

Incorporare una mappa con segnaposto nel vostro sito

Cos'è un segnaposto?

I segnaposto di Google (in inglese *markers*) vengono utilizzati per indicare una posizione su una mappa interattiva: cliccandoci sopra è possibile reperire informazioni supplementari riferite al luogo geografico in questione (indirizzo, indicazioni stradali, immagini correlate, ecc.). Mentre è molto facile incorporare in un sito web una mappa riportante un segnaposto noto (lo si può fare direttamente da [Google Maps](#))

Il codice: contenitore, CSS, scripts

Per prima cosa, serve definire all'interno della nostra pagina HTML il contenitore che ospiterà la mappa: in questo caso un `<div>` che identificheremo con un id del valore di `gmaps-canvas`.

```
<div id="gmaps-canvas"></div>
```

Le caratteristiche di tale contenitore, tra cui le dimensioni, sono definite direttamente nel file CSS:

```
div#gmaps-canvas {  
  
    width:775px;  
    height:300px;  
}
```

A questo punto serve importare un po' di codice dai server di Google, e per farlo basta inserire nel tag `<head>` della nostra pagina il seguente *script*:

```
<script type="text/javascript"  
src="http://maps.googleapis.com/maps/api/js?sensor=false"></script>
```

Al termine della riga di codice abbiamo indicato `sensor=false` (e non ad esempio `true`): il parametro `sensor` determina se il sito ha intenzione di usare un "sensore" per localizzare l'utente, cioè il visitatore. E' una finalità molto comoda in caso di applicazioni mobile, ma in questo caso non ci interessa. Non ci rimane adesso che impostare la *funzione* che andrà a creare fisicamente la nostra mappa e il nostro *marker*: anche essa deve essere posizionata all'interno del tag `<head>`.

```
<script type="text/javascript">  
    function initialize() {  
  
    }  
</script>
```

Inoltre, tale funzione deve essere "attivata" al caricamento stesso della pagina, e questo significa aggiungere l'attributo `onload="initialize()"` all'apertura del tag `<body>`:

```
<body onload="initialize()">
```

Creazione della mappa: posizione, zoom, centratura, tipo di mappa

La prima cosa da specificare all'interno della funzione `initialize()` è la *posizione*: dichiaro quindi la variabile `myLatLng`, che registra la latitudine e la longitudine esatte del luogo che voglio indicare (in questo caso `40.719614,-73.996251`).

```
var myLatLng = new google.maps.LatLng(40.719614,-73.996251);
```

Tali coordinate sono reperibili direttamente da [Google Maps](#): basta centrare il punto esatto cliccando con il tasto destro del mouse sulla mappa (*Fig. 1*), schiacciare l'icona del collegamento in alto a sinistra (la "catenella") e copiare negli appunti le coordinate esatte (*Fig. 2*).



Fig. 1: Una volta individuato il punto esatto, utilizzare il tasto destro del mouse per centrare la mappa

