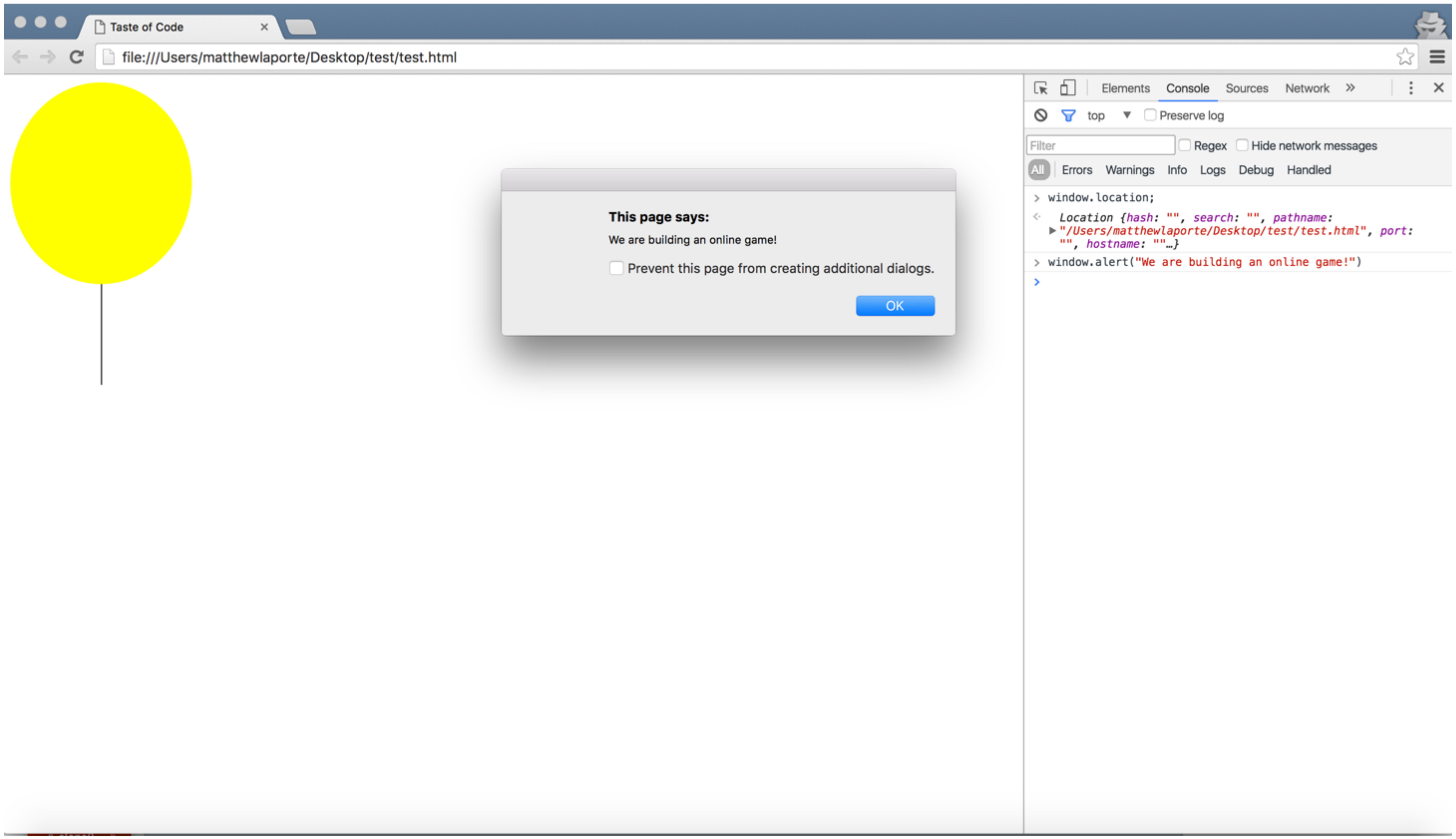
javascript console

```
window.location;
window.alert(“We are building an online game!”);
```

Objects, properties & methods

Objects, properties & methods

:{} Codaisseur



This page says:

We are building an online game!

Prevent this page from creating additional dialogs.

```
> window.location;
<  Location {hash: "", search: "", pathname:
  ▶"/Users/matthewlaporte/Desktop/test/test.html", port:
   "", hostname: ""…}
> window.alert("We are building an online game!")
>
```

Objects, properties & methods

:{) Codaisseur

Objects, properties & methods

:{} Codaisseur

```
window.alert();
```

Objects, properties & methods

$$\underbrace{\texttt{window}}_{\text{object}}\texttt{.alert();}$$

Objects, properties & methods

Codaisseur

$$\underbrace{\texttt{window}}_{\text{object}}\underbrace{\texttt{.alert();}}_{\text{method}}$$

Objects, properties & methods

:{) Codaisseur

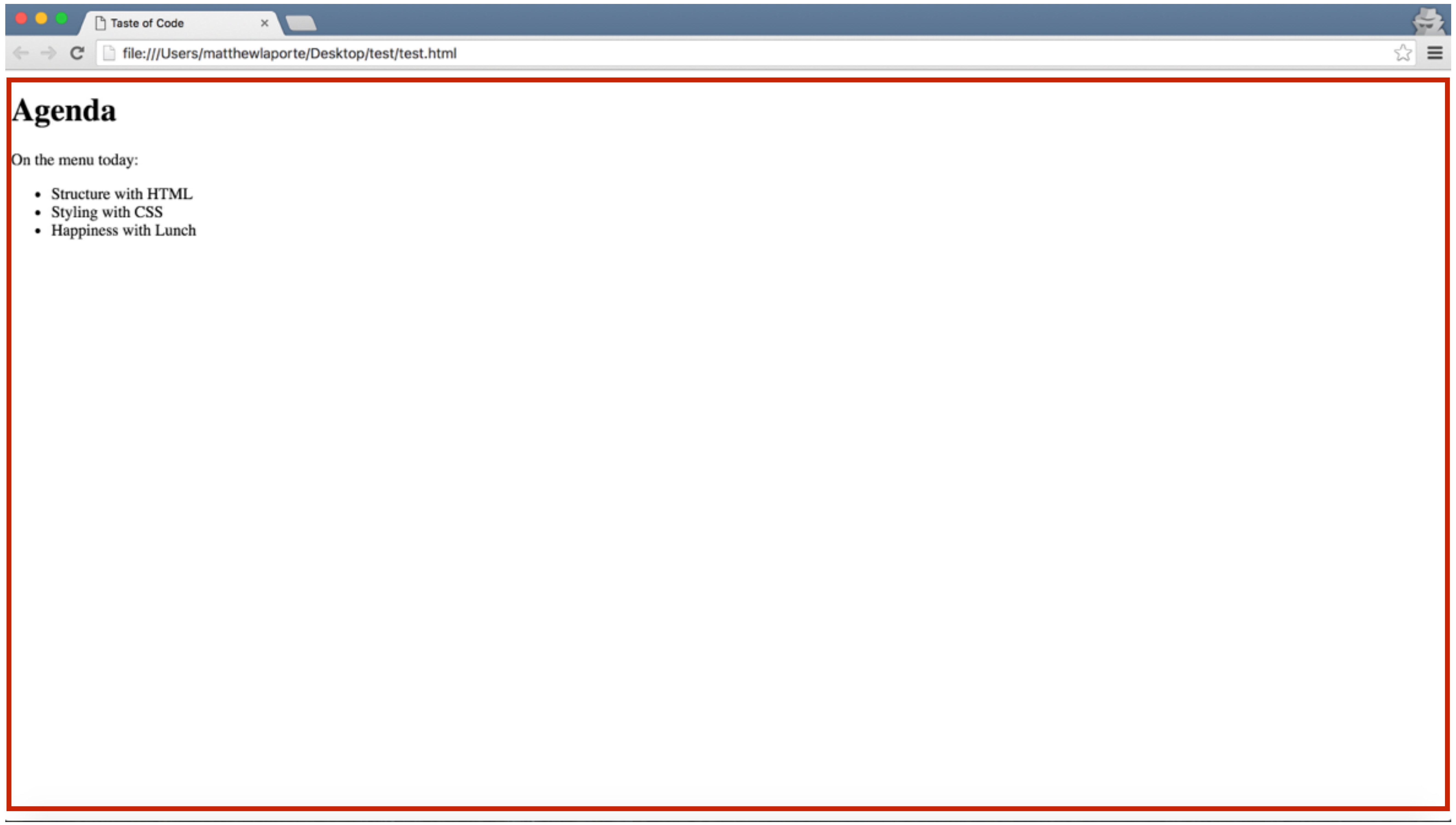
Objects, properties & methods

The current web page loaded into each window is modelled using the **document** object.

Objects, properties & methods

:{) Codaisseur

The current web page loaded into each window is modelled using the **document** object.



Agenda

On the menu today:

- Structure with HTML
- Styling with CSS
- Happiness with Lunch

✎ Exercise

Interact with the document

Carry out the actions below in your javascript console.

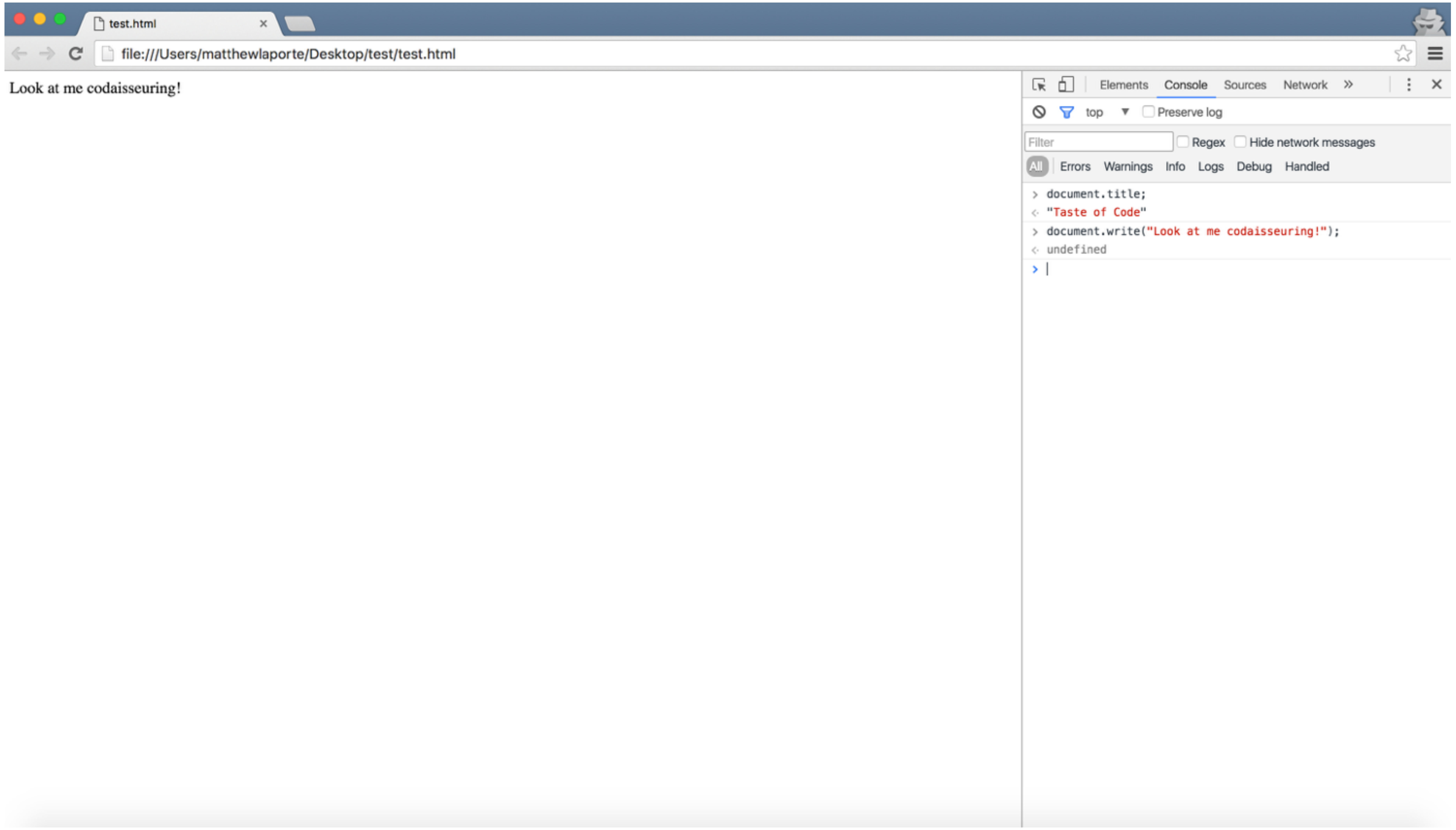
javascript console

```
document.title;
document.write("Look at me Codaisseuring!");
```

