

# HTML

Riccardo Cattaneo



# HTML

- HTML è l'acronimo di **H**yper **T**ext **M**arkup **L**anguage
- Non è un linguaggio di programmazione, non possiede quindi i costrutti logici della programmazione
- E' un linguaggio di Markup per Ipertesti
- E' lo standard utilizzato per la creazione di pagine Web



# Struttura di un tag Html

`<nometag> ..... contenuto ..... </nometag>`

Estensione : come fa il pc a capire che si tratta di un file html? Dall'estensione del file che deve essere obbligatoriamente .html

Per fare una prova proviamo a creare un file chiamato **index.html** ed apriamolo con il nostro editor di testo (ad esempio Notepad++).

# Struttura principale di una pagina

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

# Esempio

```
<!DOCTYPE html>
<html>
  <head>
    <title> Mia Prima Pagina</title>
  </head>
  <body>
    <h1>Titolo della mia pagina</h1>
    <p>Questo è il testo descrittivo</p>
  </body>
</html>
```

# Risorse di Riferimento

<https://www.w3schools.com/>

Il sito ospita un'ampia raccolta di esempi di codice sorgente di vari linguaggi con tutorial gratuiti in inglese, solitamente modificabili e eseguibili interattivamente tramite un editor live.

# Tag Fondamentali

I Tag ci permettono di assegnare un ruolo e una disposizione ai contenuti delle nostre pagine. Possiamo ad esempio distinguere il titolo dai paragrafi di un articolo ecc.

I Tag sono costituiti da un tag di apertura e un tag di chiusura :

`<nometag> ..... contenuto ..... </nometag>`

Estensione : come fa il pc a capire che si tratta di un file html? Dall'estensione del file che deve essere obbligatoriamente `.html`

Per fare una prova proviamo a creare un file chiamato **index.html** ed apriamolo con il nostro editor di.

# <!DOCTYPE html>

Il *doctype* è (o dovrebbe essere) la prima riga di codice di un documento HTML e serve per indicare al browser il **tipo di documento** di cui si tratta.

Il termine *doctype*, infatti, è la contrazione di **Document Type Declaration** (DTD) e consiste in una dichiarazione, dello sviluppatore della pagina web, circa il linguaggio utilizzato, la versione di tale linguaggio e la lingua.



# <h1>

Il tag <h1> serve ad inserire dei titoli. Se proviamo ad esempio ad inserire subito dopo il tag body il seguente tag :

<h1> titolo dell'articolo </h1>

Possiamo notare che subito dopo la scritta “titolo dell'articolo”, tutto il contenuto che segue va a capo. I valori possibili sono h1, h2, h3, h4, h5, h6. Il suo utilizzo è consigliato perchè molto gradito dai motori di ricerca in fase di indicizzazione della pagina.

# Tag di tipo block

In Html esistono 2 tipi di tag, tag di tipo inline e tag di tipo block. Ora faremo un panoramica solo per far capire la differenza, mentre più Avanti verranno approfonditi tutti i tag nel dettaglio.

Prendiamo un testo qualsiasi dal web ed incolliamolo all'interno del nostro tag `<body>` (di solito si usa il sito <https://www.lipsum.com/>) di un file che chiameremo tag-block.html

```
<title>Elementi inline</title>
</head>
<body>
  Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nam rhoncus iaculis aug
  ζFusce condimentum sem fermentum dictum cursus. Donec non maximus lectus, vel po
  ζvehicula pulvinar. Duis orci neque, varius et arcu vel, euismod malesuada erat.
  ζsuscipit sed, vulputate porttitor mi. Pellentesque quam metus, congue eu viverra
  ζNulla posuere odio ut mauris tristique, nec bibendum lacus dictum. Nam ex dui,
  ζscelerisque ut risus. Pellentesque hendrerit ante ullamcorper quam imperdiet fr
  ζdiam eget imperdiet fermentum. Praesent ac ultricies tellus. Duis in leo non er
  ζpurus. Donec sem nibh, sodales vel efficitur a, iaculis nec tortor.
</body>
</html>
```

La differenza tra i tag di tipo block e i tag di tipo inline è che i tag di tipo block vanno a capo, mentre i tag di tipo inline non vanno a capo.

Ad esempio il primo tag che studieremo è il tag **<p>** perchè è uno dei tag più utilizzati, quindi è meglio prenderci dimestichezza da subito...

<p>

Il paragrafo è indicato dal tag <p> ed ha il tag di apertura e il tag di chiusura :

<p> ... contenuto ... </p>

Se proviamo ad esempio ad inserire il tag <p> solo in una parte del testo all'interno della pagina creata, vediamo quello che succede :

```
<title>Element1 inline</title>
</head>
<body>
<p>
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nam rhoncus iaculis auctor.
Fusce condimentum sem fermentum dictum cursus. Donec non maximus lectus, vel porta
vehicula pulvinar. Duis orci neque, varius et arcu vel, euismod malesuada erat. Sed
suscipit sed, vulputate porttitor mi.
</p>
    Pellentesque quam metus, congue eu viverra ut, porta venenatis ipsum. Nulla posu
tristique, nec bibendum lacus dictum. Nam ex dui, viverra nec bibendum in, sceleris
Pellentesque hendrerit ante ullamcorper quam imperdiet fringilla. Phasellus element
fermentum. Praesent ac ultricies tellus. Duis in leo non erat congue faucibus non v
nibh, sodales vel efficitur a, iaculis nec tortor.
```

---

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nam rhoncus iaculis auctor. lectus, vel porta orci. Vivamus auctor vehicula pulvinar. Duis orci neque, varius et arm.

Pellentesque quam metus, congue eu viverra ut, porta venenatis ipsum. Nulla posuere scelerisque ut risus. Pellentesque hendrerit ante ullamcorper quam imperdiet fringilla non erat congue faucibus non vel purus. Donec sem nibh, sodales vel efficitur a, iacu

Possiamo notare come il testo va a capo in corrispondenza della chiusura del tag. Questo perchè il tag `<p>` è di tipo block e quindi va a capo in automatico lasciando anche un margine sia in alto che in basso tra il tag `<p>` ed il resto del contenuto.

# <div>

Un altro tag di tipo block è il tag <div> che sta per “divisione”. Il div non ha alcun valore semantico, ciò vuol dire che non fa riferimento a nessun contenuto in particolare, per questo è stato usato come termine div... divisore!

<div> ... contenuto ... </div>

Proviamo ora a racchiudere solo una parte del testo rimanente all'interno del tag div e vediamo cosa succede...



```
<p>
```

```
    Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nam rhoncus iacu.  
    Fusce condimentum sem fermentum dictum cursus. Donec non maximus lectus,  
    vehicula pulvinar. Duis orci neque, varius et arcu vel, euismod malesuad  
    suscipit sed, vulputate porttitor mi.
```

```
</p>
```

```
<div>
```

```
    Pellentesque quam metus, congue eu viverra ut, porta venenatis ipsum.  
    tristique, nec bibendum lacus dictum. Nam ex dui, viverra nec bibendum i  
    Pellentesque hendrerit ante ullamcorper quam imperdiet fringilla. Phasel  
    fermentum. Praesent ac ultricies tellus. Duis in leo non erat congue fau  
    nibh, sodales vel efficitur a, iaculis nec tortor.
```

```
</div>
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nam rhoncus iaculis auctor. Nullam ut dui r  
lectus, vel porta orci. Vivamus auctor vehicula pulvinar. Duis orci neque, varius et arcu vel, euismod  
mi.

Pellentesque quam metus, congue eu viverra ut, porta venenatis ipsum. Nulla posuere odio ut mauris  
scelerisque ut risus. Pellentesque hendrerit ante ullamcorper quam imperdiet fringilla. Phasellus elem  
non erat congue faucibus non vel purus. Donec sem nibh, sodales vel efficitur a, iaculis nec tortor.

Pellentesque quam metus, congue eu viverra ut, porta venenatis ipsum. Nulla posuere odio ut mauris  
scelerisque ut risus.

Ora la domanda nasce spontanea : quando utilizzare il paragrafo e quando utilizzare il div ?  
Il paragrafo è stato creato per inserire del testo, quindi è un tag specifico, mentre il div è più generico.

# <h1>

Il tag <h1> è un altro tag di tipo block. Serve ad inserire dei titoli. Se proviamo ad esempio ad inserire subito dopo il tag body il seguente tag :

