

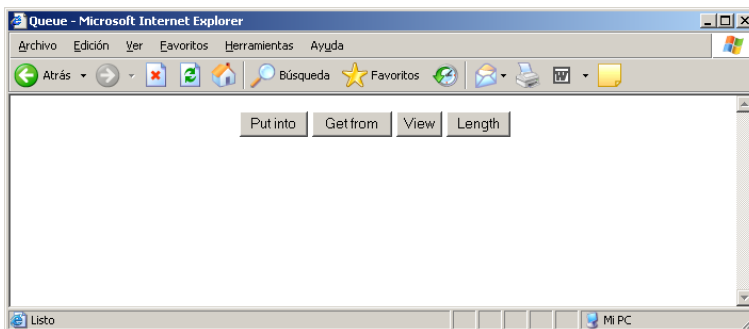
AJAX TECHNOLOGY IN WEB PROGRAMMING

EXERCISES

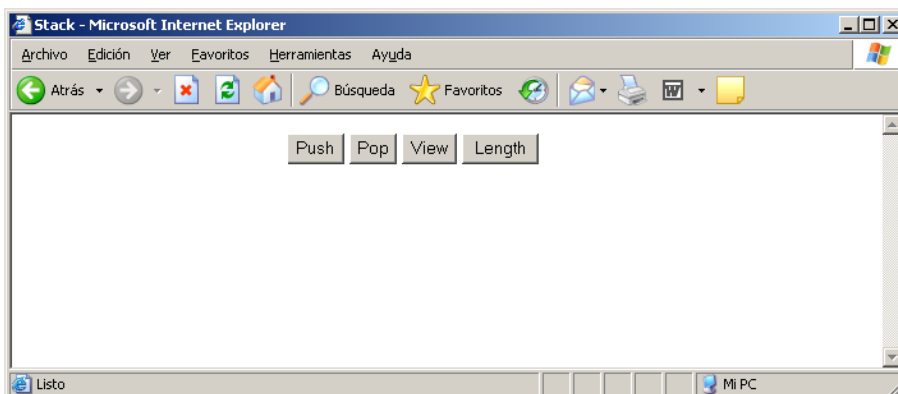
JAVASCRIPT

1. Define a queue and stack using an array:

Queue interface:



Stack interface:

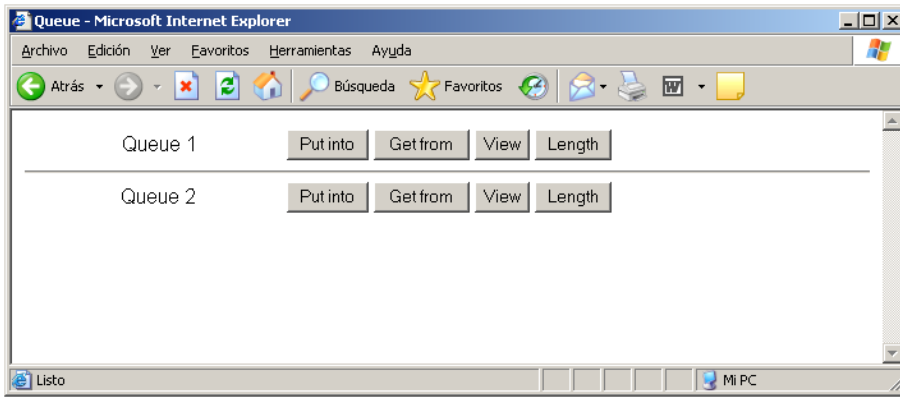


TIPS:

```
var stack = new Array();

function push()
{
    stack.push(prompt("Element value:"));
}
```

2. Encapsulate the previous code in your own object prototype, in order to create multiples objects:



TIPS:

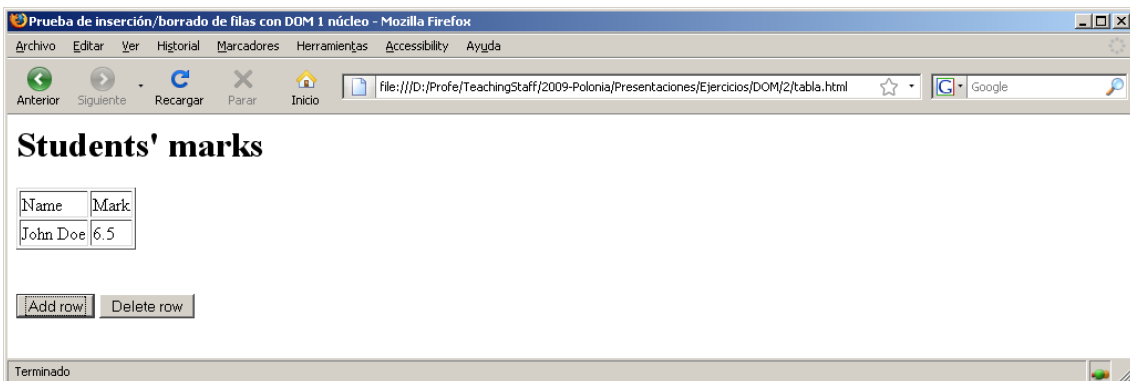
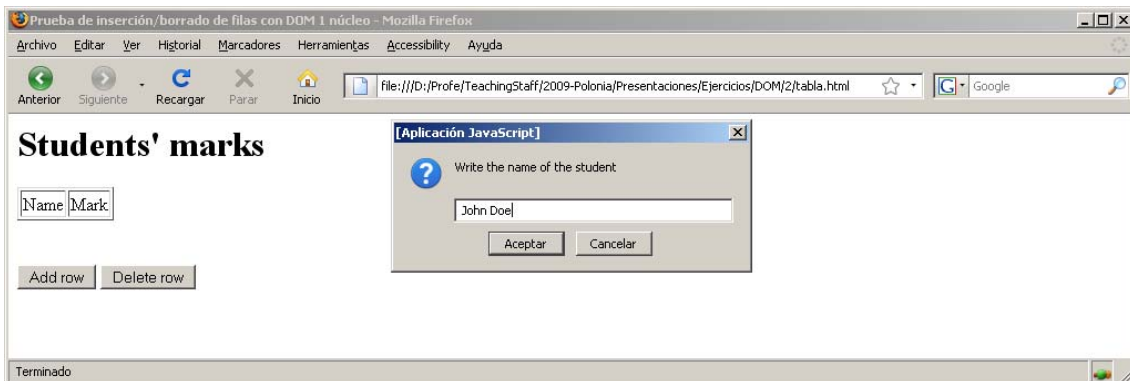
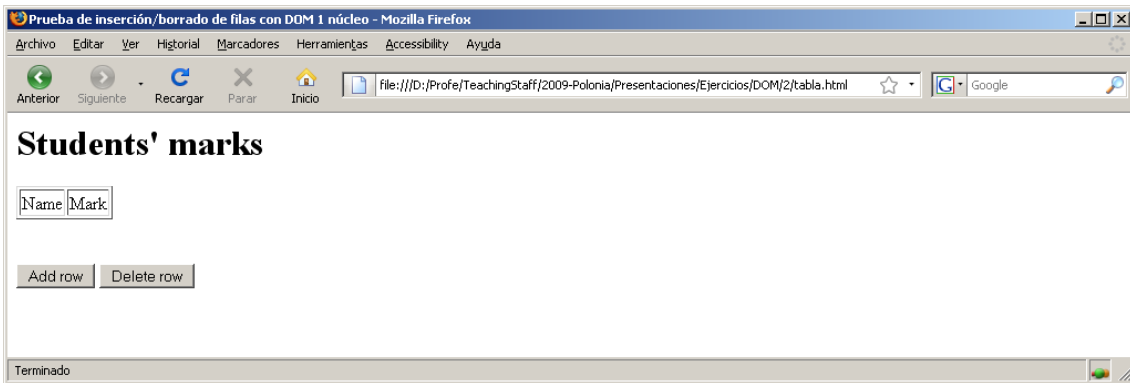
```
// Stack object
function Stack()
{
    this.stack = new Array();
    this.push = stack_push;
    this.pop = stack_pop;
    this.view = stack_view;
    this.length = stack_length;
}
```

DOM

1. Validate a web form:

- Non-empty field (mandatory field)
- Alphabetic field
- Numeric field
- Date field

2. Add rows into a table:



TIPS:

There are two possibilities: DOM 1 Core functions or DOM 1 HTML functions

DOM 1 Core functions

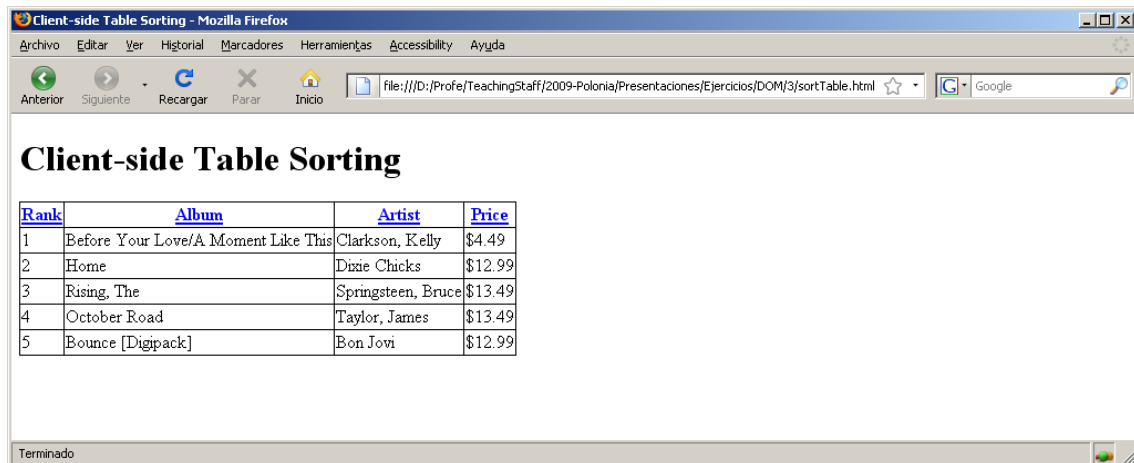
1. Create new row → `document.createElement("tr")`
2. Create new cells → `document.createElement("td")`
3. Create new text nodes →
`cell1.appendChild(document.createTextNode(content))`
4. Append cells to the row → `row.appendChild(cell)`
5. Get the tbody → `document.getElementById(id)`

6. Append row to the tbody

DOM 1 HTML functions

1. Get the tbody → `document.getElementById(id)`
2. Add a new row at the end → `tableTbody.insertRow(tableTbody.rows.length);`
3. Insert a new cell in the row → `row.insertCell(position)`
4. Append content to the cell →
`cell1.appendChild(document.createTextNode(content))`

3. Sort a table of data:



TIPS:

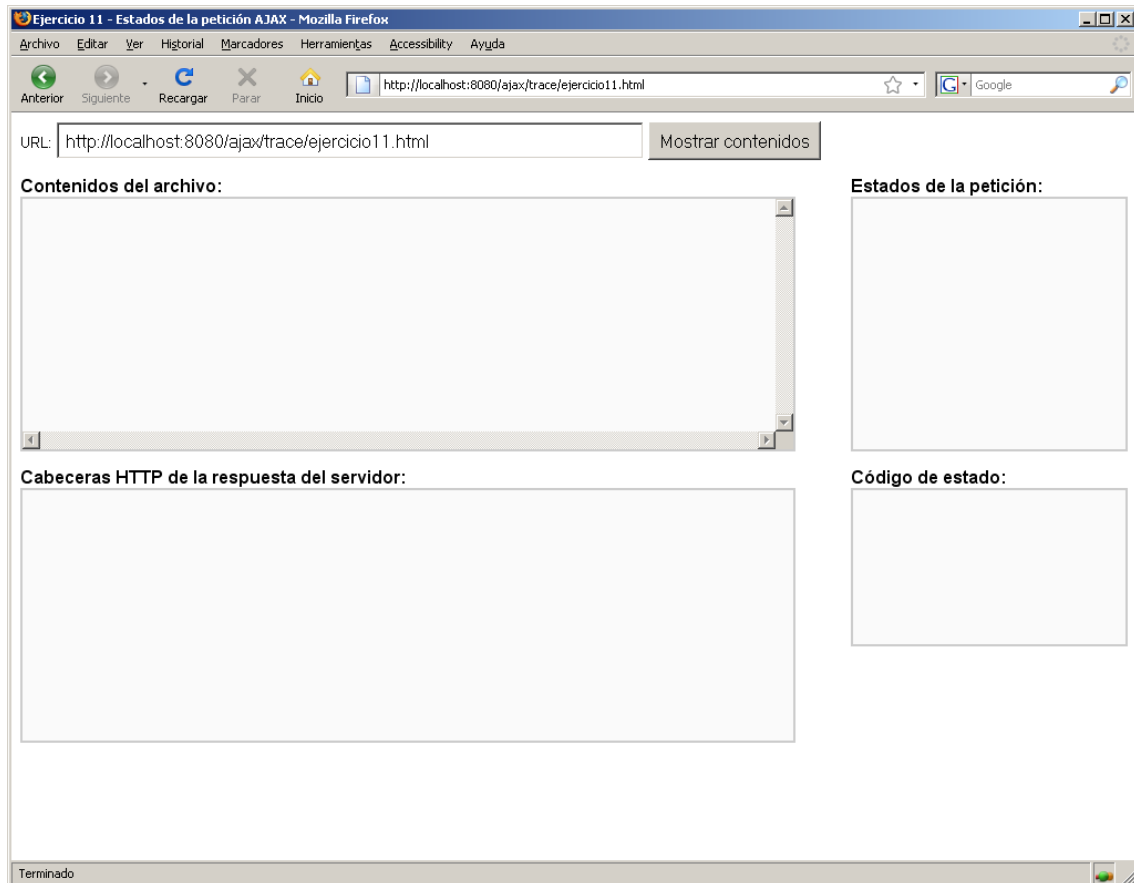
1. Get the table → `getElementById()`
2. Get the tbody → `getElementsByName()`
3. Get the rows → `getElementsByName()`
4. Insert the rows in an array, use `cloneNode()`
5. Sort the array with `sort()` method of array
6. Reverse order: use `reverse()` method of array
7. Create a new tbody → `document.createElement()`
8. Append ordered rows → `appendChild()`
9. Replace old tbody → `replaceChild()`

