

Node + Express + Socket.io = chat

Лаба: (β-версия) <https://docs.google.com/document/d/102Ngda6L-BzI65N1ArcHcvh2RgsqFgBLh8aTG1dMyRY/edit>

Архив с первым шагом: <https://github.com/gossoudarev/webteach/blob/master/express-chat/chat-express-1st-step.zip>
<http://kodaktor.ru/lr/node/chat-express-1st-step.zip>

A *socket* is one endpoint of a two-way communication link between two programs running on the network. A socket is bound to a port number so that the TCP layer can identify the application that data is destined to be sent to.

Сокет – это конечная точка двусторонней связи между двумя программами, работающими в сети. Сокет привязан к номеру порта, так чтобы слой TCP мог идентифицировать приложение, для которого предназначены данные.

С точки зрения программы сокет – это окошко, через которое виден другой участник взаимодействия. То есть интерфейс к нему.

Событие 'connection' возбуждается когда создаётся очередной сокет. В его коллбэк передаётся информация о сокете - объект. Внутри можно использовать этот объект уже для отслеживания этого конкретного соединения, в том числе отсоединения.

```
io.on('connection', function(socket){
  console.log('a user connected!');
  socket.on('disconnect', function(){
    console.log('a user disconnected!');
  });
});
```

Для того, чтобы связь создалась, и это событие возбудилось, нужно, чтобы обратился веб-клиент по соответствующему протоколу. В терминах node/express это означает, что у нас должен быть какой-то маршрут типа /chat/, обращение к которому интерпретируется как обращение веб-клиента чата.

Что такое веб-клиент чата? Это веб-страница, содержащая интерфейс чата и сценарий для взаимодействия с сервером. С ней взаимодействует человеческий клиент чата.

В простейшем виде выглядит так:

```
var t1 = '<script src="/socket.io/socket.io.js"></script><h1>Chat web-client!</h1>',
    t2 = '<script>var socket = io();</script>';
app.get('/chat', function(req, res){
  res.send(t1+t2);
});
```

Адрес /socket.io/socket.io.js автоматически генерируется серверной стороной.

выделенная красным строка инициализирует взаимодействие, т.е. посылает запрос, который возбуждает событие 'connection'.

Но в действительности, конечно, эти строки располагаются в отдельной статической веб-странице.

Далее, мы видим, что и в клиентском, и в серверном сценарии после разных действий появляется переменная с именем socket. На обеих сторонах там содержатся разные по факту объекты, но они оба – сокеты. И функциональность у них одинаковая:

