

# Modulo 2

# Web Tier

HTML - Day 1

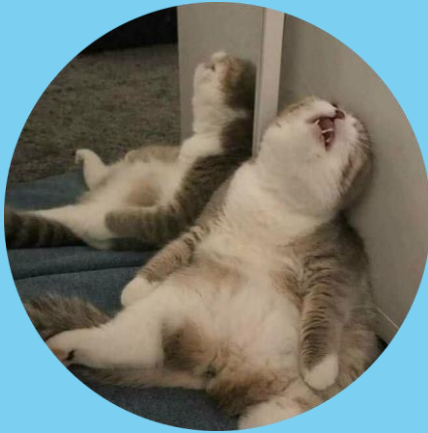
Presentato da  
*Vitale Esca*  
IBM Client Innovation Center - Italy

20/11/2023

IBM Client Innovation Center  
**Italy**

# Obiettivi del corso

Gli obiettivi di questo corso sono due:



**Non farvi annoiare o peggio ancora, farvi addormentare  
(*ci proviamo!*)**



**Fornirvi gli strumenti minimi per destreggiarsi nello sviluppo  
FrontEnd**

# Come affrontare questo corso!

Prestate attenzione al materiale didattico.

Alcune cose potrebbero sembrarvi ovvie o banali, ma vi assicuro che la maggior parte degli errori riguardano le cose più banali.



# Come affrontare questo corso!

Non temete di dire “*non ho capito*”.

Lo scopo di questa academy non è quello di giudicarvi, ma di **formarvi**. Per questo il vostro feedback costante per noi è essenziale



# Come affrontare questo corso!

Sporcatevi le mani e non abbiate paura di sbagliare.

Un approccio alle esercitazioni "*prova e sbaglia*" potrebbe aiutarvi a fissare i concetti affrontati in mattinata.



# Come affrontare questo corso!

*Attenzione alle fonti!*

**Spesso le slides riportano in basso un link utile** per approfondire la tematica trattata.

E' molto importante sapere dove e come recuperare una determinata informazione, ma attenzione ad utilizzare fonti attendibili!

# Agenda Day 1

- 1 Setup ambiente di sviluppo
- 2 Concetti HTML di Base
- 3 Quiz
- 4 Pratica



# Prerequisiti



**Installare *VS Code* come editor di testo**  
**<https://code.visualstudio.com/download>**



**Inizializzazione repository su GitHub**  
**e creazione del proprio branch individuale**



# Creazione branch individuale

Dopo aver ottenuto l'accesso al repository utilizzato per questo corso, effettuare il checkout tramite il seguente URL:

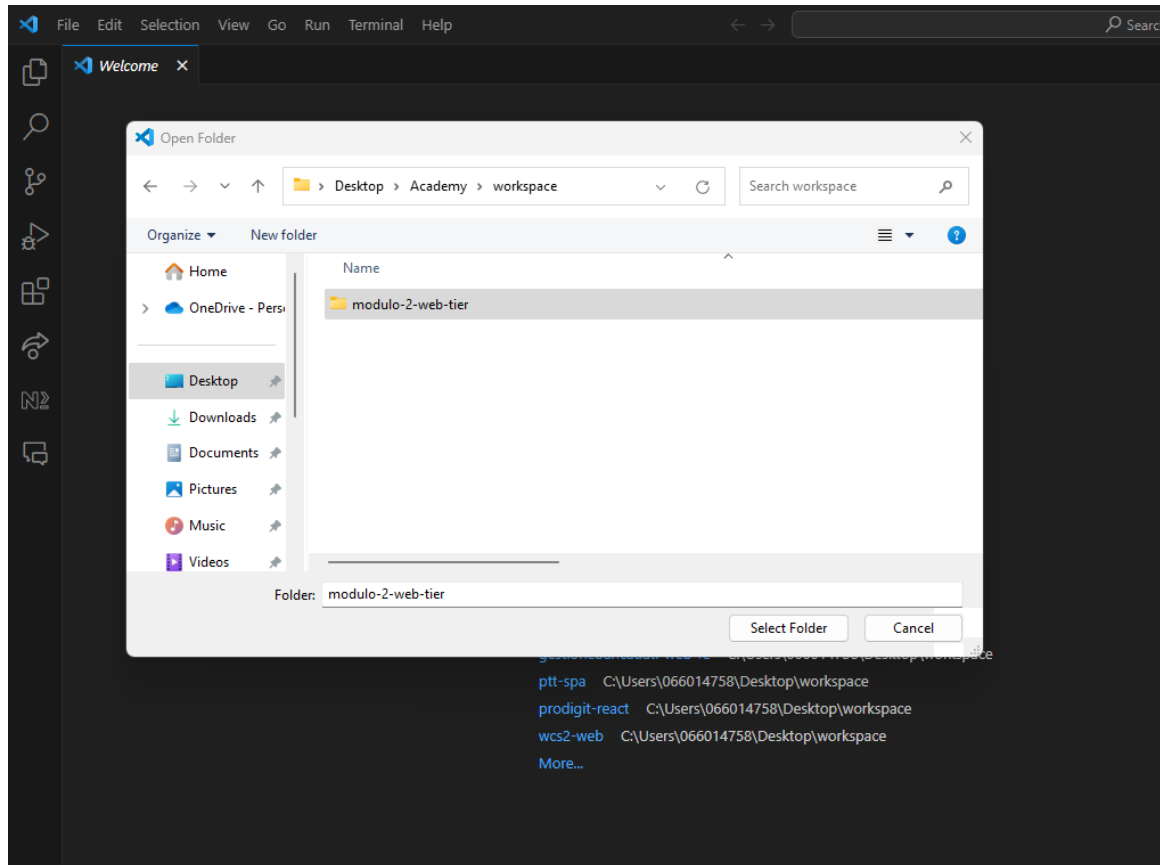
<https://github.com/ibm-cic-academy-2023-q4/es-html-day1.git>