Objects, properties & methods

Objects, properties & methods

:{) Codaisseur



Look at me codaisseuring!

```
> document.title;
< "Taste of Code"
> document.write("Look at me codaisseuring!");
< undefined
>
```

What is a Variable?

What is a Variable?

Variables can be thought of as **named containers**.

What is a Variable?

Variables can be thought of as **named containers**.

You can place data into these containers and refer to that data simply by **calling the container**.

What is a Variable?

Variables can be thought of as **named containers**.

You can place data into these containers and refer to that data simply by **calling the container**.

Variables are useful in utilising recurring data.

What is a variable?

:{) Codaisseur

javascript console

```
// Store a string
var text = “Look at me Codaisseuring!”;
document.write(text);
window.alert(text);
```

```
// Store an equation
var num1 = 1;
var num2 = 2;
num1 + num2
```

```
// Store an equation
> var sum = 10 + 5;
> document.write(sum);
```

✎ Exercise

Store some numbers in variables and use them to create some equations.

jQuery

Easier JavaScript

Achieve common JavaScript tasks **quickly and consistently**, across all major browsers.

What is jQuery?

What is jQuery?

Codaisseur

jQuery is a JavaScript library

What is jQuery?

jQuery is a JavaScript library

A JavaScript library is pre-written JavaScript code that makes things easier

What is jQuery?

:{) Codaisseur

jQuery is a JavaScript library

A JavaScript library is pre-written JavaScript code that makes things easier

jQuery makes it easier for you to utilise JavaScript!

Include jQuery in your page

Include jQuery in your page

:{) Codaisseur

```
<!DOCTYPE html>
<html>
  <head>
      …
  </head>
  <body>
      …
  </body>
  <script src=“https://ajax.googleapis.com/ajax/libs/jquery/3.0.0/jquery.min.js”></script>
</html>
```

