

Guida HTML5: nuovi elementi semantici

In questo articolo scopriremo alcuni elementi di markup che sono stati rivisitati con l'introduzione di HTML5. In particolare vedremo come dare una formattazione alle date con il tag `<time>`, come evidenziare del testo attraverso `` e com'è cambiato l'output dei form JavaScript. Bando alle chiacchiere ed iniziamo!

Time

Navigando in rete ti sarà sicuramente capitato numerose volte di imbatterti in date e orari; per esempio, è prassi porre la data al termine di un articolo di un blog.

Prima dell'avvento dell'HTML5 non v'era modo di formattare le date in maniera standardizzata e questo, ovviamente, risultava essere un problema per i motori di ricerca che tentavano di accedere a tali informazioni per indicizzarle. Fortunatamente, con l'introduzione dell'**HTML5**, è stato preso in considerazione anche questo aspetto ed introdotto il tag `<time>`.

Il tag `<time>` riveste due differenti ruoli: in primo luogo indica in quale parte del documento è presente una **data** o un **orario** e, in secondo luogo, si occupa di dare una formattazione universale che riunisca i vari standard internazionali in un unico output comprensibile da ogni **software**.

Come molti di voi sapranno, la **formattazione standard** della data prevede **4 cifre** che indicano l'anno, seguite da 2 riferite al mese e ulteriori 2 dedicate al giorno.

In poche parole: **AAAA:MM:GG** o, in inglese, **YYYY:MM:DD**.

Questo tipo di formattazione non deve però riguardare per forza anche ciò che l'utente vede, ma solo ciò che i software interpretano. Per esempio potremmo voler scrivere:

in modo che l'utente italiano legga la data in un formato familiare ma i motori di ricerca la immagazzinino in maniera corretta.

Regole molto simili valgono per la formattazione degli orari, utilizzando il consueto formato: **HH:MM+00:00**, ovvero 2 cifre per l'ora seguite da 2 cifre per i minuti e dal segno + indicante il fuso orario di corrispondenza.

Ricordate di porre attenzione al fuso orario poiché non è un attributo opzionale! Per esempio:

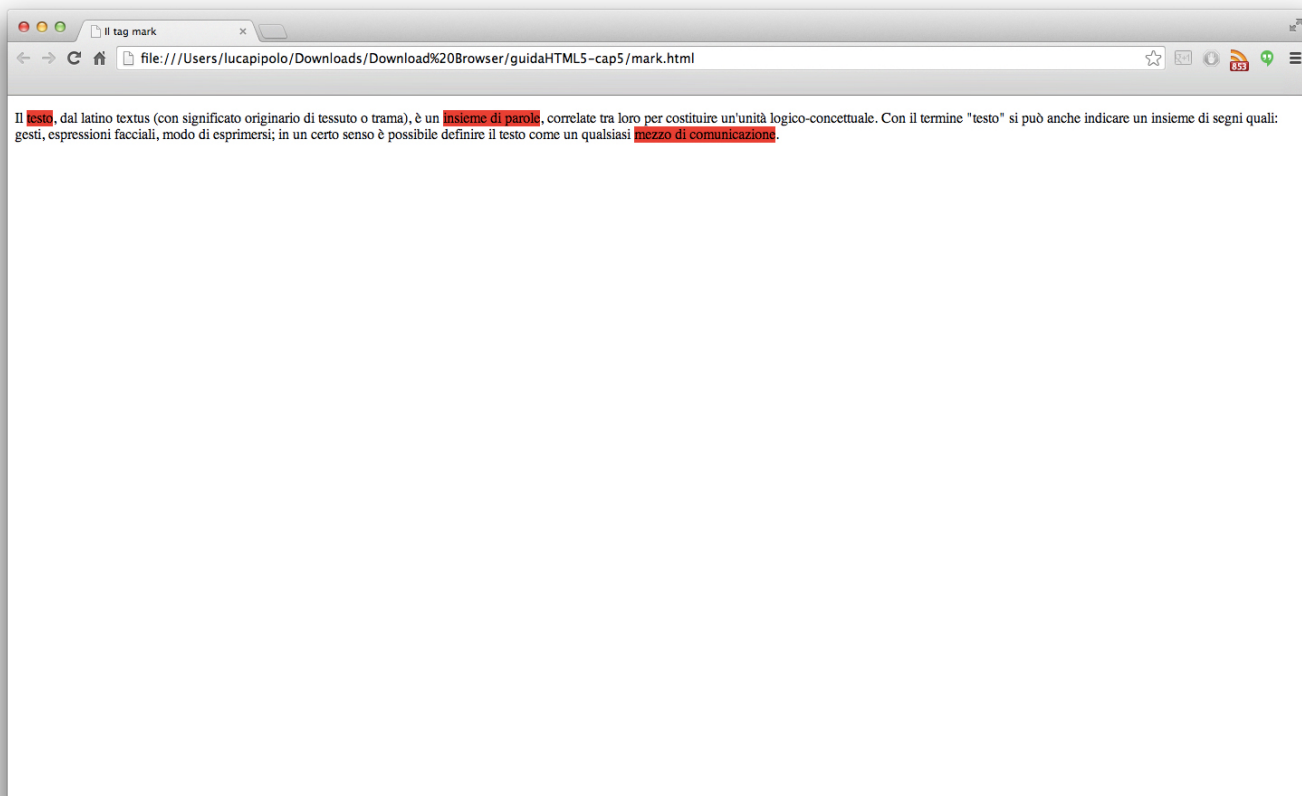
Con l'utilizzo della lettera **T** è anche possibile unire entrambe le opzioni per ottenere una data completa in questo modo:

Mark

Il tag `<mark>` ha lo scopo di evidenziare il testo racchiuso in esso. Può risultare particolarmente utile, per esempio, quando vogliamo portare l'attenzione dell'utente a quella specifica porzione di contenuto.

Il colore predefinito per il tag `<mark>` è il giallo ma è liberamente modificabile attraverso il CSS, per esempio:

Il risultato sarà:

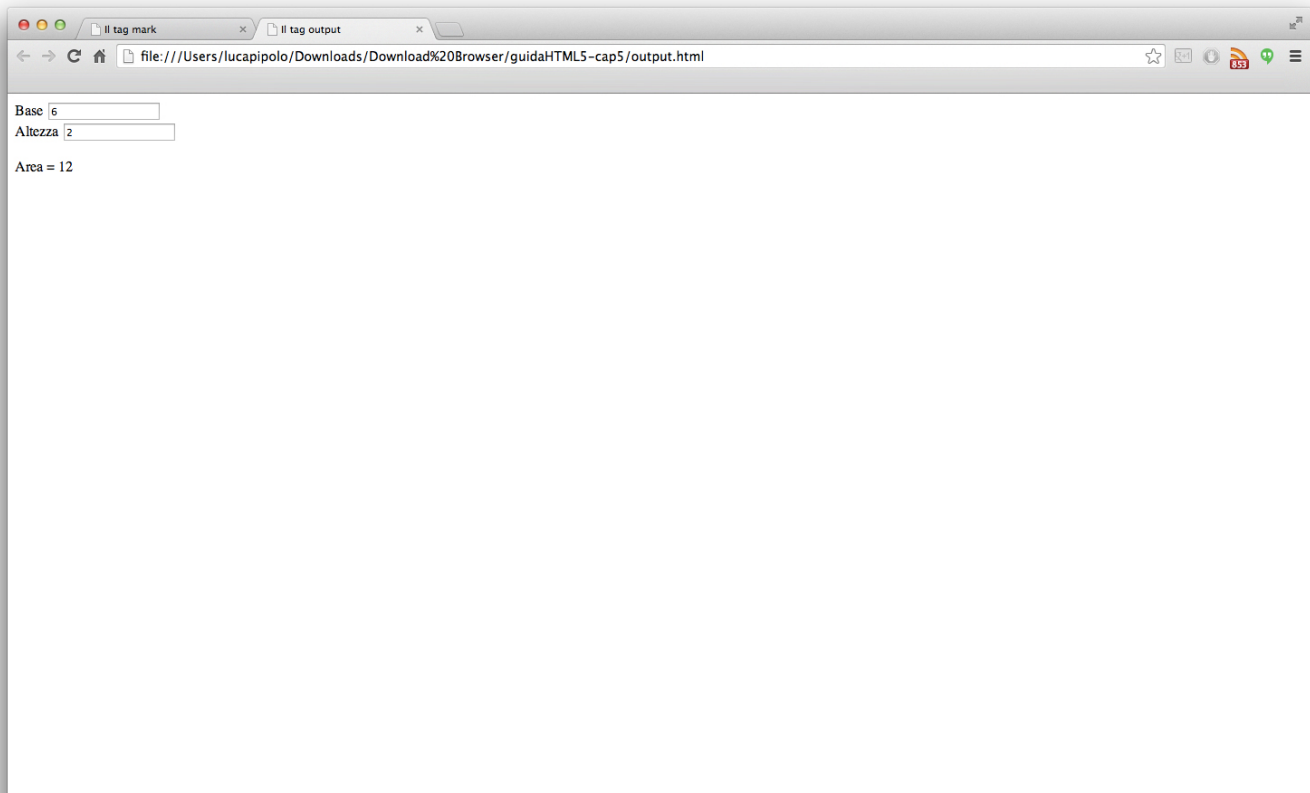


Attenzione va posta, anche in questo caso, per gli utenti di Internet Explorer. Fate riferimento, come consuetudine, al paragrafo '*compatibilità*' del precedente articolo '[Guida HTML5 - la struttura](#)'.

Output

Il tag `<output>` che non è altro che un segnaposto che il tuo **JavaScript** può utilizzare per visualizzare delle informazioni frutto di calcolo.

Ad esempio, immaginiamo di voler creare un modulo che, una volta inserita base ed altezza, restituisca l'area di un rettangolo:



Potremmo usare un codice come questo per il form:

Utilizziamo due `<input>`, entrambi caratterizzati dall'attributo `min` settato a `0` poiché non ci interessano valori negativi, per rilevare i parametri passati dall'utente. Effettuiamo l'operazione matematica in JavaScript e restituiamo il risultato nel tag `<output>` tramite `risultato.value`.

