

Turn into Coders

Workshop
Landing Page in HTML & CSS
for everyone

Coding Language(s)

HTML



CSS

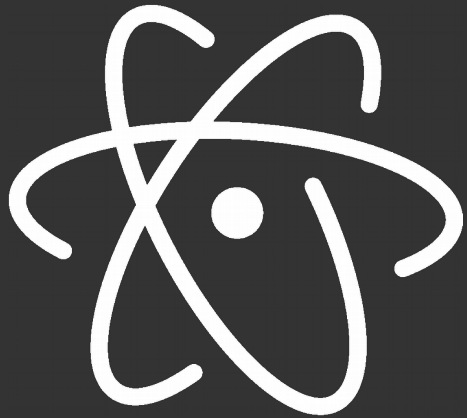


JS



Tools

ATOM.IO



**Google
Chrome**



Structure

HTML



Content and its **structure**

Design

CSS



Design and **style** of your content

Animation and logic

JS



Animate your content and
Implement (complex) **logic**

Skeleton

```
<!DOCTYPE html>
```

```
<html>
```

```
  <!-- html code -->
```

```
</html>
```

```
<!-- end of file -->
```

Head and Body

```
<!DOCTYPE html>
<html>
  <head>
    <!-- information and metadata -->
  </head>
  <body>
    <!-- content to display -->
  </body>
</html>

<!-- end of file -->
```


Head fundamentals

```
<!DOCTYPE html>
<html>
  <head>
    <title>Turn into Coders</title>
    <meta charset="utf-8">
  </head>
  <body>
    <!-- content to display -->
  </body>
</html>

<!-- end of file -->
```

Head Google example

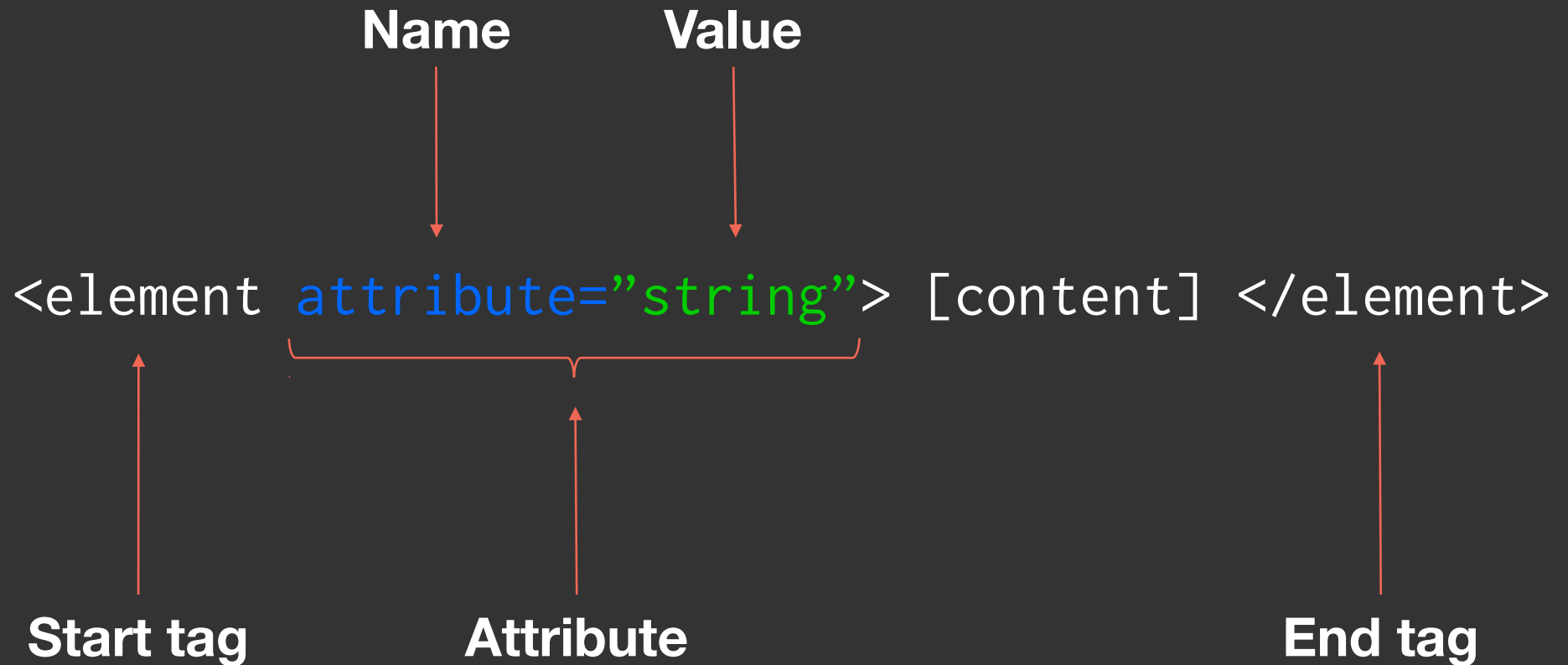
```
<!DOCTYPE html>
<html>
  <head>
    <title>Turn into Coders</title>
    <meta charset="utf-8">
    <meta name="description" content="Become a web dev...">
  </head>
  ...
```