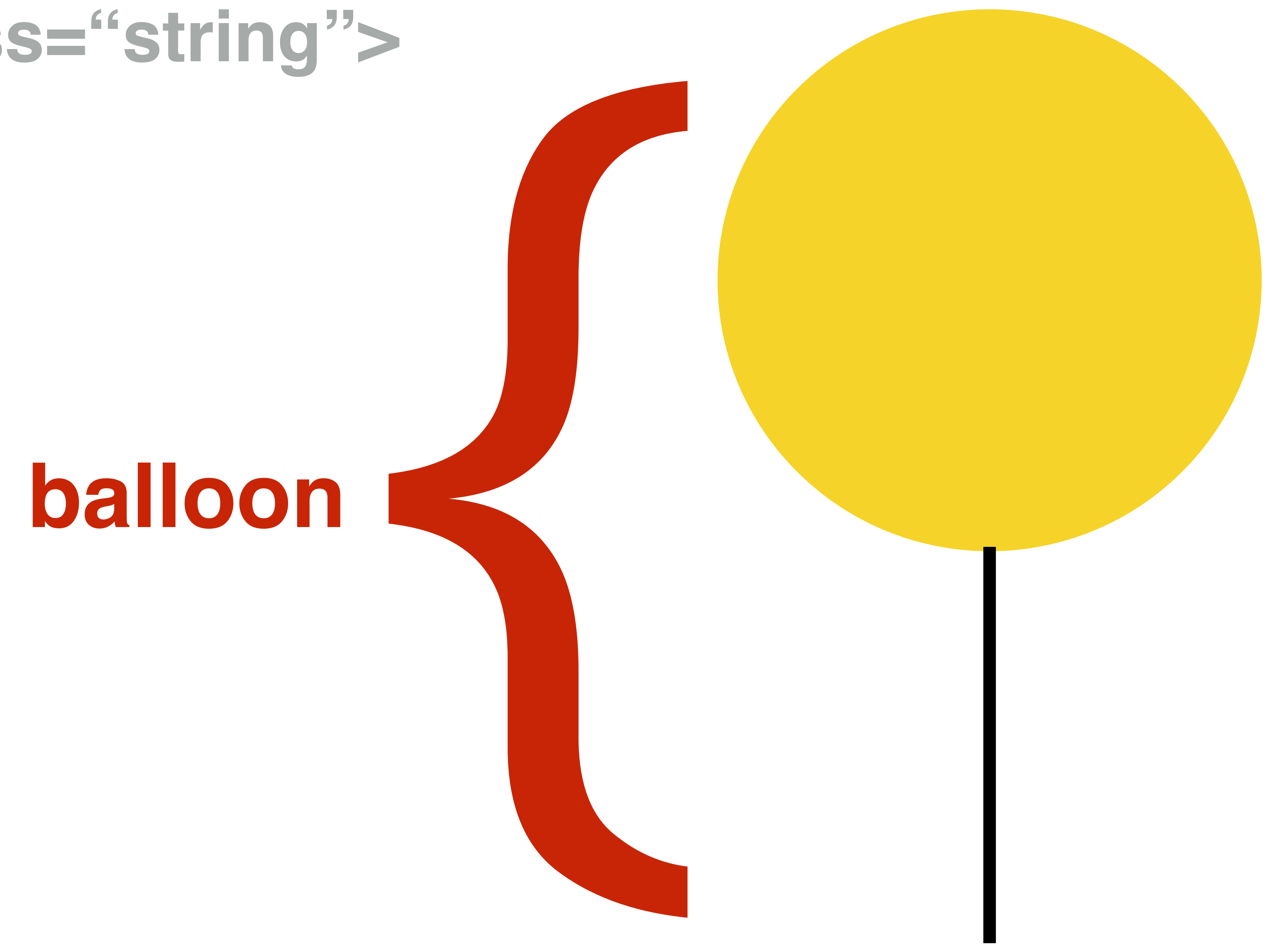
index.html

Cloning

Cloning

:{) Codaisseur

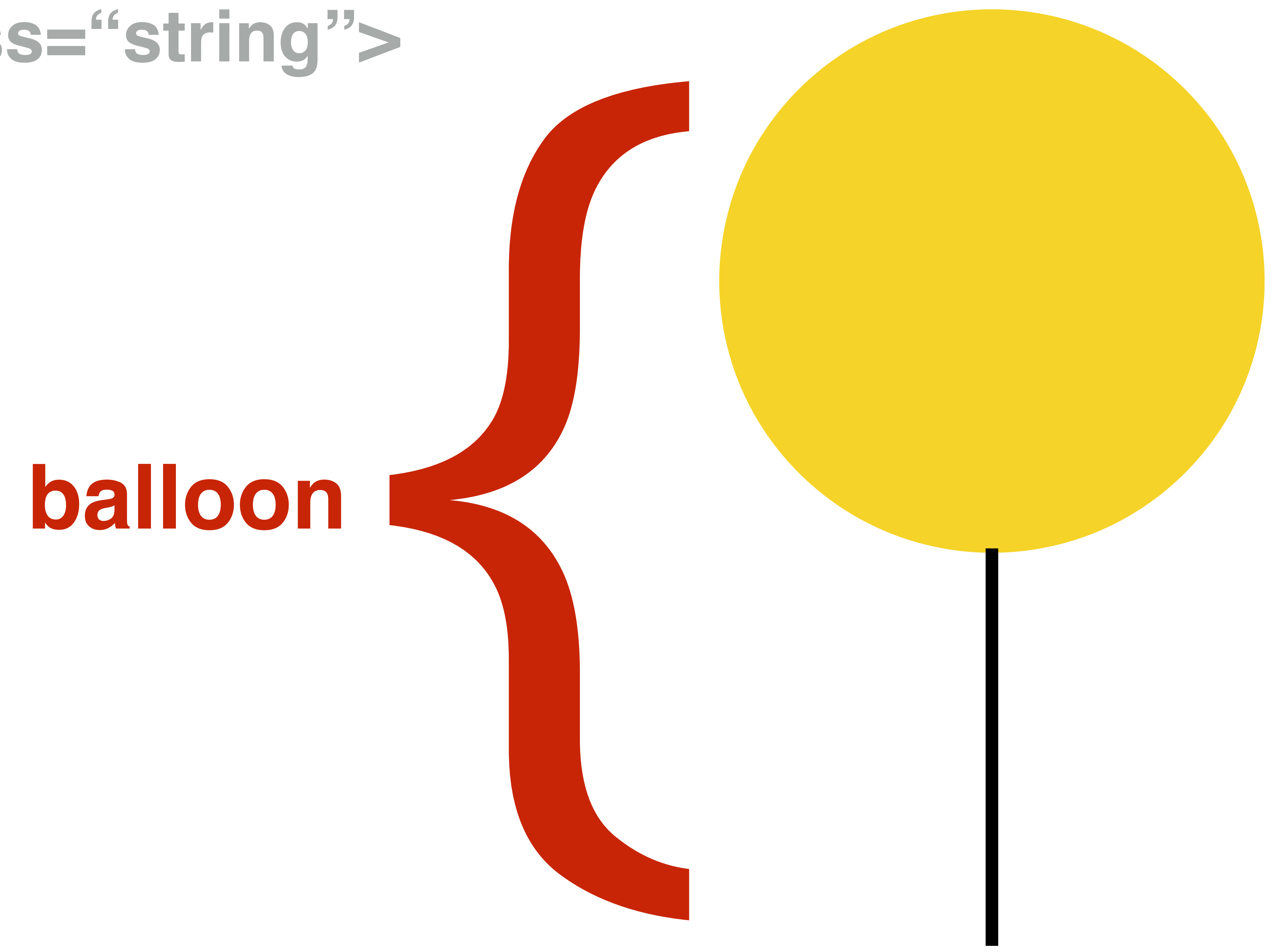
```
<div class="balloon">
  <div class="inflatable">
  </div>
  <div class="string">
  </div>
</div>
```