Potremmo inoltre voler aggiungere, per essere più precisi, al tag `<output>` alcuni attributi come `form` e `for`. Come puoi immaginare l'attributo `form` indica l'**ID** del **form** a cui il **tag output** si riferisce mentre il **tag for** segnala quali sono i **relativi controlli utilizzati**, separati da **virgola**.

Per esempio, facendo sempre riferimenti al codice per il calcolo dell'area del rettangolo, scriveremo:

Conclusioni

Nel corso di questo articolo abbiamo visto come utilizzare alcuni nuovi elementi semantici, caratteristici dell'HTML5. In particolare abbiamo introdotto un importante elemento, il **form**, che andremo ad approfondire nel prossimo articolo.

Come in tutti i precedenti articoli, puoi scaricare il codice completo di questa lezione attraverso [questo link](#).

GUIDA HTML5: GLI ARTICOLI

- 1) [Guida HTML5: Introduzione](#)
- 2) [Guida HTML5: la prima pagina](#)
- 3) [Guida HTML5: la struttura](#)
- 4) [Guida HTML5: Immagini e outlines](#)
- 5) [Guida HTML5: nuovi elementi semantici](#)
- 6) [Guida HTML5: i form – Parte 1](#)
- 7) [Guida HTML5: i form – Parte 2](#)
- 8) [Guida HTML5: i form – Parte 3](#)
- 9) [Guida HTML5: i form – Parte 4](#)
- 10) [Guida HTML5: i tag audio e video – parte 1](#)
- 11) [Guida HTML5: i tag audio e video – parte 2](#)
- 12) [Guida HTML5: I player video](#)
- 13) [HTML5: Il Canvas – Parte 1](#)
- 14) [Guida HTML5: Il Canvas - Parte 2](#)
- 15) [HTML5: Il Canvas - Parte 3](#)
- 16) [HTML5: Il Canvas - Parte 4](#)
- 17) [HTML5: Web storage](#)