- Creare un nuovo branch personale con il comando:  
**git checkout -b <NomeCognome>**
- Portare il branch appena creato sul repository remoto con il comando:  
**git push --set-upstream origin <NomeCognome>**

Day 1 – Setup ambiente di sviluppo

# Import Progetto su VS Code

Una volta clonato il repository, aprire VS Code e importare la folder del progetto.



# Struttura documento HTML

HTML sta per ***HyperText Markup Language***. Un documento HTML è composto da:

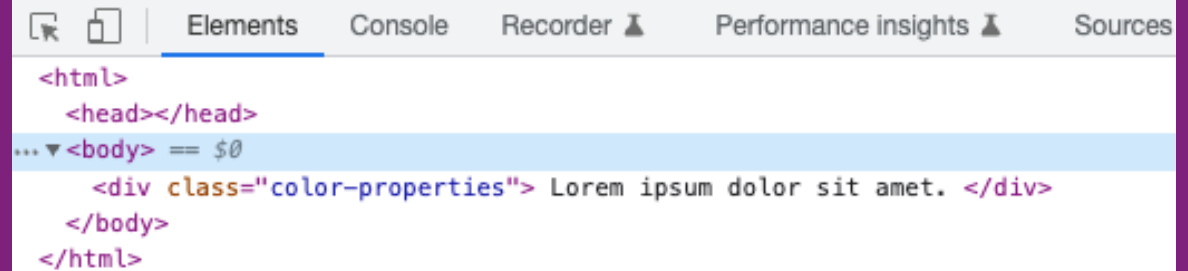
```
<!DOCTYPE html>
<html>
  <head>
    <title>Sample page</title>
  </head>
  <body>
    <h1>Sample page</h1>
    <p>This is a <a href="demo.html">simple</a> sample.</p>
    <!-- this is a comment -->
  </bod>
</html>
```

# Elemento *div*

- Definisce una sezione in un documento HTML
- Viene renderizzato dal browser come un *rettangolo*
- Affinchè sia renderizzato, è necessario definire le dimensioni, oppure aggiungere del contenuto
- Utilizzato per definire un blocco o un'area della pagina web

```
<div class="color-properties">  
    Lorem ipsum dolor sit amet.  
</div>
```

Lorem ipsum dolor sit amet.



```
<html>  
  <head></head>  
  ...▼ <body> == $0  
    <div class="color-properties"> Lorem ipsum dolor sit amet. </div>  
  </body>  
</html>
```