Cloning

:{) Codaisseur

```
<div class=“balloon”>
  <div class=“inflatable”>
  </div>
  <div class=“string”>
  </div>
</div>
```



Syntax

```
var balloon = $( ".balloon" );
var balloonCopy = balloon.clone();
balloonCopy.appendTo( "body" );
```

Cloning

Cloning

:{) Codaisseur

```
<!DOCTYPE html>
<html>
  <head>
      ...
  </head>
  <body>
      ...
  </body>
 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.0.0/jquery.min.js"></script>
 <script>
  var balloon = $( ".balloon" );
  var balloonCopy = balloon.clone();
  balloonCopy.appendTo( "body" );
 </script>
</html>
```

index.html

✎ Exercise

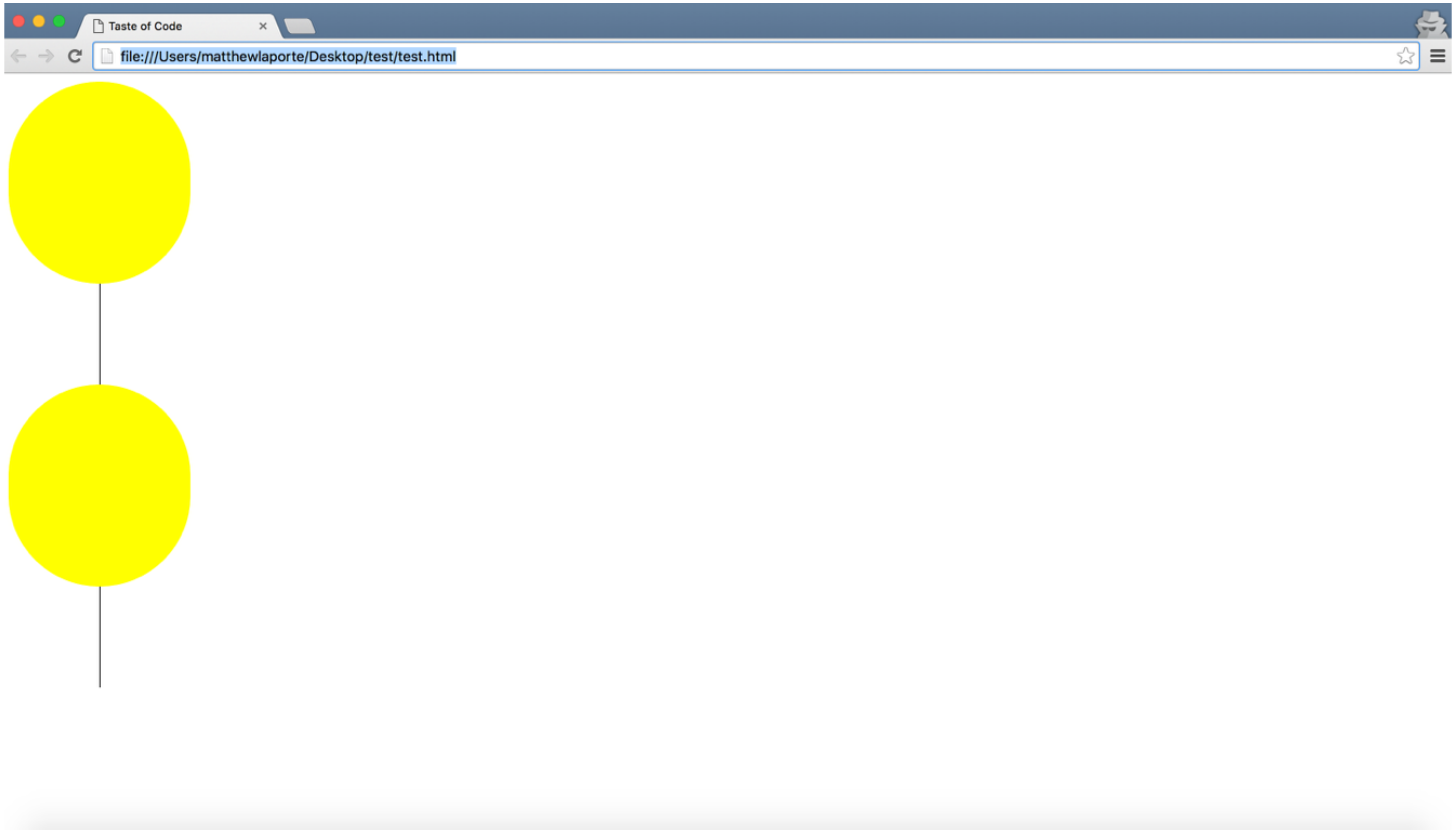
Using the syntax below, clone your balloon

```
var balloon = $( ".balloon" );
var balloonCopy = balloon.clone();
balloonCopy.appendTo( "body" );
```

index.html

Cloning

Cloning


Taste of Code
file:///Users/matthewlaporte/Desktop/test/test.html

Loops

Loops

jQuery allows you to **loop**
through the properties of an object,
using the **.each()** method.

Loops

jQuery allows you to **loop**
through the properties of an object,
using the **.each()** method.

Often you will want to perform
a **series of actions** on each of the elements.

