

Cross-origin resource sharing *CORS* is a mechanism to allows the restricted resources from another domain in web browser

For suppose, if you click on *HTML5- video player* in html5 demo sections. it will ask camera permission. if user allow the permission then only it will open the camera or else it doesn't open the camera for web applications

Making a CORS request

here Chrome, Firefox, Opera and Safari all use the XMLHttpRequest2 object and Internet Explorer uses the similar XMLHttpRequest object, object.

```
function createCORSRequest(method, url) {
    var xhr = new XMLHttpRequest();

    if ("withCredentials" in xhr) {
        // Check if the XMLHttpRequest object has a "withCredentials" property.
        // "withCredentials" only exists on XMLHttpRequest2 objects.
        xhr.open(method, url, true);
    }

    else if (typeof XMLHttpRequest != "undefined") {
        // Otherwise, check if XMLHttpRequest.
        // XMLHttpRequest only exists in IE, and is IE's way of making CORS requests.
        xhr = new XMLHttpRequest();
        xhr.open(method, url);
    }

    else {
        // Otherwise, CORS is not supported by the browser.
        xhr = null;
    }
    return xhr;
}

var xhr = createCORSRequest('GET', url);
if (!xhr) {
    throw new Error('CORS not supported');
}
```

Event handles in CORS

Event Handler	Description
onloadstart	Starts the request
onprogress	Loads the data and send the data
onabort	Abort the request
onerror	request has failed
onload	request load successfully
ontimeout	time out has happened before request could complete
onloadend	When the request is complete either successful or failure

Example of onload or onerror event

```
xhr.onload = function() {
    var responseText = xhr.responseText;

    // process the response.
    console.log(responseText);
};

xhr.onerror = function() {
    console.log('There was an error!');
};
```

Example of CORS with handler

Below example will show the example of makeCorsRequest and onload handler

```
// Create the XHR object.
function createCORSRequest(method, url) {
    var xhr = new XMLHttpRequest();

    if ("withCredentials" in xhr) {

        // XHR for Chrome/Firefox/Opera/Safari.
        xhr.open(method, url, true);
    }

    else if (typeof XDomainRequest != "undefined") {
        // XDomainRequest for IE.
        xhr = new XDomainRequest();
        xhr.open(method, url);
    }

    else {
        // CORS not supported.
        xhr = null;
    }
    return xhr;
}

// Helper method to parse the title tag from the response.
function getTitle(text) {
    return text.match('<title>(.*?)</title>')[1];
}

// Make the actual CORS request.
function makeCorsRequest() {

    // All HTML5 Rocks properties support CORS.
    var url = 'http://www.tutorialspoint.com';

    var xhr = createCORSRequest('GET', url);

    if (!xhr) {
        alert('CORS not supported');
        return;
    }

    // Response handlers.
    xhr.onload = function() {
        var text = xhr.responseText;
        var title = getTitle(text);
        alert('Response from CORS request to ' + url + ': ' + title);
    };

    xhr.onerror = function() {
        alert('Woops, there was an error making the request.');
```

