

JavaScript (7)



Marina Ribaudò, marina.ribaudò@unige.it

API Drag & Drop



2

- L'API Drag & Drop permette di **muovere gli elementi** all'interno di una pagina HTML
- Gli elementi che possono essere spostati devono avere la proprietà **draggable = true**
- Quando si muove un elemento (**drag**) vengono generati vari eventi
 - **dragstart** (cambia l'aspetto del cursore)
 - **drag** (ripetuto durante l'operazione)
 - **dragend**

API Drag & Drop



3

- Di default il browser **disabilita lo spostamento** degli elementi di una pagina
- Si può **cambiare questo comportamento**, rendendo le aree della pagina “drop zone” or “droppable” con il metodo **preventDefault()**

“The `preventDefault()` method of the `Event` interface tells the user agent that if the event does not get explicitly handled, its default action should not be taken as it normally would be.”

API Drag & Drop

4



- Quando si posiziona/rilascia un elemento su un'area valida (**drop**) vengono generati gli eventi
 - **dragenter**
 - **dragover**
 - **dragleave** oppure **drop**
- Il browser non modifica l'aspetto dell'elemento che si sta muovendo, ma si possono usare i CSS
- Sempre con i CSS si può modificare l'aspetto del cursore



<https://www.openstreetmap.org/>

5

Open Street Map



<https://leafletjs.com/>

6

- “*OpenStreetMap (OSM) is a **collaborative project to create a free editable geographic database of the world**. The geodata underlying the maps is considered the primary output of the project.*”
- La libreria JavaScript Leaflet permette di interagire con OpenStreetMap



GeoJSON

7

- GeoJSON è un **formato aperto** standard per la rappresentazione di **dati geospaziali** che si basa su JSON
- Permette di descrivere entità geografiche come **punti**, **linee** e **poligoni**, cui possono essere associate altre informazioni che caratterizzano le entità stesse

GeoJSON: Point

8

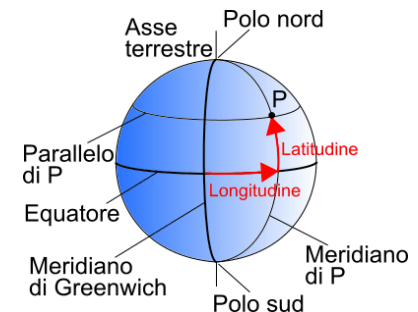
- Punto (**Point**)
 - Rappresenta un singolo punto nel piano cartesiano tridimensionale
 - Definito da una coppia di coordinate (latitudine e longitudine) e può includere anche l'altezza

```
{
```

```
  "type": "Point",
```

```
  "coordinates": [longitude, latitude, altitude]
```

```
}
```



GeoJSON: LineString

9

- Linea (**LineString**)
 - Sequenza ordinata di segmenti di linea
 - Può essere utilizzata per rappresentare strade, fiumi o qualsiasi altra entità lineare

```
{  
  "type": "LineString",  
  "coordinates": [ [100.0, 0.0], [101.0, 1.0] ]  
}
```

GeoJSON: Polygon

10

- Poligono (**Polygon**)
- Descrive un'**area chiusa** e può rappresentare aree geografiche come paesi, regioni, laghi o aree di interesse

```
{  
  "type": "Polygon",  
  "coordinates": [ [[100.0, 0.0], [101.0, 0.0], [101.0, 1.0], [100.0, 1.0], [100.0, 0.0]],  
                    // possibili poligoni interni per creare anelli  
                  ]  
}
```

GeoJSON: Feature

11

- **Feature**
 - Rappresenta una entità geospaziale, caratterizzata da una **geometria** che può essere un punto, una linea o un poligono
 - Oltre alla geometria, ci sono altre **proprietà** non standard
- Più feature rappresentano una **FeatureCollection**

GeoJSON: Feature

12

```
{
  "type": "Feature",
  "properties": {
    "COMUNE": "MATERA",
    "INDIRIZZO": "VIA SAN PARDO",
    "X": 635075.025400, "Y": 4504104.421000, "Z": 389.600000,
    "DESCRIZIONE": "CHIUSA",
    "SERBATOIO": "JAZZO GATTINI VECCHIO" },
  "geometry": {
    "type": "Point",
    "coordinates": [ 16.598280500055612, 40.676768336918862 ]
  }
}
```

Source: <http://dati.comune.matera.it/>