# Elemento *p*

- Definisce un paragrafo nel documento HTML
- Un elemento `<p></p>` non può contenere un altro elemento figlio `<p></p>`, a differenza del `<div>`, con cui è permesso l'annidamento illimitato
- È interscambiabile con `<div></div>`, ma non dovrebbe essere fatto
- Nel caso in cui sia definito un elemento successivo, il contenuto dell'elemento successivo sarà renderizzato in una nuova riga

```
<!DOCTYPE html>
<html>
<body>

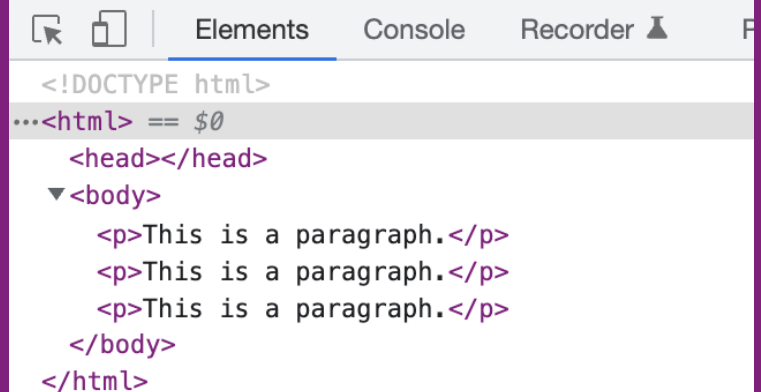
  <p>This is a paragraph.</p>
  <p>This is a paragraph.</p>
  <p>This is a paragraph.</p>

</body>
</html>
```

This is a paragraph.

This is a paragraph.

This is a paragraph.



```
<!DOCTYPE html>
...<html> == $0
  <head></head>
  ▼<body>
    <p>This is a paragraph.</p>
    <p>This is a paragraph.</p>
    <p>This is a paragraph.</p>
  </body>
</html>
```

# Elemento *span*

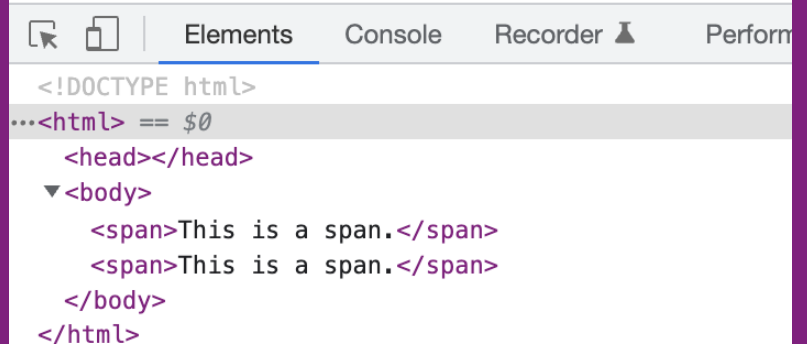
- L'elemento `<span>` va utilizzato per definire sezioni o aree in un documento HTML, ma è un elemento inline, al contrario del `<div>`
- Così come il `<div>`, gli `<span>` possono essere annidati

```
<!DOCTYPE html>
<html>
<body>

  <span>This is a span.</span>
  <span>This is a span.</span>

</body>
</html>
```

This is a span. This is a span.



```
<!DOCTYPE html>
...<html> == $0
  <head></head>
  ▼ <body>
    <span>This is a span.</span>
    <span>This is a span.</span>
  </body>
</html>
```

## Day 1 – Concetti HTML di Base

# I commenti in HTML

- Un commento si definisce con `<!--`, che rappresenta l'inizio del commento e `-->`, che definisce la fine del commento
- I commenti si utilizzano per non mostrare del contenuto, oppure per aggiungere commenti al codice HTML
- I commenti non vengono renderizzati dal browser

```
<!DOCTYPE html>
<html>
  <body>
    <!-- Inserisco qui un commento -->
    <h1>Ciao <!-- Inserisco qui un altro commento --> mondo!</h1>
  </body>
</html>
```