Turn into Coders

www.turnintocoders.it/en/ ▼

Become a web developer with **Turn into Coders**, an intensive training program that makes you a JavaScript ninja and helps you find your first job in the web ...

HTML syntax fundamental



HTML examples

```
<body>
  <div>
    <h1>Turn into Coders</h1>
  </div>
  <div>
    
  </div>
</body>
```

CSS syntax fundamental

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

One or more elements can be selected to
apply style rules

Application of CSS rules

There are **three** ways to apply style rules to HTML content:

- In-line
- `<style>` section (internal stylesheet)
- Separate .css file (external stylesheet)

Application of CSS rules

In-line:

`style` is an **attribute** of HTML elements

```
<h1 style="font-family: HelveticaNeue;  
      font-size: 48px">Turn into Coders</h1>
```

Application of CSS rules

Internal stylesheet:

`<style>` can be a **section** of the HTML file

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      body {
        background-color: #333333;
      }
      h1 {
        color: white;
        font-family: HelveticaNeue;
      }
    </style>
  </head>
```


Application of CSS rules

External stylesheet:
A .css file can be **linked** to the HTML file



index.html

```
<head>
  <link rel="stylesheet" href="./style.css">
</head>
<body>
  <h1>Turn into Coders</h1>
</body>
```



style.css

```
h1 {
  color: rgb(255, 255, 255);
  font-family: HelveticaNeue;
}
```

Colors

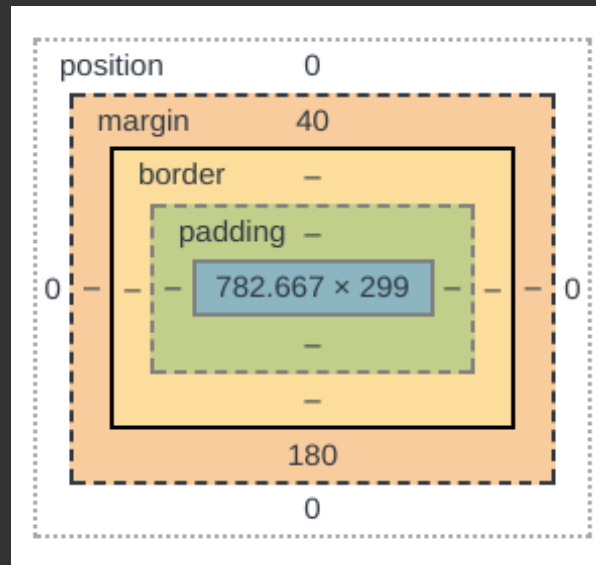
```
h2 {  
  color: red;  
  color: #FF0000;  
  color: #F00;  
  color: rgb(255, 0, 0);  
  color: rgba(255, 0, 0, 1.0);  
}
```

background-color **property** applies to the
background of the element

Font properties

font-size
font-weight
font-family
font-style
color
text-align
text-decoration
text-transform
line-height
letter-spacing

Div and Box model



Everything on a web page is a **box**
(or it is inside a box)

```
* {  
  border: 1px solid red !important;  
}
```

Positioning

Relative position:

- Offset from **initial** position
- Use it **to pin an element**

Positioning

Absolute position:

- Offset from 1st positioned parent
- If there is no pinned parent offset from body

id(s) and class(es)

```
<body>
  <div>
    
  </div>
  <div>
    <ul>
      <li></li>
      <li></li>
      <li></li>
      <li></li>
    </ul>
  </div>
</body>
```

How to resize the logo **only ?**

id(s) and class(es)

How to resize the logo **only** ?



style.css

```
img {  
  width: 100px;  
  height: 100px;  
}
```


id(s) and class(es)

How to resize the logo **only** ?



style

img

width: 100px;

height: 100px;

}



id(s) and class(es)

How to resize the logo **only** ?

