Walk-through 002

CREATE AND RUN A WEB SERVER

Objectives:

- create a web-server
- have it listen to a given port
- receive a request
- send back data

Steps:

1. create another web module / new folder under your project and call it '01_helloServer'
2. For better code hints, you'll need to configure Node to appear in your external libraries open 'preferences' and in the '**Node.js and NPM**' tab, click the 'configure' button…
3. Create a new file and call it **server.js**
4. For convenience - instead of typing console.log (…) every time, we can shorten it by adding the following function:

   (**d** stands for **debug**, but you can call it whatever you like...)

```
function d(){
    console.log.apply(this,arguments);
}
```

5. write the following code on top

```javascript
//import the built in http module
var http = require('http');

//define some variables
var host = 'localhost';
var port = '3000';

//create a web server
var server = http.createServer(function(req,res){
        //log the request url
        d('recieved request: ',req.url);

        //write a simple string response to the browser
        res.writeHead(200,{'Content-type':'text/plain'});
        res.end('hello server');
})

//have the server listen to a given port
server.listen(port,host,function(){
    d('server is listening on ',host,' | ',port);
});
```

6. Create a new Run/Debug configuration pointing at server.js

   from the top menu - Run > Edit configuration > **+** > Node.js

7. Run the app from the IDE - you will see the **listening** prompt in the console.

8. Open a browser and navigate to **localhost:3000** and hit enter

   you will see the URL response in the console and the message sent back to the response in the browser.

9. Notice that the same message will appear in the browser if you navigate to a different route like **localhost:3000/users** but the server console prints out the difference in URL.

   This is because right now we are sending the same response back no matter what is the URL.

10. Change the text returned to the browser via res.end()

11. Refresh the browser - notice the text in the browser was not changed - you'll have to restart the server manually to see the changes you've made to the code…

12. In your IDE - stop the server and run it again, then refresh the browser to see the changes.

- Be sure to go through the full http / https reference here -

  http://nodejs.org/api/http.html

  http://nodejs.org/api/https.html