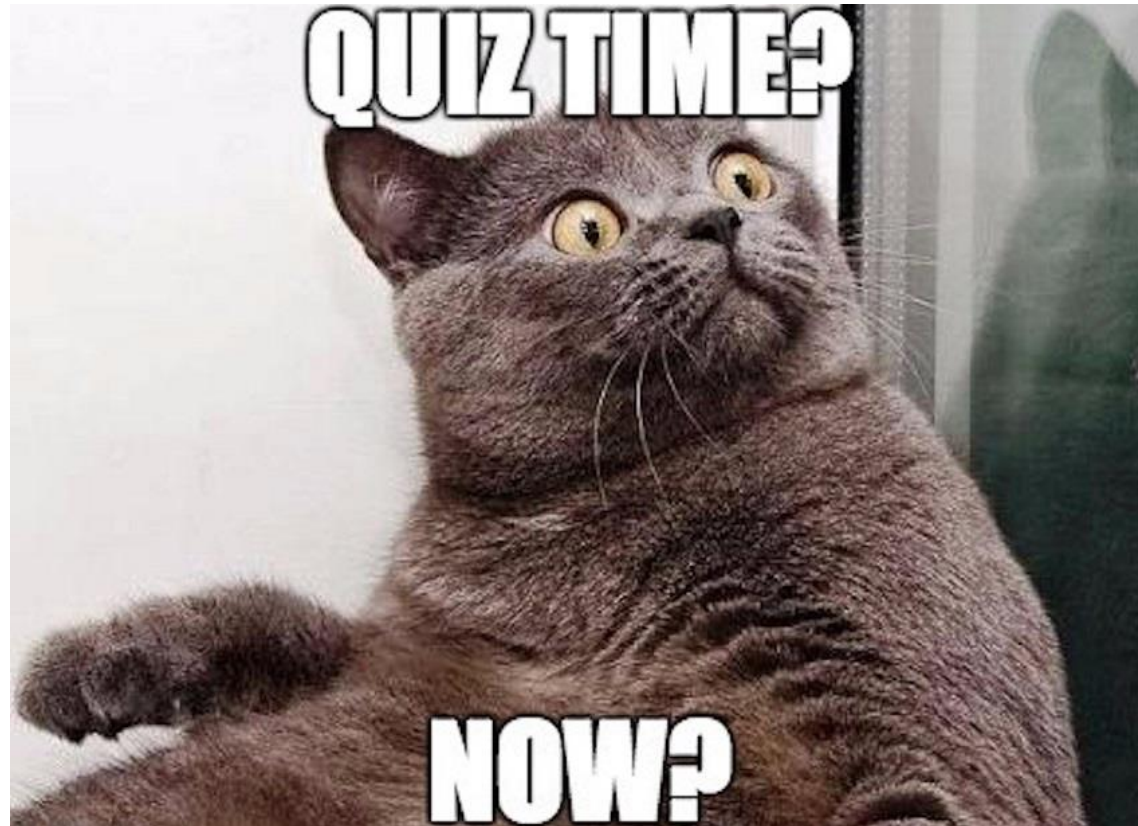
**Ciao mondo!**

```
...<!DOCTYPE html> == $0
<html>
  <head></head>
  <body>
    <!-- Inserisco qui un commento -->
    <h1>
      "Ciao "
      <!-- Inserisco qui un altro commento -->
      " mondo!"
    </h1>
  </body>
</html>
```

Per approfondimenti: [https://www.w3schools.com/html/html\\_comments.asp](https://www.w3schools.com/html/html_comments.asp)



## Day 1 – Quiz



## Day 1 – Concetti HTML di Base

# Elemento *form*, *input* e *label*

- `<form>` è utilizzato per creare form per gli input dell'utente

Gli elementi utilizzabili all'interno del tag `<form>` sono i seguenti:

- `<input>`
- `<textarea>`
- `<button>`
- `<select>`
- `<option>`
- `<fieldset>`
- `<label>`

Per approfondimenti:

[https://www.w3schools.com/tags/tag\\_form.asp](https://www.w3schools.com/tags/tag_form.asp)

<https://developer.mozilla.org/en-US/docs/Web/HTML/Element/input>

```
<!DOCTYPE html>
<html>
<body>

  <form action="/action_page.php">
    <label for="fname">First name:</label>
    <input type="text" id="fname" name="fname">
  </form>

</body>
</html>
```

First name:

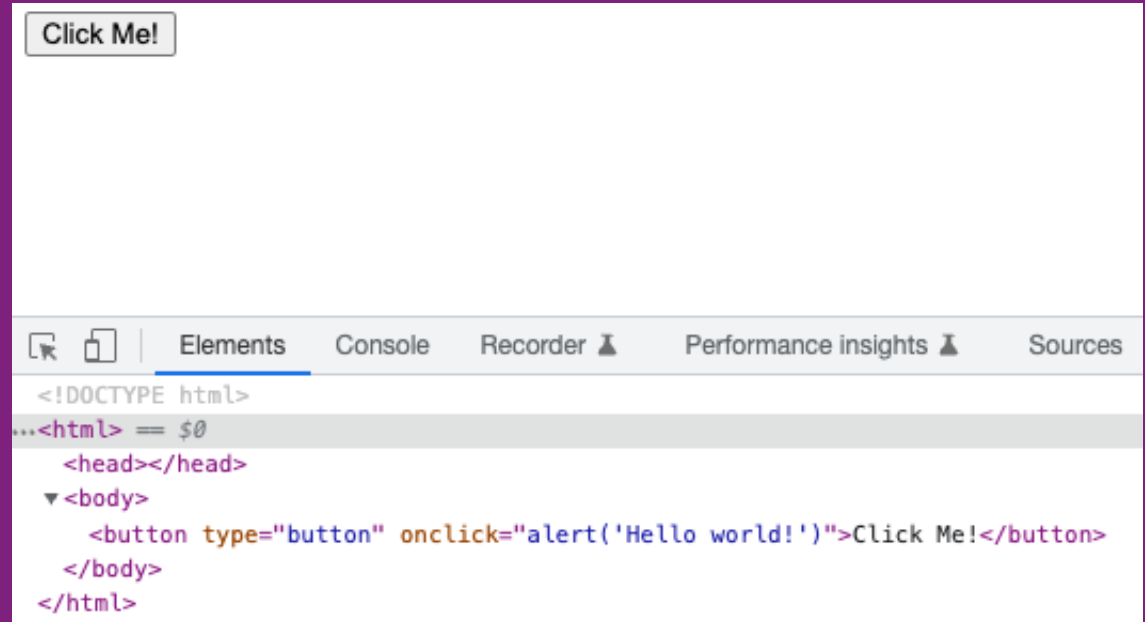
Elements Console Recorder Performance insi

```
<!DOCTYPE html>
...<html> == $0
  <head></head>
  ▼<body>
    ▼<form action="/action_page.php">
      <label for="fname">First name:</label>
      <input type="text" id="fname" name="fname">
    </form>
  </body>
</html>
```

# Elemento *button*

- Con `<button>` è possibile definire un elemento cliccabile
- Un *button* è possibile implementarlo anche con il tag `<input>`, la differenza è che il contenuto di `<input>` può essere esclusivamente testo, mentre all'interno del tag `<button>` può esserci codice HTML
- È consigliabile usare sempre l'attributo `type` per definire il tipo di button (button, reset, submit)

```
<!DOCTYPE html>
<html>
<body>
  <button
    type="button"
    onclick="alert('Hello world!')"
  >Click Me!</button>
</body>
</html>
</html>
```



# Definire una lista

- `<ol>` definisce una lista ordinata. È possibile ordinarla numericamente o alfabeticamente
- `<ul>` definisce una lista **non** ordinata. Pur definendo l'attributo `start`, la lista sarà renderizzata con i *bullet point*
- Le liste possono essere annidate

```
<!DOCTYPE html>
<html>
<body>

  <ol>
    <li>Coffee</li>
    <li>Tea</li>
    <li>Milk</li>
  </ol>

  <ol start="50">
    <li>Coffee</li>
    <li>Tea</li>
    <li>Milk</li>
  </ol>

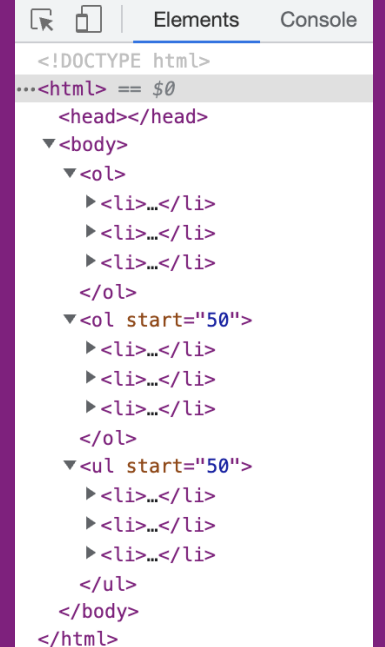
  <ul start="50">
    <li>Coffee</li>
    <li>Tea</li>
    <li>Milk</li>
  </ul>

</body>
</html>
```

1. Coffee  
2. Tea  
3. Milk

50. Coffee  
51. Tea  
52. Milk

- Coffee
- Tea
- Milk



# Definire una tabella

Una tabella HTML è composta da:

- `<table>` che definisce la tabella
- `<tr>` che definisce la riga
- `<th>` per definire l'header
- `<td>` per definire una cella

