

# Exercises: JSON-RPC

When you use JSON-RPC, make a separate Dynamic Web Project from the one you used for the previous exercises. Grab the JAR files out of the WEB-INF/lib folder of my “json-rpc” project and drop them in the WEB-INF/lib folder of your new project. Also, grab jsonrpc.js out of the scripts folder of my project and drop it into yours. If you want, you can also cut and paste my web.xml entries (servlet-mapping for JSON-RPC and listener declaration for the initializer) and copy my JsonRpcInitializer class.

1. Make a pushbutton that when pressed, generates five random numbers in JavaScript using `Math.random()` and sends them to the server. The server should add them together and send back the sum. The client should look at the sum, and if it is less than 2.5, show a message saying “you are a loser”. Otherwise it should show a message saying “you are a winner”. Use a synchronous RPC call.
2. Repeat the process, but use asynchronous calls.
3. Redo any/all of the exercises from the previous exercises, but use JSON-RPC instead of explicitly using `JSONObject` and `JSONArray`. Please note that if the server-side method returns a `List`, what arrives on the client is not a JavaScript array. Instead, it is a JSON object with a “list” property that is a JavaScript array. So, you would do something like this:

```
function customerTable(inputField, resultRegion) {
    var idList = getRawValue(inputField);
    var callback = function(customerList, exception) {
        if(exception) {
            alert(exception.message);
        } else {
            var customers = customerList.list;
            var headings = ["ID", "First Name", "Last Name", "Balance"];
            ...
        }
        ...
    }
}
```

4. Make a form that collects a customer ID, a first name, a last name, and a balance. When the user presses the pushbutton, send the info to the server. If the customer ID does not already exist, create a `Customer` on the server that has the given information, and display a confirmation message similar to “Customer successfully created”. If the customer ID exists already, display an error message. The customer need only persist on the server until the next time the server is restarted.