# 04. AJAX Server Response.

## 4.1. The onreadystatechange Property

The **readyState** property holds the status of the XMLHttpRequest.

The **onreadystatechange** property defines a function to be executed when the readyState changes.

The **status** property and the **statusText** property holds the status of the XMLHttpRequest object.

| Property | Description |
|---|---|
| onreadystatechange | Defines a function to be called when the readyState property changes |
| readyState | Holds the status of the XMLHttpRequest.<br>0: request not initialized<br>1: server connection established<br>2: request received<br>3: processing request<br>4: request finished and response is ready |
| status | 200: "OK"<br>403: "Forbidden"<br>404: "Page not found"<br>...<br>For a complete list go to the Http Messages Reference |
| statusText | Returns the status-text (e.g. "OK" or "Not Found") |

The **onreadystatechange** function is called every time the readyState changes.

When **readyState** is 4 and **status** is 200, the response is ready:

**Example**:

```
function loadDoc() {
    var xhttp = new XMLHttpRequest();
    xhttp.onreadystatechange = function() {
        if (this.readyState == 4 && this.status == 200) {
            document.getElementById("demo").innerHTML =
            this.responseText;
       }
```

```
        };
        xhttp.open("GET", "ajax_info.txt", true);
        xhttp.send();
    }
```

The **onreadystatechange** event is triggered four times (1-4), one time for each change in the readyState.

## 4.2. Using a Callback Function

A callback function is a function passed as a parameter to another function.

If you have more than one AJAX task in a website, you should create one function for executing the XMLHttpRequest object, and one callback function for each AJAX task.

The function call should contain the URL and what function to call when the response is ready.

**Example**:

```
    loadDoc("url-1", myFunction1);

    loadDoc("url-2", myFunction2);

    function loadDoc(url, cFunction) {
      var xhttp;
      xhttp = new XMLHttpRequest();
      xhttp.onreadystatechange = function() {
        if (this.readyState == 4 && this.status == 200) {
          cFunction(this);
        }
      };
      xhttp.open("GET", url, true);
      xhttp.send();
    }

    function myFunction1(xhttp) {
      // action goes here
    }

    function myFunction2(xhttp) {
      // action goes here
    }
```

## 4.3. Server Response Properties

| Property | Description |
|---|---|
| responseText | get the response data as a string |
| responseXML | get the response data as XML data |

## 4.4. Server Response Methods

| Method | Description |
|---|---|
| getResponseHeader() | Returns specific header information from the server resource |
| getAllResponseHeaders() | Returns all the header information from the server resource |

## 4.5. The responseText Property

The responseText property returns the server response as a JavaScript string, and you can use it accordingly:

**Example**:

```
document.getElementById("demo").innerHTML = xhttp.responseText;
```

## 4.6. The responseXML Property

The XML HttpRequest object has an in-built XML parser.

The responseXML property returns the server response as an XML DOM object.

Using this property you can parse the response as an XML DOM object:

**Example**:

```
xmlDoc = xhttp.responseXML;
txt = "";
x = xmlDoc.getElementsByTagName("ARTIST");
```

```
for (i = 0; i < x.length; i++) {
  txt += x[i].childNodes[0].nodeValue + "<br>";
  }
document.getElementById("demo").innerHTML = txt;
xhttp.open("GET", "cd_catalog.xml", true);
xhttp.send();
```

You will learn a lot more about XML DOM in the DOM chapters of this tutorial.

## 4.7. The getAllResponseHeaders() Method

The *getAllResponseHeaders*() method returns all header information from the server response.

**Example**:

```
var xhttp = new XMLHttpRequest();
xhttp.onreadystatechange = function() {
  if (this.readyState == 4 && this.status == 200) {
    document.getElementById("demo").innerHTML =
    this.getAllResponseHeaders();
  }
};
```

## 4.8. The getResponseHeader() Method

The *getResponseHeader*() method returns specific header information from the server response.

**Example**:

```
var xhttp = new XMLHttpRequest();
xhttp.onreadystatechange = function() {
  if (this.readyState == 4 && this.status == 200) {
    document.getElementById("demo").innerHTML =
    this.getResponseHeader("Last-Modified");
  }
};
xhttp.open("GET", "ajax_info.txt", true);
xhttp.send();
```