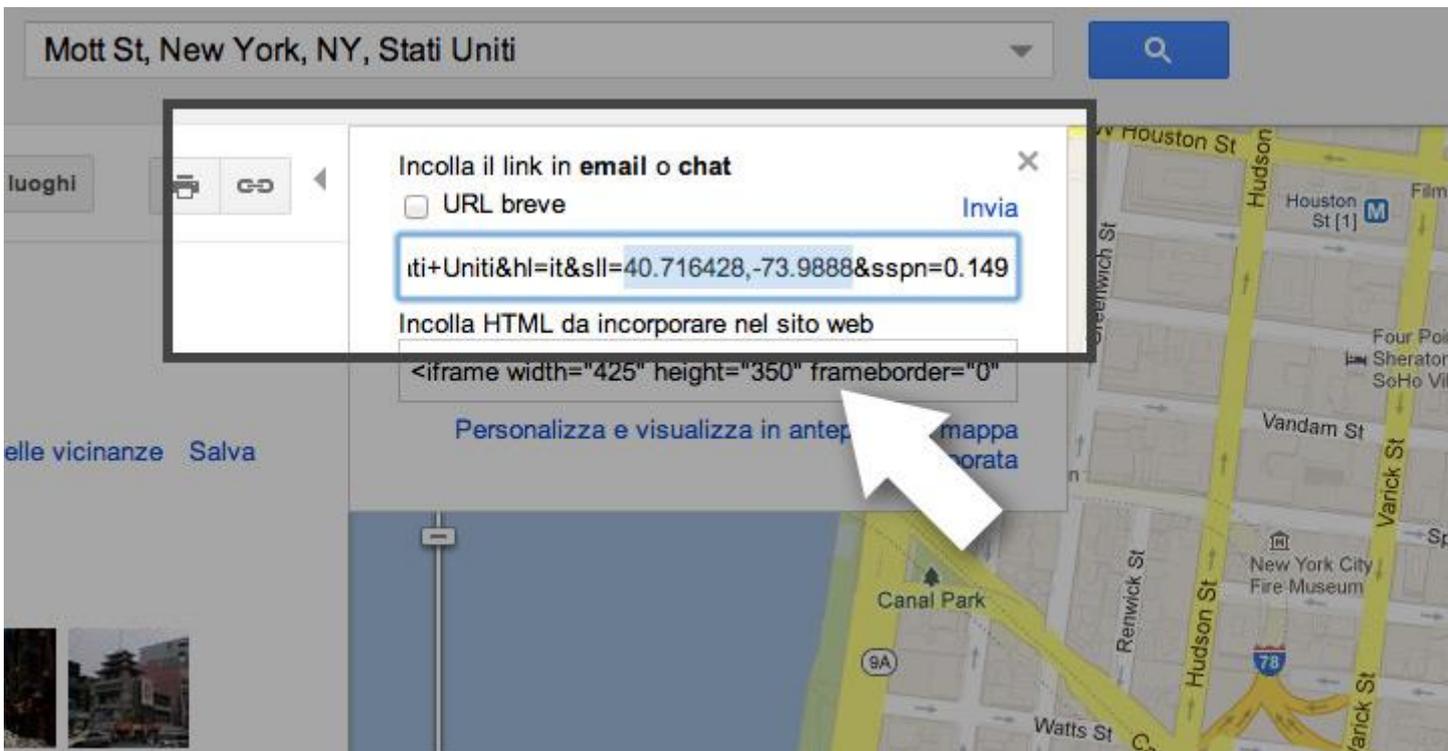


Fig. 2: Cliccare sull'icona del link per reperire le coordinate

Dentro la variabile `myOptions` dobbiamo specificare alcune cose: lo *zoom* (un numero compreso tra 0 e 18, dove 1 è la messa a fuoco più lontana e 18 quella più vicino al suolo), il *centro* della nostra mappa (sfruttiamo ancora il valore della variabile `myLatLng`), e il *tipo* di mappa che vogliamo rappresentare. Per

quest'ultima opzione utilizziamo ROADMAP (le alternative disponibili sono SATELLITE, HYBRID e TERRAIN).

```
var myOptions = {  
    zoom: 14,  
    center: myLatLng,  
    mapTypeId: google.maps.MapTypeId.ROADMAP  
}
```

L'ultima riga di codice serve infine a creare la mappa vera e propria all'interno del nostro <div> contenitore. In questo caso inseriamo 2 *parametri*: id dell'oggetto che conterrà la nostra mappa e variabile myOptions (che riporta le opzioni precedentemente specificate).

```
var map = new google.maps.Map(document.getElementById("gmaps-canvas"),  
myOptions);
```

Ecco quindi la funzione initialize() aggiornata con i comandi appena inseriti:

```
function initialize() {  
    var myLatLng = new google.maps.LatLng(40.719614,-73.996251);  
    var myOptions = {  
        zoom: 14,  
        center: myLatLng,  
        mapTypeId: google.maps.MapTypeId.ROADMAP  
    }  
  
    var map = new google.maps.Map(document.getElementById("gmaps-canvas"), myOptions);  
}
```

Nel caso ci interessasse mostrare solamente la mappa, potremmo fermarci qui.