```
<h1> titolo dell'articolo </h1>
```

Possiamo notare che subito dopo la scritta “titolo dell'articolo”, tutto il contenuto che segue va a capo. Questo vale per tutti i tag h1, h2, h3, h4, h5, h6.

# <br/>

Esiste un tag che **non è un tag di tipo block**, ma soprattutto non ha apertura e chiusura, ed è il tag `<br/>` che significa break, e viene usato per andare a capo in un preciso punto.

Proviamo ad inserirlo all'interno del testo di prova finora utilizzato per vedere il risultato a video (possiamo notare che a differenza del tag paragrafo non ha margine!!!).

# Tag di tipo inline

Il tag inline è quell tag che non manda a capo l'elemento seguente, non ha una dimensione fissa e se li mettiamo uno a fianco a l'altro ... rimangono uno a fianco a l'altro!!!

Vediamo subito un esempio : proviamo a scrivere del testo di prova all'interno del tag body di una nuova pagina chiamata tag-inline.html

## <a>

Il tag <a> definisce un'ancora: possiamo dunque utilizzarlo sia per creare dei link a documenti o pagine esterne (attraverso l'attributo href) che per inserire uno o più collegamenti all'interno della stessa pagina.

Se proviamo ad inserire un tag <a> all'interno del nostro testo possiamo vedere che il testo non va a capo in quanto il tag <a> è un tag di tipo inline.

# <img>

Un altro tag di tipo inline è il tag `<img>` che viene utilizzato per includere immagini all'interno di una pagina web

```

```

Provare ad inserire un' immagine all' interno del nostro testo inserito in precedenza e vediamo cosa succede.

# <span>

Il tag span è un tag di tipo inline generico per il contenuto di frasi, che non ha alcun valore semantico. Può essere utilizzato per raggruppare gli elementi per scopi di stile... e lo vedremo più avanti

<span> testo di prova </span>



# Attributi dei TAG Html

Gli attributi HTML sono i parametri dei tag html. Sono utilizzati per indicare dei valori e modificare la formattazione del markup.

In genere gli attributi sono codici ( nome dell'attributo ) inseriti all'interno del tag html, sono seguiti dal simbolo uguale (=) e da un valore tra virgolette che definisce il parametro.

**<tag attributo="valore">**

Il valore dell'attributo è inserito tra doppie virgolette per facilitare l'operazione di interpretazione da parte del browser.

Gli attributi permettono di ottenere effetti diversi, ad esempio l'attributo bgcolor consente di cambiare il colore dello sfondo, l'attributo color assegna il colore del testo, l'attributo href assegna un indirizzo URL in un collegamento ipertestuale, ecc.

Ad esempio, nel seguente codice sorgente ho un testo compreso in un tag.

```
<p>testo da allineare a destra</p>
```

Sul browser il testo del paragrafo viene visualizzato a partire da sinistra, in quanto tutti i browser in occidente hanno la visualizzazione di default a sinistra

Ora voglio allineare a destra il testo compreso in un tag. Per farlo devo soltanto inserire nel tag <p> l'attributo ALIGN con il valore RIGHT.

**<p align="right">testo da allineare a destra</p>**

L'effetto sullo schermo del browser è il seguente:

**testo da allineare a destra**

Tramite l'uso dell'attributo ho modificato la formattazione del paragrafo e ho forzato la visualizzazione del testo a partire da destra.

# Elenchi Puntati e Numerati

Vediamo quali sono le tre tipologie di elenco disponibili in HTML :

- Elenchi **ordinati**
- Elenchi **non ordinati**
- Elenchi di **definizioni**

Tutti e tre i tipi di elenchi funzionano nel medesimo modo: si apre il tag, si elencano i vari elementi della lista (ciascuno con il proprio tag), si chiude il tag dell'elenco.

# <UL> gli elenchi non ordinati

L'elenco non ordinato è forse il più usato e si descrive utilizzando il tag **<ul>**. Al suo interno possiamo inserire gli elementi della lista utilizzando il tag **<li>**. Ecco un esempio :

*<ul>*

*<li>primo elemento</li>*

*<li>secondo elemento</li>*

*<li>terzo elemento</li>*

*</ul>*

# Annidare le liste non numerate

possiamo inserire diversi livelli all'interno delle liste, creando delle strutture "ad albero". E' sufficiente inserire un nuovo elenco all'interno di un elemento

`<ul>`

`<li>primo della 1a lista</li>`

`<li>secondo della 1a lista`

`<ul>`

`<li>primo della 2a lista</li>`

`<li>secondo della 2a lista`

`<ul>`

`<li>primo della 3a lista</li>`

`</ul>`

`</li>`

`<li>terzo della 2a lista</li>`

`</ul>`

`</li>`

`</ul>`

# <OL> gli elenchi ordinati

Gli elenchi ordinati sono contraddistinti dall'enumerazione degli elementi che compongono la lista. Il tag da utilizzare per aprire un elenco ordinato è **<ol>** e anche in questo caso gli elementi sono individuati dal tag **<li>** :

**<ol>**

*<li>primo elemento</li>*

*<li>secondo elemento</li>*

*<li>terzo elemento</li>*

**</ol>**

# Tabelle

Per definire una tabella possiamo riprodurre un esempio simile al seguente :