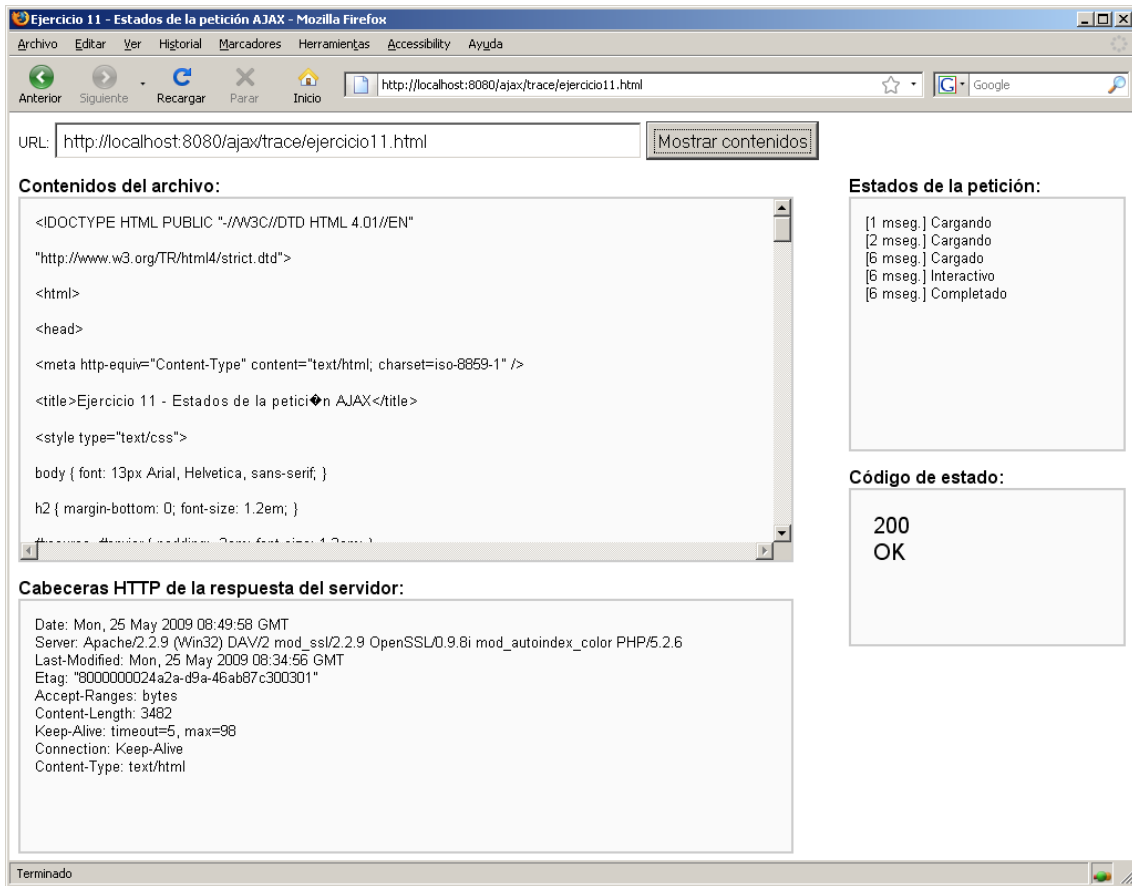
AJAX

1. AJAX tracer:

Create a web page that traces the AJAX communication:

1. User can input the URL of a document
2. When the document is loaded, it must be shown in a “Content” box
3. In a “HTTP headers” box, the headers must be shown
4. In a “State” box, the different states of the communication and the elapsed time must be shown
5. In a “State code” box,



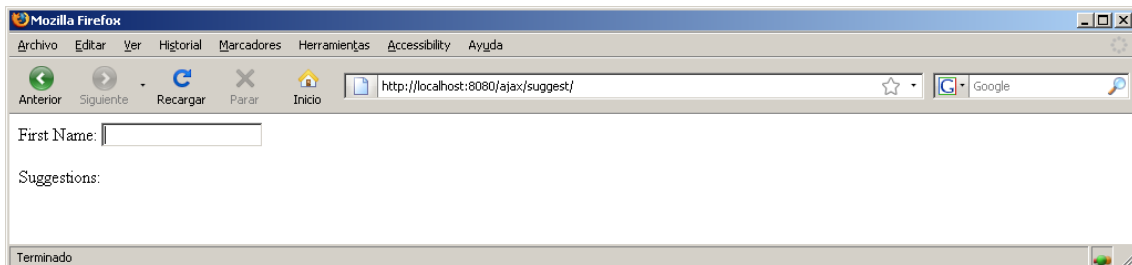


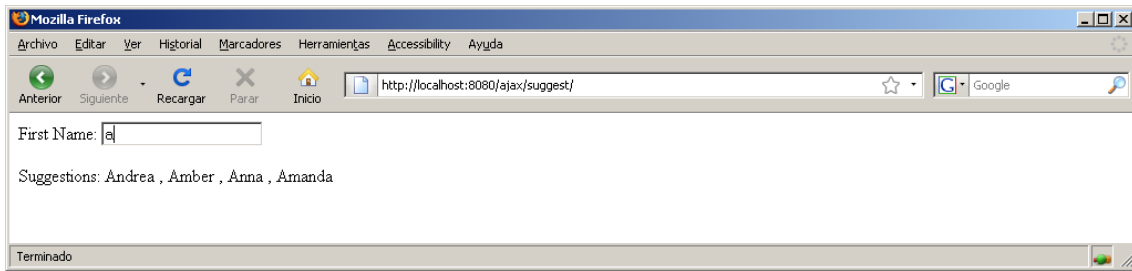
TIPS:

1. Use new Date() to calculate the elapsed time
2. Use getAllResponseHeaders() to show HTTP headers
3. Use status and statusText

2. Suggestion:

Create a web page that shows suggestions of names while user is typing the name.





TIPS:

1. Use onkeyup event to detect user's typing

3. News ticker:

Create a web page with a news ticker:

1. Every second, send a request for a piece of news and show it in the ticker.
2. Besides, show the time of the response.
3. When "Stop" button is pressed, the requests for more news are stopped. Then, the label of this button is changed to "Start"

The buttons "Previous" and "Next" stop the requests for more news. Besides, these buttons allow showing the previous or next news to the current piece of news that is shown.



