

ALUMNO: Ignacio Ortega Salinas

Asignatura: Programación de Sistemas Distribuidos

Curso: 2021/2022
Semestre: 2º

Fecha: 08-02-2022

Ejercicio práctico clase:

Tomando el fichero que tenemos en el campus sockets_alumno.html

- 1. Ábrelo con el navegador en tú ordenador. ¿Qué aparece en pantalla? ¿Y en consola?**

Aparece “WebSocket Test” escrito como título en la pantalla.

En la consola aparece un erro que nos indica que no se conecta al socket porque la dirección url no existe.

- 2. ¿Qué deberías hacer cambiar el output de la aplicación en el estado anterior?**

Cambiar el enlace del socket en la variable wsUri a un socket que funcione.

```
var wsUri =  
"wss://demo.piesocket.com/v3/channel_1?api_key=oCdCMcMPQpbvNjUIzqtvF1d2X2okWpDQj4AwARJu  
AgtjhzKxVEjQU6IdCjwm&notify_selfws://nebrija.sockets.com/v2/test";
```

- 3. ¿Para qué sirve window.addEventListener?**

Sirve para que la aplicación espere, en el momento en el que cargue, llamará a la función init.

- 4. Tomando como referencia lo que tenemos en la función onMessage ¿Cómo completarías la función onError(evt)? ¿Debes realizar la llamada a esta función en otra parte de la aplicación?**

Añadiendo un log en color rojo (por ejemplo) con el mensaje Error, para darte cuenta de que algo está fallando. Y si, habría que hacer la llamada en la función testWebSocket().

```
function onError(evt)  
{  
    writeLog('<span style="color: red;">ERROR! </span>');
```



```
}
```

```
function testWebSocket()
{
    websocket = new WebSocket(wsUri);
    websocket.onopen = function(evt) { onOpen(evt) };
    websocket.onmessage = function(evt) { onMessage(evt) };
    websocket.onerror = function(evt) { onError(evt)};
    websocket.onclose = function(evt) { onClose(evt)};
}
```

5. ¿Cómo ultimarías la función onClose(evt)?

Creando un Log el cual te avise de que te estas desconectando y mandar un mensaje de despedida al igual que se mandaba un mensaje de bienvenida.

```
function onClose(evt)
{
    writeLog("DISCONNECTED");
    sendMessage("Chao");
}
```

6. Investiga en internet sobre XML y busca su sintaxis y relación con los sistemas distribuidos actuales.

XML es un lenguaje similar a HTML con ciertas características que los diferencian, lo que más caracteriza a XML es su uso basado en etiquetas con herencias, las cuales su uso es obligatorio y unas puedes gestionar las herencias, cada etiqueta “padre” tiene un id, lo que facilita al programador el enterarse sobre que etiqueta está trabajando en cada momento y como encontrarlas.

Se relaciona con sistemas distribuidos con respecto a que XML es el sistema principal por el que se comparte información entre distintos dispositivos.