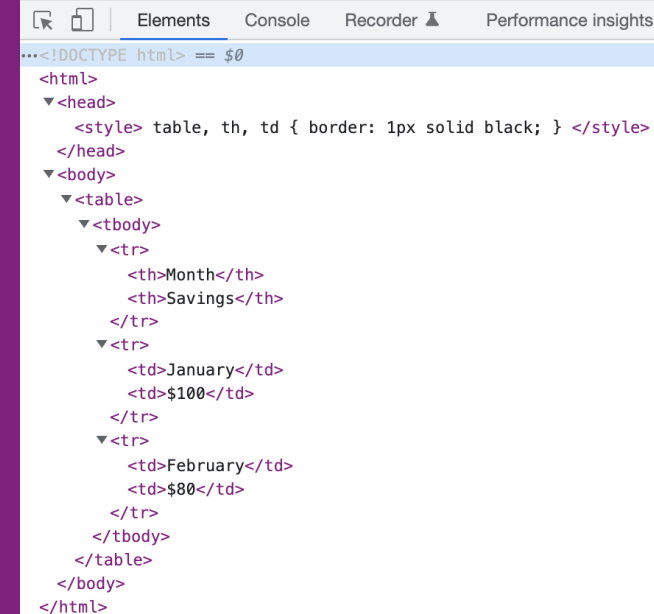
```
<!DOCTYPE html>
<html>
<head>
  <style>
    table,
    th,
    td {
      border: 1px solid black;
    }
  </style>
</head>

<body>

  <table>
    <tr>
      <th>Month</th>
      <th>Savings</th>
    </tr>
    <tr>
      <td>January</td>
      <td>$100</td>
    </tr>
    <tr>
      <td>February</td>
      <td>$80</td>
    </tr>
  </table>

</body>
</html>
```

Month	Savings
January	\$100
February	\$80



```
...<!DOCTYPE html> == $0
<html>
  <head>
    <style> table, th, td { border: 1px solid black; } </style>
  </head>
  <body>
    <table>
      <tbody>
        <tr>
          <th>Month</th>
          <th>Savings</th>
        </tr>
        <tr>
          <td>January</td>
          <td>$100</td>
        </tr>
        <tr>
          <td>February</td>
          <td>$80</td>
        </tr>
      </tbody>
    </table>
  </body>
</html>
```

# Elemento *image*

- Con il tag `<img>` è usato per incorporare un'immagine nel documento HTML
- Le immagini non sono incluse nel documento HTML, ma sono linkate. Il link dell'immagine va definito con l'attributo `src`
- È consigliato sempre specificare l'altezza e la larghezza dell'immagine, altrimenti la pagina HTML potrebbe spostarsi su e giù durante il rendering mentre viene scaricata l'immagine

```
<!DOCTYPE html>
<html>
  <body>
    <h1>The img element</h1>
    
  </body>
</html>
```

## The img element



Elements Console Network Performance insights Recorder Sources Performance Memory

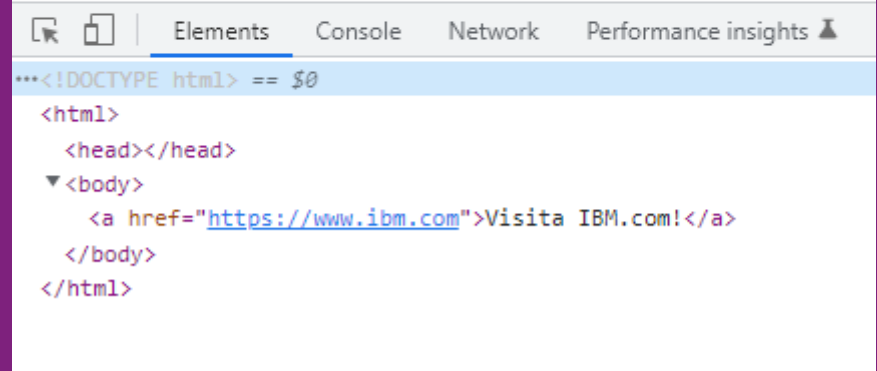
```
<!DOCTYPE html>
...<html> == $0
  <head></head>
  ▼ <body>
    <h1>The img element</h1>
    
  </body>
</html>
```

# Elemento *a*

- Con il tag `<a>` è possibile definire un *hyperlink*, ovvero un link che rimanda ad un'altra pagina
- Il link target è da specificare con l'attributo href
- Il testo definito nell'elemento `<a>` di default assume i seguenti colori:
  - blu e sottolineato: pagina **non visitata** o elemento non cliccato
  - viola e sottolineato: pagina **visitata**, quindi elemento cliccato
  - rosso e sottolineato: pagina **attiva**, ad esempio se trascino il link

```
<!DOCTYPE html>
<html>
  <body>
    <a href="https://www.ibm.com">Visita IBM.com!</a>
  </body>
</html>
```

Visita IBM.com!