✎ Exercise

Using the syntax below, make use of the loop function to create 10 balloons

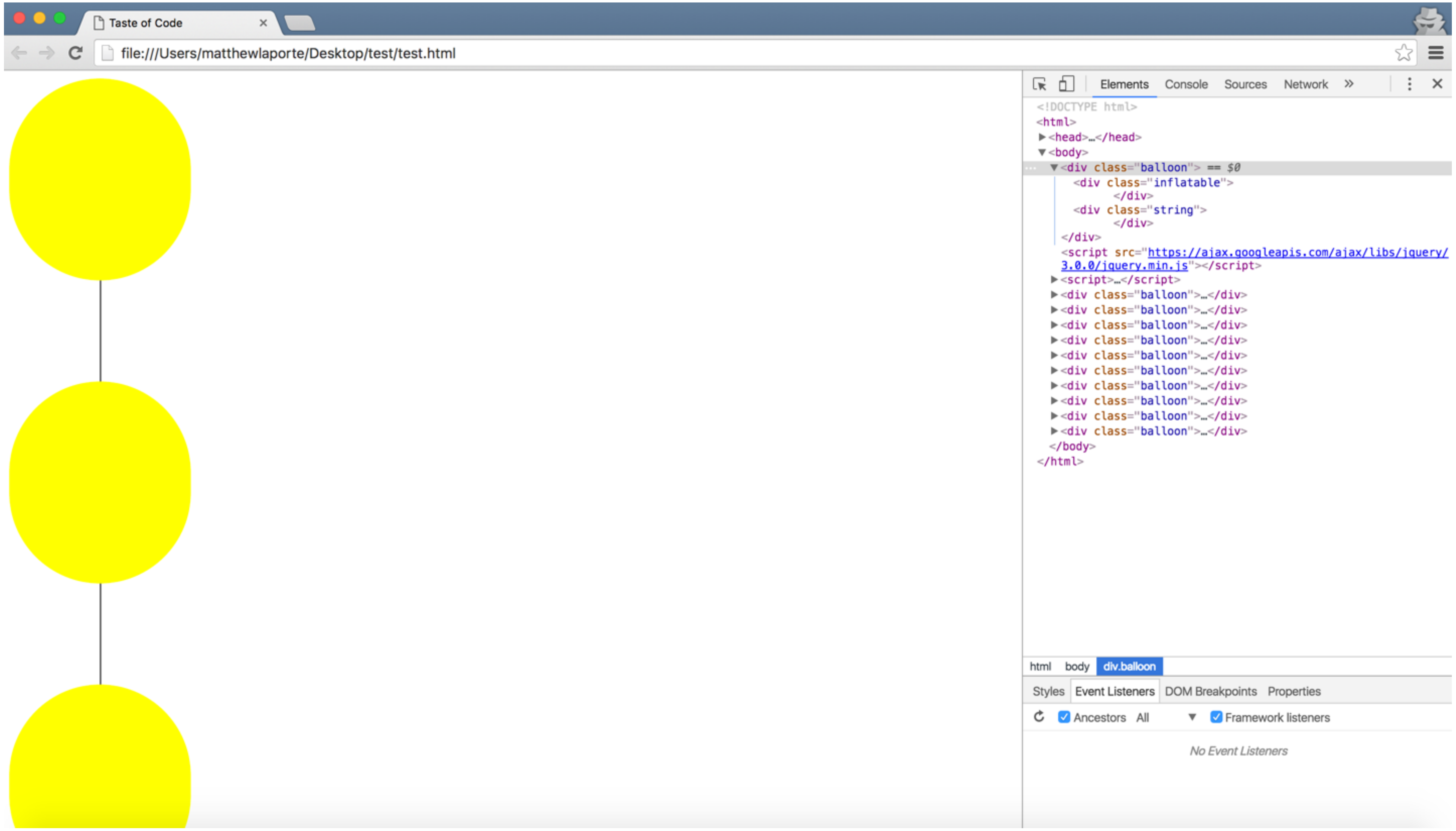
```
var balloon = $(".balloon");
for(var i=0; i<10; i++){
    var balloonCopy = balloon.clone();
    balloonCopy.appendTo(“body”);
}
```

index.html

Loops

Loops

:{) Codaisseur



Taste of Code
file:///Users/matthewlaporte/Desktop/test/test.html
Elements
Console
Sources
Network

```
<!DOCTYPE html>
<html>
▶<head>…</head>
▼<body>
 ▼<div class="balloon"> == $0
    <div class="inflatable">
        </div>
    <div class="string">
        </div>
  </div>
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.0.0/jquery.min.js"></script>
 ▶<script>…</script>
 ▶<div class="balloon">…</div>
 ▶<div class="balloon">…</div>
 ▶<div class="balloon">…</div>
 ▶<div class="balloon">…</div>
 ▶<div class="balloon">…</div>
 ▶<div class="balloon">…</div>
 ▶<div class="balloon">…</div>
 ▶<div class="balloon">…</div>
 ▶<div class="balloon">…</div>
 ▶<div class="balloon">…</div>
 </body>
</html>
```

html body div.balloon
Styles Event Listeners DOM Breakpoints Properties
Ancestors All Framework listeners
No Event Listeners

Events

Events

Events are things that happen to HTML elements.

Events

Codaisseur

Events are things that happen to HTML elements.

Javascript lets you execute code when events are detected

Events

Events are things that happen to HTML elements.

Javascript lets you execute code when events are detected

We can use a **click** event to assist in “popping” our balloons.

