

JavaScript (8)

STUDENTS@SAW at 3 PM or



Marina Ribaudò, marina.ribaudò@unige.it

API Geolocation

2

Concepts and usage

You will often want to retrieve a user's location information in your web app, for example to plot their location on a map, or display personalized information relevant to their location.

The Geolocation API is accessed via a call to `navigator.geolocation`; this will cause the user's browser to ask them for permission to access their location data. If they accept, then the browser will use the best available functionality on the device to access this information (for example, GPS).

API Geolocation

3

- Oggetto: **navigator.geolocation**
- Per di ottenere **latitudine** e **longitudine** del dispositivo dell'utente (GPS, Wi-Fi, IP)
- Per accedere a questa informazione il browser **chiede l'autorizzazione**
- Funziona solo in HTTPS

API Geolocation

4

- Possiamo verificare se l'oggetto esiste
(oggi dovrebbero essere supportato da tutti i browser)

```
if (navigator.geolocation) {  
    /* geolocation is available */  
}  
else {  
    /* geolocation IS NOT available */  
}
```

API Geolocation

5

- **geolocation.getCurrentPosition(success, error, options)**

*"[...]initiates an **asynchronous request to detect the user's position**, and queries the positioning hardware to get up-to-date information. When the position is determined, the **defined callback function** is executed. ..."*

- Ha tre parametri, solo il primo obbligatorio
 - **success** definisce la callback che viene chiamata in caso di successo
 - **error** definisce la callback che viene chiamata in caso di errore
 - **options**

<https://developer.mozilla.org/en-US/docs/Web/API/Geolocation/getCurrentPosition>

API Geolocation

6

- **geolocation.watchPosition(success, error, options)**

*“[...]the watchPosition() function, which has the **same input parameters** as getCurrentPosition(). **The callback function is called multiple times**, allowing the browser to either update your location as you move, or provide a more accurate location as different techniques are used to geolocate you. The error callback function, which is optional just as it is for getCurrentPosition(), can be called repeatedly.”*

<https://developer.mozilla.org/en-US/docs/Web/API/Geolocation/watchPosition>

API Geolocation



7

- I metodi dell'API Geolocation restituiscono un oggetto **GeolocationCoordinates** che permette di leggere
 - **latitude** e **longitude** dell'utente
 - **accuracy**, che indica il margine di errore in metri delle coordinate
 - **altitude** (se disponibile) rispetto al livello del mare
 - **heading** e **speed** (se disponibili) per conoscere la direzione e la velocità del dispositivo

Tabelle belle :)



8

4. DataTables <https://datatables.net/>

- **Features:** jQuery-based table library with sorting, filtering, pagination, and real-time updates. Suitable for applications where you need to display large datasets and handle them efficiently.
- **Pros:** Easy to use, good performance, many plugins available.
- **Cons:** Requires jQuery, which might be considered a downside in modern frameworks.
- **Website:** [DataTables](#)

5. Tabulator <https://tabulator.info/>

- **Features:** A feature-rich table library that supports sorting, filtering, pagination, real-time data updates, and exporting data to various formats. It is highly performant and works well with large datasets.
- **Pros:** Easy to use, feature-rich, highly customizable.
- **Cons:** Larger bundle size compared to some other libraries.
- **Website:** [Tabulator](#)

6. Grid.js

<https://gridjs.io/>

- **Features:** A lightweight, vanilla JavaScript table library that supports sorting, pagination, and real-time updates. It integrates well with React, Vue, and other frameworks.
- **Pros:** Lightweight, framework-agnostic, modern API.
- **Cons:** Fewer built-in features compared to more established libraries.
- **Website:** [Grid.js](#)