# Elemento *h1* ... *h6*

- I tag **h1**, **h2**, **h3**, **h4**, **h5**, **h6** sono utilizzati per definire le intestazioni in una pagina HTML
- Sono ordinati per importanza, quindi h1 rappresenta l'intestazione più importante
- Una best practice è utilizzarne solo uno per pagina, partendo dal più importante perchè questo tag rappresenta l'intestazione principale dell'intera pagina.

## Day 1 – Concetti HTML di Base

# La formattazione del testo

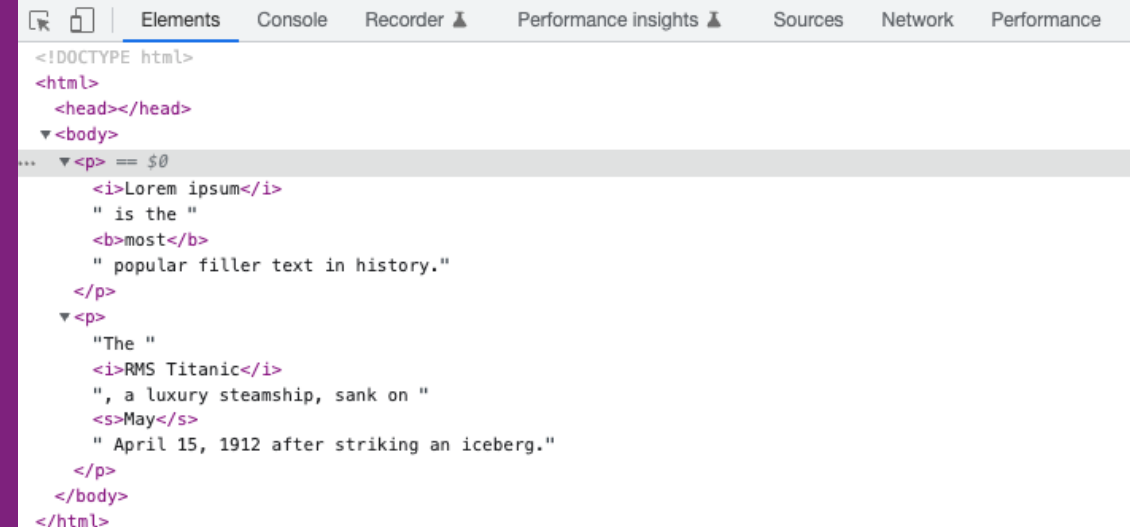
- Il tag `<b>` rende il testo in grassetto. È possibile definire un testo in grassetto con lo stile CSS, con la classe `font-weight: bold`
- Il tag `<i>` rende il testo corsivo. Lo stesso effetto si può ottenere con la classe CSS `font-style: italic`
- Il tag `<s>` rende il testo sbarrato

```
<!DOCTYPE html>
<html>
<body>
  <p><i>Lorem ipsum</i> is the <b>most</b>
popular filler text in history.</p>

  <p>The <i>RMS Titanic</i>, a luxury
steamship,
      sank on <s>May</s> April 15, 1912 after
striking an iceberg.</p>
</body>
</html>
```

*Lorem ipsum* is the **most** popular filler text in history.

The *RMS Titanic*, a luxury steamship, sank on ~~May~~ April 15, 1912 after striking an iceberg.



Per approfondimenti:

[https://www.w3schools.com/tags/tag\\_b.asp](https://www.w3schools.com/tags/tag_b.asp)

[https://www.w3schools.com/tags/tag\\_i.asp](https://www.w3schools.com/tags/tag_i.asp)

# HTML5

Dal 2014/2015 HTML subisce un importante aggiornamento che è conosciuto come HTML5.

HTML5 introduce alcuni tag particolari utili a suddividere semanticamente il contenuto di una pagina web e il suo utilizzo diventa raccomandato.

Le sezioni di una pagina web diventano molto più distinguibili tra loro, sia lato **codice** (ogni blocco ha un suo tag specifico), sia lato **semantico** (ogni blocco ha il nome di ciò che rappresenta).