Events

```
balloonCopy.click(function(){
    $( this ).remove();
});
```

Events

```
balloonCopy.click(function(){
     $( this ).remove();
});
```

Events

:{) Codaisseur

```
balloonCopy.click(function(){
     $( this ).remove();
});
```

Callback

Events

```
balloonCopy.click(function(){
     $( this ).remove();
});
```

Callback

method as an argument
of a method

Events

:{) Codaisseur

```
balloonCopy.click(function(){
      $( this ).remove();
});
```

Callback

method as an argument
of a method

Events

:{) Codaisseur

```
balloonCopy.click(function(){
      $( this ).remove();
});
```

this?

Callback

method as an argument of a method

Events

:{) Codaisseur

```
balloonCopy.click(function(){
     $( this ).remove();
});
```

this?
it's really just a shortcut reference
to the object that invoked the method

Callback

method as an argument
of a method

Events

:{) Codaisseur

```
balloonCopy.click(function(){
    $( this ).remove();
});
```

this?

it’s really just a shortcut reference to the object that invoked the method

Callback

method as an argument of a method

✎ Exercise

Using the syntax below, incorporate the click event to start popping balloons.

```
  balloonCopy.appendTo(“body”);
  balloonCopy.click(function() {
    $(this).remove();
  });
};
balloon.remove();
```

index.html

Positioning

Positioning

We can adjust the positioning of the balloons using the **absolute** css value.

Positioning

:{) Codaisseur

We can adjust the positioning of the balloons using the **absolute** css value.

Absolute is a type of positioning that allows you to literally place any element exactly where you want it.

✎ Exercise

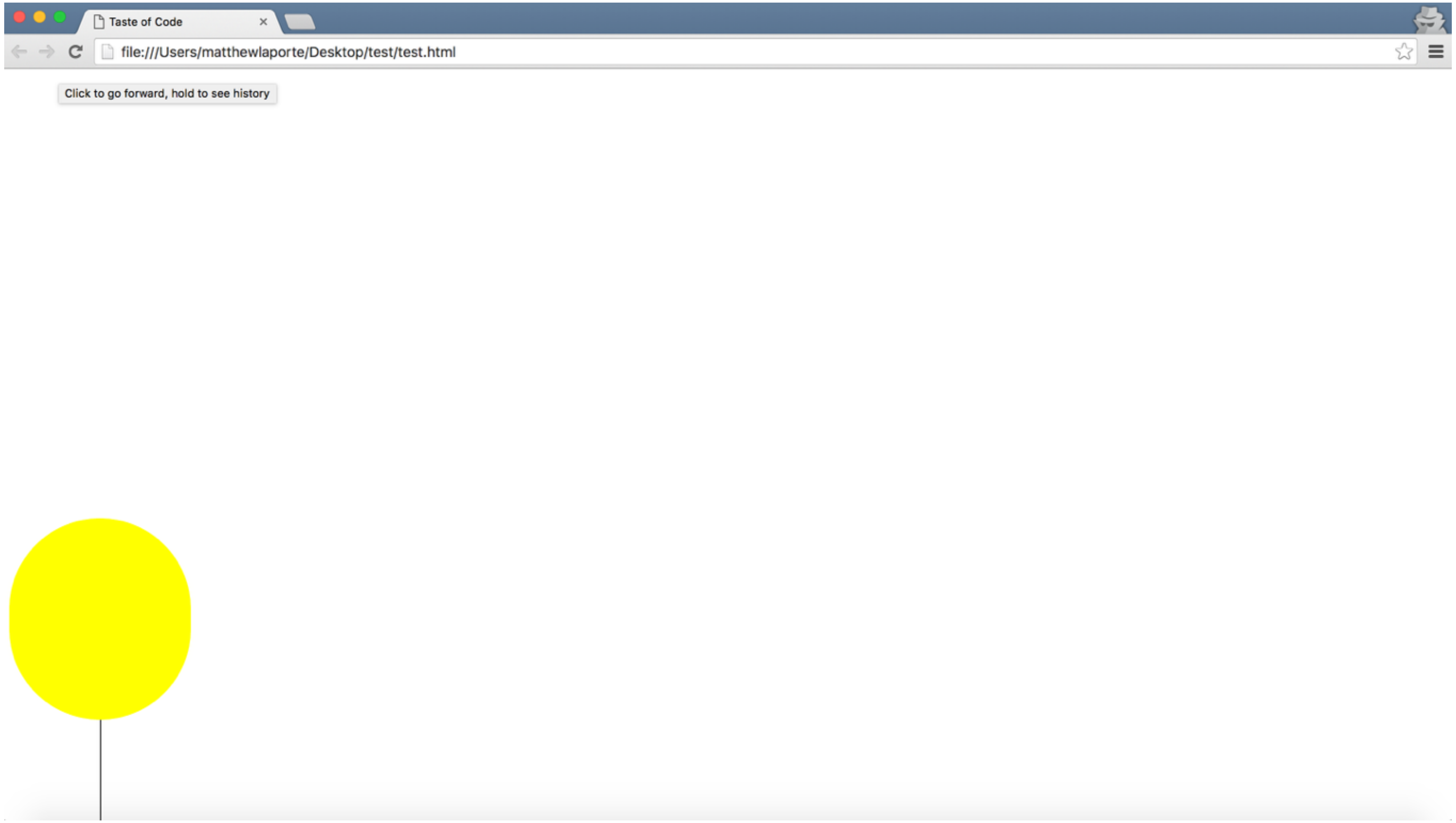
Using the syntax below, add position attributes to the CSS balloon class so they change position.

```
.balloon {
    position: absolute;
    bottom: 0;
}
```

index.html

Positioning

:{) Codaisseur


Taste of Code
file:///Users/matthewlaporte/Desktop/test/test.html
Click to go forward, hold to see history

