3.Create Express.JS application to Read request and return response by using Client Server model

Server-side code:

```
const express = require('express');


const app = express();


app.get('/', (req, res) => {
  res.send('Hello World');
});


app.listen(3000, () => {
  console.log('Server listening on port 3000');
});
```

This code creates an Express.js app and sets up a route for the root path ('/') using the app.get() method. When a request is received at this path, the callback function sends a response with the message "Hello World". The app listens for requests on port 3000.

Client-side code:

```
const http = require('http');


const options = {
  hostname: 'localhost',
  port: 3000,
  path: '/'
};


const req = http.request(options, res => {
  console.log(`statusCode: ${res.statusCode}`);
```

```
  res.on('data', d => {
    process.stdout.write(d);
  });
});

req.on('error', error => {
  console.error(error);
});

req.end();
```

This code creates an HTTP request using the http module, and sends it to the server using the options object that specifies the hostname, port, and path. When a response is received, the status code is logged to the console, and the response data is written to stdout.