# HTML5 – Tag semantici

I Tag semantici più utilizzati sono i seguenti:

- **header** – per identificare le intestazione all'interno delle pagina web
- **nav** – per definire aree dedicate alle navigazioni principali
- **section** - per definire una parte generica di una pagina
- **article** – per definire articoli o contenuti
- **aside** – per evidenziare del contenuto che presenta informazioni di supporto all'elemento article
- **footer** – per definire il piè di pagina di un intero documento HTML

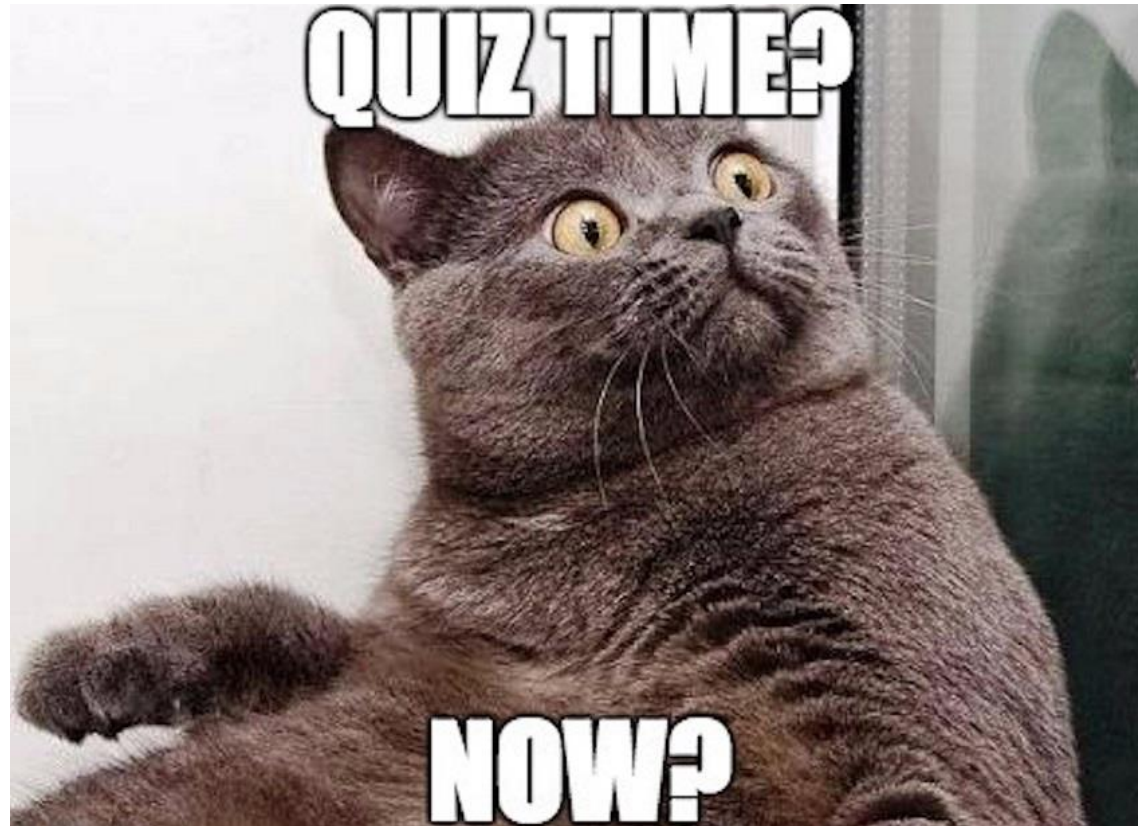
# HTML5 – Tag semantici

I principali vantaggi dell'utilizzo dei tag semantici sono la leggibilità e l'accessibilità. Nel seguente esempio vediamo uno stesso snippet di codice scritto prima senza l'ausilio dei tag semantici (a sinistra) e dopo utilizzando i tag semantici di HTML5 (a destra):

```
<div id="header"></div>
<div class="section">
  <div class="article">
    <div class="figure">
      <img>
      <div class="figcaption"></div>
    </div>
  </div>
</div>
<div id="footer"></div>
```

```
<header></header>
<section>
  <article>
    <figure>
      <img>
      <figcaption></figcaption>
    </figure>
  </article>
</section>
<footer></footer>
```

## Day 1 – Quiz





Experience.  
Create.  
Inspire.



# Thank You

*Vitale Esca*

IBM Client Innovation Center  
**Italy**