```
<table>
```

```
  <tr>
```

```
    <td>Colonna 1</td><td>Colonna 2</td>
```

```
  </tr>
```

```
  <tr>
```

```
    <td>Dato 1</td><td>Dato 1</td>
```

```
  </tr>
```

```
  <tr>
```

```
    <td>Dato 2</td><td>Dato 2</td>
```

```
  </tr>
```

```
  <tr>
```

```
    <td>Dato 3</td><td>Dato 3</td>
```

```
  </tr>
```

```
</table>
```



Nell'esempio riusciamo a definire una griglia formata da righe e colonne, il risultato è piuttosto povero ma ci aiuta a presentare i tag di base:

Tag	Descrizione
<b>&lt;table&gt;</b>	È il contenitore di tutta la tabella e la definisce
<b>&lt;tr&gt;</b>	<i>"table row"</i> Contiene una riga della tabella
<b>&lt;td&gt;</b>	<i>"table data"</i> Una cella che contiene i valori all'interno di una riga

Questo tipo di tabella era un elemento predominante nel “vecchio Web” (ed è ancora presente in quello attuale), perché funzionale anche alla suddivisione degli spazi, oltre che alla rappresentazione dei dati.

Con il consolidarsi di pratiche più specifiche per il markup dei layout, questo utilizzo delle tabelle è caduto in disuso.

Per scrivere una tabella che fornisca una rappresentazione più chiara dei dati introduciamo un template leggermente più ricco:

```
<table>
```

```
  <tr><th>Colonna 1</th><th>Colonna 2</th></tr>
```

```
  <tr><td>Totale 1</td><td>Totale 2</td></tr>
```

```
  <tr><td>Dato 1,1</td><td>Dato 1,2</td></tr>
```

```
  <tr><td>Dato 2,1</td><td>Dato 2,2</td></tr>
```

```
  <tr><td>Dato 3,1</td><td>Dato 3,2</td></tr>
```

```
</table>
```

# <th>

*“table header”* Indica una cella che contiene un'intestazione (ad esempio il titolo di una colonna o di una riga) e serve a dare una definizione dei dati cui si riferisce.

I miei dati

Colonna 1	Colonna 2
Dato 1,1	Dato 1,2
Dato 2,1	Dato 2,2
Dato 3,1	Dato 3,2
Totale 1	Totale 2

TH

# Annidare tabelle in HTML

Chiudiamo mostrando come è possibile annidare le tabelle le une dentro le altre e lo facciamo esaminando un semplice esempio

Campo	Tabella	
Campo1	Campo	Descrizione
	Campo1	Descrizione1