✎ Exercise

Using the syntax below, add css attribute so the balloons they appear in a straight line.

```
var balloon = $(".balloon");
for(var i=0; i<10; i++){
    var balloonCopy = balloon.clone();
    balloonCopy.css({
      left: i * 200
    });
    balloonCopy.appendTo("body");
```

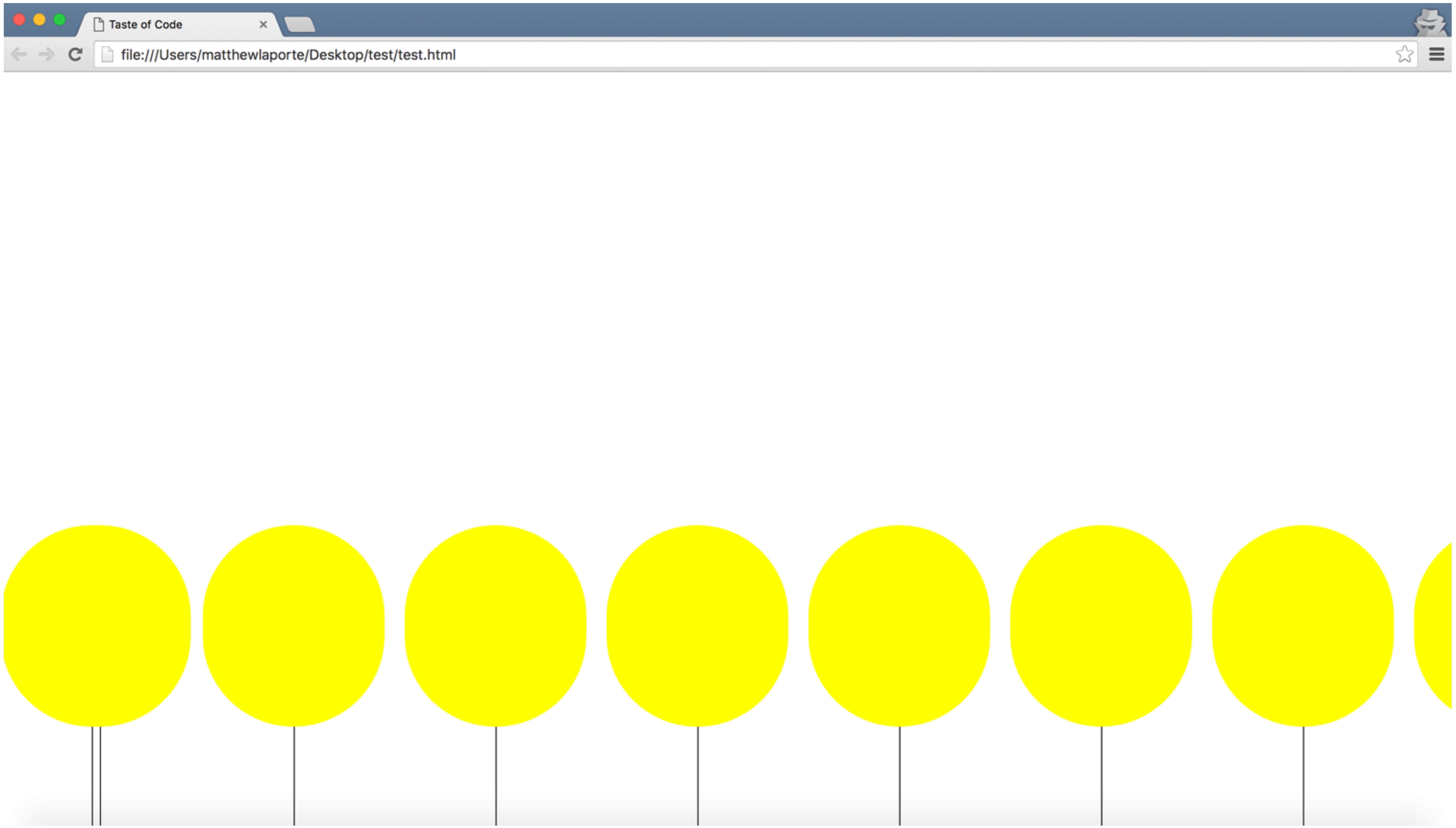
index.html

Positioning

:{) Codaisseur

Positioning

:{) Codaisseur



Animate

Animate

We can make our balloons rise by using the **animate** method.

Animate

We can make our balloons rise by using the **animate** method.

Animate changes an element from one state to another gradually, by adjusting the CSS attributes.

✎ Exercise

Using the syntax below, make your balloons rise.

```
balloonCopy.click(function() {
   $(this).remove();
});

balloonCopy.animate({ bottom: "100%"}, 8000);
};
balloon.remove();
```

index.html

✎ Exercise

Using the syntax below to setup up the score counter.

```
var balloon = $(“.balloon”);
var counter = 0;
...
balloonCopy.click(function() {
   $(this).remove();
   counter = counter + 1;
   $(“.counter”).html(counter);
});
```

index.html

✎ Exercise

Using the syntax below, pop some balloons and keep score

```
<div class = “counter”>
  0
 </div>
 <div class=“balloon”>
```

index.html

Extras

Extras

Extra Repetition

> Style the counter with CSS
> Make the loop run 15 times instead of 10
> Use something else instead of balloons
> Make different size balloons

Extras

Codaisseur

Extra Repetition

> Style the counter with CSS
> Make the loop run 15 times instead of 10
> Use something else instead of balloons
> Make different size balloons

Extra Enhancement

> Change the colours of your balloons
> Advanced animation (various speeds)
> Add sound
> Change your cursor to a crosshair
> Change the background to an image