

 無料電子ブック

学習

WebSockets

Free unaffiliated eBook created from
Stack Overflow contributors.

#websocket

.....	1
1: WebSocket	2
.....	2
Examples.....	2
.....	2
web-socket.....	2
2: WebSocketAJAX	4
.....	4
Examples.....	4
.....	4
.....	6

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [websockets](#)

It is an unofficial and free WebSockets ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official WebSockets.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

1: WebSocketをいめる

このセクションでは、websocketのと、なぜがそれをいたいのかをします。

websocketのきなテーマについてもし、するトピックにリンクするがあります。 websocketのドキュメンテーションはしいので、それらのトピックのバージョンをするがあります。

Examples

インストールまたはセットアップ

ウェブソケットのセットアップまたはインストールにするな。

web-socket クライアントの

これはjavascriptのWebソケットクライアントのです。
それ

1. ライブデモサーバーにします。
2. メッセージをします。
3. メッセージをします。
4. にします。

```
var mySocket      = null;
var serverUrl     = 'wss://echo.websocket.org'; // wss: is ws: but using SSL.
var oWebSocket   = window.WebSocket || window.MozWebSocket;
if (oWebSocket) {
  mySocket = new oWebSocket (serverUrl);
  if (mySocket) {
    console.log (mySocket);
    mySocket.onopen      = onSocketOpen;
    mySocket.onclose     = onSocketClose;
    mySocket.onmessage   = onSocketMessage;
    mySocket.onerror     = onSocketError;

    setTimeout (closeSocket, 5000); // Be polite and free socket when done.
  }
}

function onSocketOpen (evt) {
  console.log ("Socket is now open.");
  mySocket.send ("Hello from my first live web socket!");
}

function onSocketClose (evt) {
  console.log ("Socket is now closed.");
}

function onSocketMessage (evt) {
  console.log ("Recieved from socket: ", evt.data);
}
```

```
function onSocketError (evt) {
  console.log ("Error with/from socket!");
  console.log (evt);
}

function closeSocket () {
  if (mySocket.readyState !== mySocket.CLOSED) {
    console.log ("Closing socket from our end (timer).");
    mySocket.close ();
  }
  else
    console.log ("Socket was already closed (timer).");
}
```

コンソールでそのコードをすると、のようにながられます。

```
WebSocket { url: "wss://echo.websocket.org/", readyState: 0, bufferedAmount: 0, onopen: null,
  onerror: null, onclose: null, extensions: "", protocol: "", onmessage: null,
  binaryType: "blob"
}
2 // <--- 2 was the socket ID in this instance.
Socket is now open.
Recieved from socket: Hello from my first live web socket!
Closing socket from our end (timer).
Socket is now closed.
```

オンラインでWebSocketをいめるをむ <https://riptutorial.com/ja/websocket/topic/7958/websocket>
をいめる

2: WebSocketとAJAXのい

き

AjaxはHTTPプロトコルをし、POST / GETメソッドをしてクライアントからサーバーにをできます。

WebSocketは、HTTPとはなり、クライアントとサーバーのするためのプロトコルです。

Ajaxでは、リクエストをすると、サーバーはそのリクエストにするをし、はします。

サーバーとのをするときにWebSocketをすると、クライアントとサーバーのでなだけをし、をできます。

Examples

をる

Ajax

1. しいリクエスト

- クライアントからのサーバー、しいがありますか
- サーバーからをするクライアント、NO
-

2. しいリクエスト

- クライアントからのサーバー、しいがありますか
- サーバーからをするクライアント、NO
-

3. しいリクエスト

- クライアントからのサーバー、しいがありますか
- サーバーからをするクライアント、NO
-

4. しいリクエスト

- クライアントからのサーバー、しいがありますか
- サーバーはをしますクライアント、はいここにあります
-

あなたはなのがあることがわかります

ウェブソケット

- がされました
- クライアントはうサーバー、しいがありますか

XX

- サーバーはうはいはっている

XX

- サーバーはうはいはっている
- クライアントはうがっている、サーバーは、しいメールをする

XX

- サーバーはうあなたはしいメールをとっている

Webソケットをすると、ながないことはわかです

オンラインでWebSocketとAJAXのいをむ

<https://riptutorial.com/ja/websocket/topic/8139/websocketとajaxのい>

クレジット

S. No		Contributors
1	WebSocketをいめる	Brock Adams , Community , Farhad
2	WebSocketとAJAX のい	Farhad , jalanb