```
<body>
  <div>
    
  </div>
  <div>
    <ul>
      <li></li>
      <li></li>
      <li></li>
      <li></li>
    </ul>
  </div>
</body>
```

id(s) and class(es)

How to resize the logo **only** ?



style.css

```
#logo {  
  width: 100px;  
  height: 100px;  
}
```

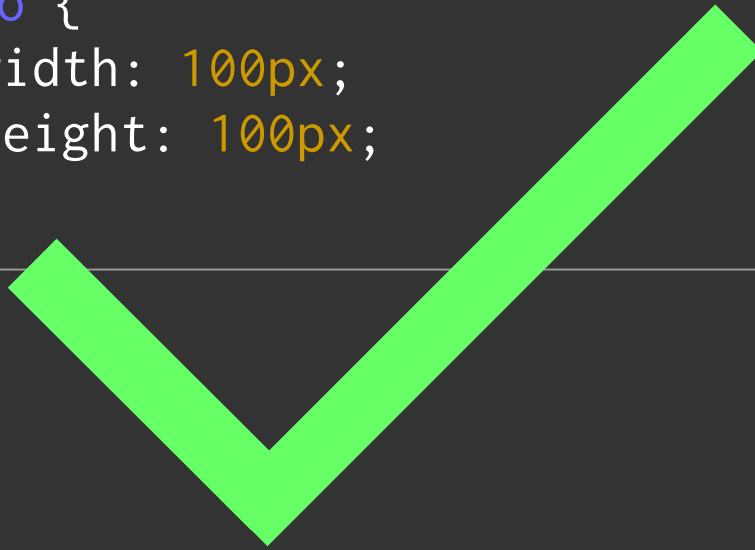
id(s) and class(es)

How to resize the logo **only** ?



style.css

```
#logo {  
  width: 100px;  
  height: 100px;  
}
```



id(s) and class(es)

```
<body>
  <div>
    
  </div>
  <div>
    <ul>
      <li></li>
      <li></li>
      <li></li>
      <li></li>
    </ul>
  </div>
</body>
```

How to resize **all** and only the **cities** ?

id(s) and class(es)

How to resize **all** and only the **cities** ?

```
<body>
  <div>
    
  </div>
  <div>
    <ul>
      <li></li>
      <li></li>
      <li></li>
      <li></li>
    </ul>
  </div>
</body>
```

id(s) and class(es)

How to resize **all** and only **cities** ?



style.css

```
#logo {  
    width: 100px;  
    height: 100px;  
}  
  
.city {  
    width: 200px;  
    height: 200px;  
}
```

id(s) and class(es)

How to resize **all** and only **cities** ?



style.css

```
#logo {  
  width: 100px;  
  height: 100px;  
}  
  
.city {  
  width: 200px;  
  height: 200px;  
}
```



id(s) or class(es) ?

unique element

```
<body>
  <div>
    
  </div>
  <div>
    <ul>
      <li></li>
      <li></li>
      <li></li>
      <li></li>
    </ul>
  </div>
</body>
```

Recurrent elements

Combine id(s) and class(es)

```
. . .  
<li></li>  
<li></li>  
<li></li>  
<li></li>  
. . .
```



style.css

```
#hq {  
    border: 2px solid red;  
}  
.city {  
    width: 200px;  
    height: 200px;  
}  
.shadow {  
    Box-shadow: 3px 3px 1px black;  
}
```

Media queries

Use the `@media` rule to include a **block** of CSS properties **only if a certain condition is true.**

```
@media (max-width: 760px) { /*mobile*/  
  
    /*set of CSS rules for screens smaller than 760 pixel (width)*/  
  
}
```

```
@media (min-width: 761px) { /*desktop*/  
  
    /*set of rules for screens wider than 761 pixels*/  
  
}
```