```
<table>
  <tr>
    <th>Campo</th><th>Tabella</th>
  </tr>
  <tr>
    <td>Campo1</td>
    <td>
      <table>
        <tr><th>Campo</th><th>Descrizione</th></tr>
        <tr><td>Campo1</td><td>Descrizione1</td></tr>
      </table>
    </td>
  </tr>
</table>
```

# I Form

L'**HTML** mette a disposizione una serie di tag il cui scopo è quello di creare **moduli** con **caselle di testo**, **opzioni**, **bottoni di invio** ed altro ancora.

Il termine tecnico che riassume tutti questi elementi è **form** ed il loro scopo è quello di creare interazione e scambio dati tra gli utenti ed il sito. Più precisamente, attraverso un **form HTML** è possibile "raccogliere" gli input degli utenti per poi inviarli ad un qualche sistema di elaborazione (normalmente uno script lato server che vedremo più avanti).

Accedi con il tuo account Google



Email o telefono

Avanti

[Trova il mio account](#)



## Iscriviti

È gratis e lo sarà sempre.

Nome

Cognome

Numero di cellulare o indirizzo e-mail

Nuova password

Data di nascita

14



gen



1993



[Perché devo fornire la mia data di nascita?](#)

☐ Donna

☐ Uomo

Cliccando su Crea account, accetti le nostre [Condizioni](#) e confermi di aver letto la nostra [Normativa sui dati](#), inclusa la [Normativa sull'uso dei cookie](#). Potresti ricevere notifiche tramite SMS da Facebook e puoi disattivare questa opzione in qualsiasi momento.

Crea account



# Il tag <form>

Per la creazione di un form utilizziamo l'omonimo tag **form**, come nell'esempio che segue:

```
<form method="" action="" name="">
```

*...*

```
</form>
```

Scopo di questo tag è di fare da contenitore ad una serie di tag che costituiranno gli specifici controlli del modulo.

I principali attributi del **tag form**, come visto nell'esempio, sono i seguenti:

- **method** - specifica il metodo di invio dei dati ed accetta i valori **get** o **post**;
- **action** - specifica lo script che riceverà, controllerà ed elaborerà i dati.
- **name** - specifica il nome attribuito al form, in quanto è possibile inserire in una pagina più form e questo tag serve ad identificarlo.

# I tag utilizzati per creare moduli HTML

Una volta definito un form mediante l'omonimo tag sarà necessario "popolarlo" con una serie di tag annidati al suo interno.

Attraverso i singoli tag, infatti, sarà possibile creare i vari elementi per l'interazione con l'utente come, ad esempio, caselle di testo o menu di selezione.

Passiamo adesso in rassegna i singoli tag che generano gli elementi dei form HTML :

# <input>

**input** - genera la maggior parte degli elementi dei form HTML, a seconda del **type** specificato. Gli input più utilizzati sono:

- **text** - è utilizzato per creare *caselle di testo* in cui l'utente può scrivere del contenuto su "singola linea";

Account:

- **file** - è utilizzato per creare *caselle di selezione di file in locale* al fine di poterli trasmettere al server remoto;

Nessun file selezionato

- **radio** - permette di creare un gruppo di opzioni al cui interno deve essere fatta una scelta (non ammette scelte multiple);

☒ male

☐ female

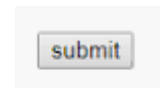
- **checkbox** - permette di creare un gruppo di opzioni al cui interno devono essere fatta delle scelte (ammette scelte multiple);



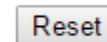
- **button** - permette di creare bottoni "neutri" ai quali, cioè, può essere associata un'azione mediante Javascript;



- **submit** - permette di creare bottoni di invio attraverso i quali viene, appunto inviato e processato il form;



- **reset** - permette di creare bottoni per il reset del form (in sostanza vengono cancellate le scelte effettuate dall'utente ed il modulo torna al suo stato iniziale).



## <select>

crea una casella di riepilogo a scorrimento, chiamata in gergo **selectbox**;

Select Country :

Select State :

## <textarea>

genera un'area di testo in cui è possibile andare a capo e viene utilizzata per permettere di inserire descrizioni, commenti o comunque testi piuttosto lunghi.

hello world

```
<form method="post" action="esegui.php">
  Nome <input type="text" name="nome"><br>
  Cognome <input type="text" name="cognome"><br>

  Paese<br>
  <select name="paese">
    <option value="I">Italia</option>
    <option value="E">Estero</option>
  </select><br>

  Sesso<br>
  <input type="radio" name="sesso" value="M"> M<br>
  <input type="radio" name="sesso" value="F"> F<br>