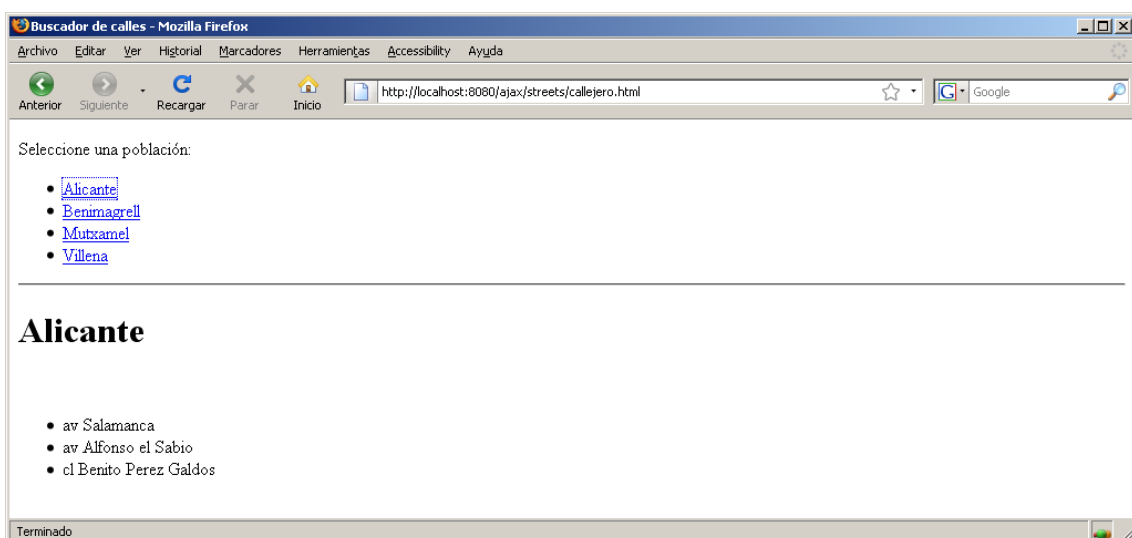
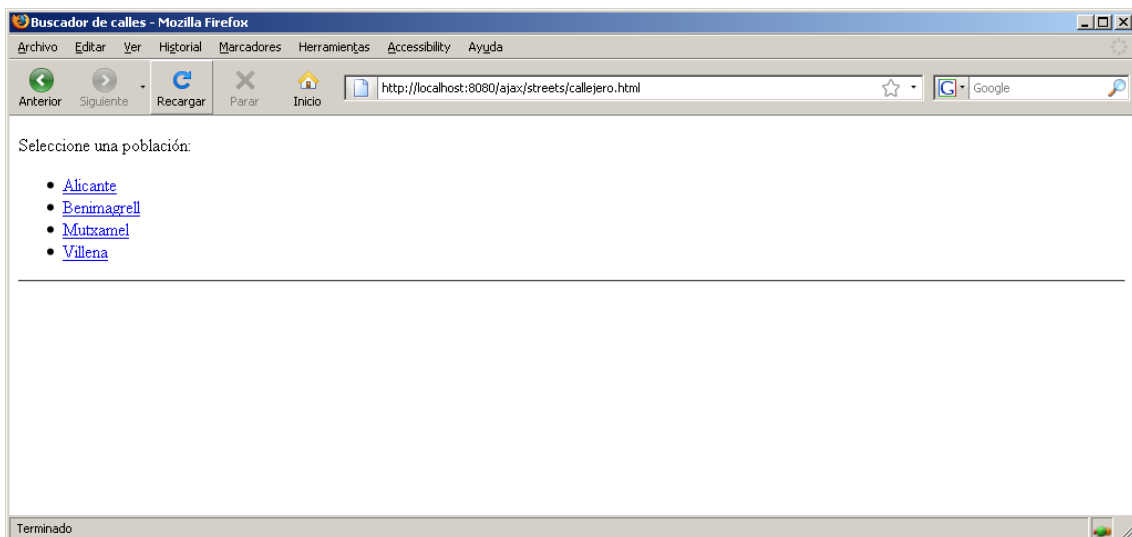
TIPS:

1. Use setInterval() to set a periodical event and clearInterval() to stop it
2. Use new Date() and functions getHours(), getMinutes(), and getSeconds()
3. Use an array and function push() to store the news

4. Street search:

Create a web page to search for the type and the name of streets in a city. The result must be an XML document similar to the following example:

```
<streets>
<city>Alicante</city>
<street type="av">Salamanca</street>
<street type="av">Alfonso el Sabio</street>
<street type="cl">Benito Perez Galdos</street>
</streets>
```



TIPS:

1. In PHP, use `header("Content-Type: application/xml");`

2. In JavaScript, responseXML contains an XML document → Use `getElementsByTagName()`, `firstChild`, `nodeValue`, and `getAttribute()`
3. In JavaScript, use `innerHTML` to quickly write HTML content