Creazione del marker

Per creare un segnaposto, basta dichiarare una nuova variabile che chiamiamo *marker* della quale specifichiamo almeno *posizione* (usiamo ancora *myLatLng*) e *mappa* sulla quale il segnaposto si posizionerà (cioè la variabile *map*: la nostra mappa). Sebbene facoltativo, ho introdotto anche un parametro *title* a cui ho aggiunto del testo: quando il mouse passerà sopra il segnaposto, apparirà il classico riquadro giallo riportante il testo inserito.

```
var marker = new google.maps.Marker({  
    position: myLatLng,  
    map: map,  
    title:"Mi trovo a New York, nei pressi di Mott Street"  
});
```

Aggiungere la nuvoletta

La classica nuvoletta con le informazioni si chiama invece in gergo tecnico *info window*, e si crea con questi semplici comandi:

```
var contentString = '<em>Sono qui!</em>';  
var infoWindow = new google.maps.InfoWindow({  
    content: contentString  
});
```

In questo caso, la variabile *contentString*, richiamata all'interno della stessa variabile *infoWindow*, ospiterà il testo della nuvoletta (il linguaggio utilizzato è l'HTML). Rimane solo un'ultima cosa: fare in modo che al click del mouse sul segnaposto la nuvoletta si apra (in caso contrario rimarrà invisibile).

```
google.maps.event.addListener(marker, 'click', function() {  
    infoWindow.open(map,marker);  
});
```

E con la creazione di questo *evento*, l'esercizio è finito! Si potrebbero aggiungere al progetto altri mille aspetti, per i quali rimando alla [documentazione completa di Google Maps](#).

La pagina web al completo

Ecco qui tutto il codice finora creato per la nostra pagina web:

```
<!DOCTYPE html>
<html lang="it">

<head>

<title>Segnaposto personalizzato</title>
<style type="text/css">

    div#gmaps-canvas {

        width:775px;
        height:300px;
    }

</style>

<script type="text/javascript"
src="http://maps.googleapis.com/maps/api/js?sensor=false"> </script>
<script type="text/javascript">
function initialize() {

var myLatLng = new google.maps.LatLng(40.719614,-73.996251);
var myOptions = {
    zoom: 14,
    center: myLatLng,
```

```

    mapTypeId: google.maps.MapTypeId.ROADMAP
  }

  var map = new google.maps.Map(document.getElementById("gmaps-
canvas"), myOptions);

  var marker = new google.maps.Marker({
    position: myLatLng,
    map: map,
    title:"Mi trovo a New York, nei pressi di Mott Street"
  });

  var contentString = '<em>Sono qui!</em>';
  var infoWindow = new google.maps.InfoWindow({
    content: contentString
  });

  google.maps.event.addListener(marker, 'click', function() {
    infoWindow.open(map,marker);
  });
}
</script>

</head>

<body onload="initialize()">
<div id="gmaps-canvas"></div>
</body>

</html>

```

La [pagina dei contatti](#) di un progetto che ho realizzato di recente è un esempio funzionante di questo tutorial.

Aggiornamento (gennaio 2014): *navigando qua e là ho trovato [Google Map Builder](#), un tool per creare mappe di Google attraverso un'interfaccia visiva chiara e intuitiva. Per i progetti più semplici può essere davvero molto comodo.*

Approfondimenti

- [Creare una mappa interattiva con la libreria Leaflet](#) (una pratica alternativa a Google Maps)
- Per creare mappe personalizzate, puoi integrare [il servizio di Snazzy](#)
- [Google Map Builder](#)

di [Mattia Frigeri](#)