  Hobby<br>
  <input type="checkbox" name="hobby" value="S"> Sport<br>
  <input type="checkbox" name="hobby" value="L"> Lettura<br>
  <input type="checkbox" name="hobby" value="C"> Cinema<br>
  <input type="checkbox" name="hobby" value="I"> Internet<br>

  Commento <textarea name="commento" rows="5" cols="30"></textarea>
  <br><br>

  <!-- SUBMIT -->
  <input type="submit" name="invia" value="Invia i dati">
</form>
```

Nome

Cognome

Paese

Sesso

☐ M☐ F

Hobby

☐ Sport☐ Lettura☐ Cinema☐ Internet

Commento

Invia i dati



# Alcune osservazioni

- ogni elemento dei form HTML ha una caratteristica in comune, ovvero la specifica del **name** che dev'essere diverso da elemento a elemento ed uguale solo per gruppi di elementi (come nel caso dei radiobutton e dei checkbox);
- i tag **input** prevedono (oltre a *name*) due attributi principali:
  - *type*: che definisce il tipo di input;
  - *value*: definisce il valore dell'input; nei campi "text" è facoltativo e, se previsto, rappresenta una sorta di valore predefinito che gli utenti potranno modificare (a meno che il campo non sia stato impostato come *read only*).

- gli input **radio** e **checkbox** ammettono l'attributo *checked* qualora il campo debba comparire come selezionato al momento del caricamento della pagina;
- all'interno del tag *select* vengono annidati delle sequenze di tag **option** contenenti le varie opzioni disponibili all'interno della selectbox; questi tag hanno, normalmente, un attributo *value* con il valore da inviare nel caso l'opzione venga selezionata;
- i tag *option* ammettono l'attributo facoltativo *selected* attraverso il quale effettuare una selezione di default al momento del caricamento del modulo;
- il tag **textarea** prevede, oltre all'attributo *name*, gli attributi *rows* e *cols* che ne definiscono le linee e colonne.

# Tag <label>

Il tag **label** definisce un testo da associare ad un campo di un form HTML; si usa in particolar modo con i campi di tipo radio e checkbox. Consente di selezionare un'opzione anche cliccando sul suo testo descrittivo, non solo sul pulsante di opzione vero e proprio.

```
<input type="radio" name="sesso" id="U" value="U" />  
<label for="U">Uomo</label>  
<input type="radio" name="sesso" id="D" value="D" />  
<label for="D">Donna</label>
```

Supporta l'attributo **for** che riporta lo stesso valore dell'id del campo, che, allo scopo, si rende obbligatorio.

# L'attributo di Input: required

L'attributo required, come ci suggerisce il nome, ci consente di impedire l'invio del modulo se i campi obbligatori non sono stati compilati. Il suo funzionamento è davvero molto semplice :

```
<input type="text" name="user" required />
```

# Esercizio 1

## Benvenuti nel Nostro Sito!

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

## Crea un Nuovo Account

Nome	Cognome
Il tuo indirizzo e-mail	
Nuova Password	
Uomo <input type="radio"/> Donna <input type="radio"/>	
Come hai conosciuto il sito?	Consigliato da Amici ▼

## Parlaci di te!

☐ Accetto le [Condizioni d'Uso](#)

Crea Account

## Esercizio 2

Eseguire esercizio «la mia città» sostituendo le informazioni della città di Pescara con la propria città di appartenenza. (file pdf in allegato)

# Esercizio 3

- Creare un sito web utilizzando una tabella con le seguenti voci di menu : Home Page – Chi Siamo – Prodotti – Servizi – Contatti.
- Home Page : menu visibile in alto ed al centro della pagina visualizzare un'immagine a proprio piacimento.
- Chi Siamo : menu visibile in alto, subito sotto un testo descrittivo a proprio piacimento. Sotto il testo inserire un immagine centrata dell'azienda.

- Prodotti : menu visibile in alto, subito sotto un elenco di 5 prodotti composti da un'immagine ed un testo descrittivo (mettere l'immagine affiancata al testo)
- Servizi : menu visibile in alto, subito sotto un elenco di 5 servizi puntati non numerati a proprio piacimento
- Contatti : menu visibile in alto, subito sotto un form per essere contattati composto dai seguenti input : nome, cognome, email, messaggio con sotto il bottone invia.