**HTML5 Geolocation**

The HTML Geolocation API is used to locate a user's position.

Locate the User's Position

The HTML Geolocation API is used to get the geographical position of a user.

Since this can compromise privacy, the position is not available unless the user approves it.

Note: Geolocation is most accurate for devices with GPS, like smartphone.

Browser Support

The numbers in the table specify the first browser version that fully supports Geolocation.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| API |  |  |  |  |  |
| Geolocation | 5.0 - 49.0 (http) 50.0 (https) | 9.0 | 3.5 | 5.0 | 16.0 |

Note: As of Chrome 50, the Geolocation API will only work on secure contexts such as HTTPS. If your site is hosted on an non-secure origin (such as HTTP) the requests to get the users location will no longer function.

Using HTML Geolocation

The getCurrentPosition() method is used to return the user's position.

The example below returns the latitude and longitude of the user's position:

Example

<script>  
var x = document.getElementById("demo");  
function getLocation() {  
  if (navigator.geolocation) {  
    navigator.geolocation.getCurrentPosition(showPosition);  
  } else {  
    x.innerHTML = "Geolocation is not supported by this browser.";  
  }  
}  
  
function showPosition(position) {  
  x.innerHTML = "Latitude: " + position.coords.latitude +   
  "<br>Longitude: " + position.coords.longitude;   
}  
</script>

Example explained:

Check if Geolocation is supported

If supported, run the getCurrentPosition() method. If not, display a message to the user

If the getCurrentPosition() method is successful, it returns a coordinates object to the function specified in the parameter (showPosition)

The showPosition() function outputs the Latitude and Longitude

The example above is a very basic Geolocation script, with no error handling.

Handling Errors and Rejections

The second parameter of the getCurrentPosition() method is used to handle errors. It specifies a function to run if it fails to get the user's location:

Example

function showError(error) {  
  switch(error.code) {  
    case error.PERMISSION\_DENIED:  
      x.innerHTML = "User denied the request for Geolocation."  
      break;  
    case error.POSITION\_UNAVAILABLE:  
      x.innerHTML = "Location information is unavailable."  
      break;  
    case error.TIMEOUT:  
      x.innerHTML = "The request to get user location timed out."  
      break;  
    case error.UNKNOWN\_ERROR:  
      x.innerHTML = "An unknown error occurred."  
      break;  
  }  
}

Displaying the Result in a Map

To display the result in a map, you need access to a map service, like Google Maps.

In the example below, the returned latitude and longitude is used to show the location in a Google Map (using a static image):

Example

function showPosition(position) {  
  var latlon = position.coords.latitude + "," + position.coords.longitude;  
  
  var img\_url = "https://maps.googleapis.com/maps/api/staticmap?center=  
  "+latlon+"&zoom=14&size=400x300&sensor=false&key=YOUR\_:KEY";  
  
  document.getElementById("mapholder").innerHTML = "<img src='"+img\_url+"'>";  
}

Location-specific Information

This page has demonstrated how to show a user's position on a map.

Geolocation is also very useful for location-specific information, like:

Up-to-date local information

Showing Points-of-interest near the user

Turn-by-turn navigation (GPS)

The getCurrentPosition() Method - Return Data

The getCurrentPosition() method returns an object on success. The latitude, longitude and accuracy properties are always returned. The other properties are returned if available:

|  |  |
| --- | --- |
| Property | Returns |
| coords.latitude | The latitude as a decimal number (always returned) |
| coords.longitude | The longitude as a decimal number (always returned) |
| coords.accuracy | The accuracy of position (always returned) |
| coords.altitude | The altitude in meters above the mean sea level (returned if available) |
| coords.altitudeAccuracy | The altitude accuracy of position (returned if available) |
| coords.heading | The heading as degrees clockwise from North (returned if available) |
| coords.speed | The speed in meters per second (returned if available) |
| timestamp | The date/time of the response (returned if available) |

Geolocation Object - Other interesting Methods

The Geolocation object also has other interesting methods:

watchPosition() - Returns the current position of the user and continues to return updated position as the user moves (like the GPS in a car).

clearWatch() - Stops the watchPosition() method.

The example below shows the watchPosition() method. You need an accurate GPS device to test this (like smartphone):

Example

<script>  
var x = document.getElementById("demo");  
function getLocation() {  
  if (navigator.geolocation) {  
    navigator.geolocation.watchPosition(showPosition);  
  } else {  
    x.innerHTML = "Geolocation is not supported by this browser.";  
  }  
}  
function showPosition(position) {  
  x.innerHTML = "Latitude: " + position.coords.latitude +   
  "<br>Longitude: " + position.coords.longitude;   
}  
</script>