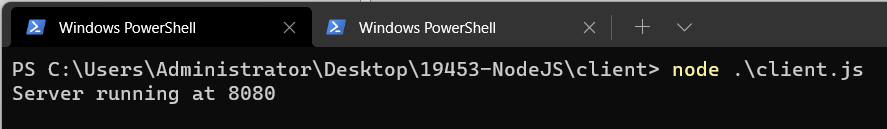
**Lab Assignment - 2 (NodeJS)**

*19CSE445 – Cloud Computing*

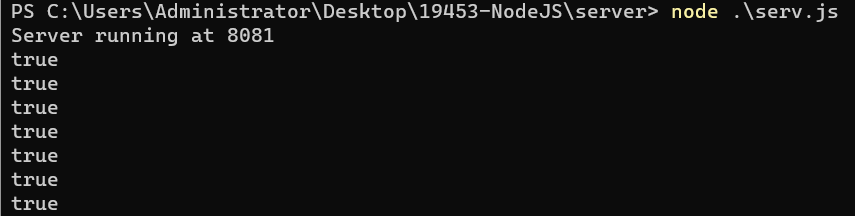
**HTML with CSS:**



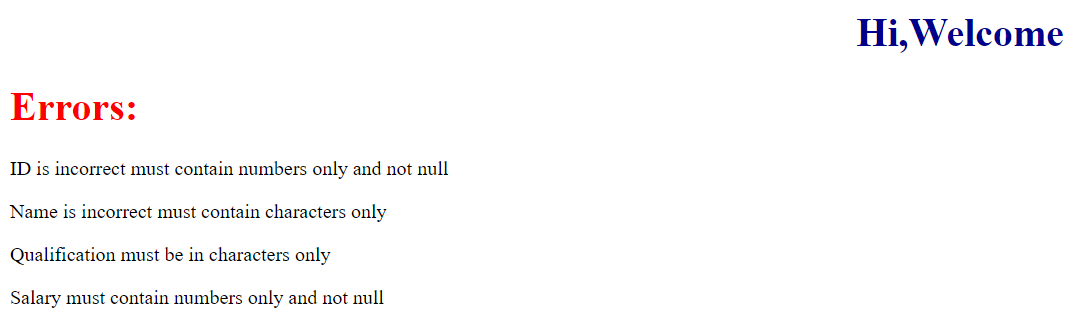
**Loading HTML through NodeJS: (Client Side)**

* 
* 

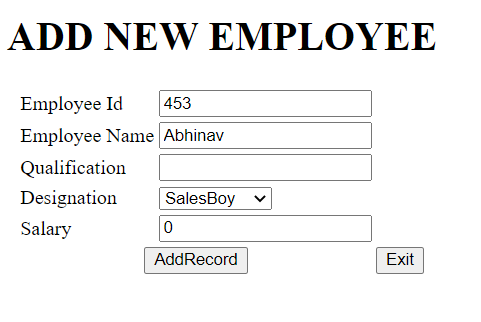
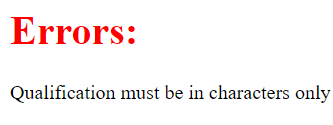
**Validation JS file (Server side):**

* 

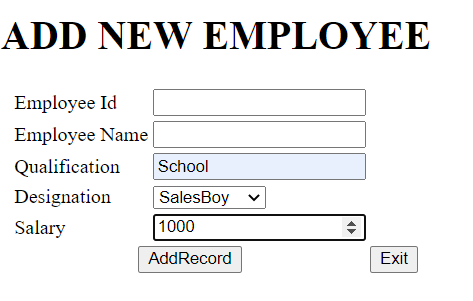
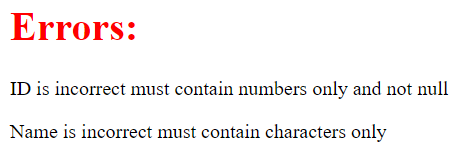
**If form is empty, validation page shows:**



**Case(i): Qualification not entered**

* 
* 

**Case(ii): Id and Name not entered**

* 
* 

**HTML:**

<!DOCTYPE html>  
<html>  
 <head>  
 <title>Abhinav</title>  
 <link href="style.css" rel="stylesheet">  
 </head>  
 <body>  
 <div class="container" align="center"><br>  
 <form action="http://localhost:8081" method="GET">  
 <h1>ADD NEW EMPLOYEE</h1>  
 <table>  
 <tr>  
 <td class="tdLabel"><label for="number" class="label">Employee Id</label></td>  
 <td class="d1"><input type="number" name="id" value="" id="id"/></td>  
 </tr>  
 <tr>  
 <td class="tdLabel"><label for="text" class="label">Employee Name</label></td>  
 <td><input type="text" name="name" id="name"/></td>  
 </tr>  
 <tr>  
 <td class="tdLabel"><label for="register\_email" class="label">Qualification</label></td>  
 <td class="d1"><input type="text" name="qual" value="" id="qual"/></td>  
 </tr>  
 <tr>  
 <td class="tdLabel"><label for="register\_country" class="label">Designation</label></td>  
 <td><select name="country" id="des">  
 <option value="i">SalesBoy</option>  
 <option value="p">Marketing</option>  
 <option value="a">Accountant</option>  
 <option value="c">Audit</option>  
 <option value="other">other</option>  
 </select>  
 </td>  
 </tr>  
 <tr>  
 <td class="tdLabel"><label for="register\_name" class="label">Salary</label></td>  
 <td><input type="number" name="number" value="" id="sal"/></td>  
 </tr>  
 <tr>  
 <td colspan="2"><div align="center"><input type="submit" id="submit" value="AddRecord"  
 class="btn btn-success" autocomplete="off" onClick="create\_account()"/>  
 </div></td>  
 <td colspan="2"><div align="left"><input type="reset" id="reset" value="Exit"/>  
 </tr>  
   
 </table>  
 </form>  
 <br><br>  
 </div>  
 </body>  
</html>

**CSS:**

body {  
 font-family: "Open Sans", sans-serif;  
 color: #272829;  
}  
  
a {  
 color: #149ddd;  
 text-decoration: none;  
}  
  
a:hover {  
 color: #37b3ed;  
 text-decoration: none;  
}  
  
h1,  
h2,  
h3,  
h4,  
h5,  
h6 {  
 font-family: "Lucida Handwriting", sans-serif;  
 font-size: 35px;  
 color: blue;  
}  
  
.container {  
 border-radius: 5px;  
  
 background-color: peachpuff;  
 width: 400px;  
 height: 330px;  
 border: 2px solid black;  
 padding: 50px;  
 position: absolute;  
 left: 27%;  
}  
  
.td {  
 width: 160px;  
 padding: 50px;  
}  
  
.tdLabel {  
 font-size: 18px;  
 color: rgba(255, 0, 0, 0.671);  
}  
  
.d1 {  
 font-weight: italic;  
}

**Client.js (Accessing html file thorugh server):**

var http = require("http");  
var path = require("path");  
var fs = require('fs');  
  
http.createServer(function (req, res) {  
 fs.readFile(\_\_dirname + '/index.html', function (err,data) {  
 if (err) {  
 res.writeHead(404);  
 res.end(***JSON***.stringify(err));  
 return;  
 }  
 res.writeHead(200);  
 res.end(data);  
 });   
}).listen(8080);  
  
// Console will print the message  
***console***.log('Server running at 8080');

**Server.js (Validation of added record):**

var http = require("http");  
var gow = require("url");  
  
http  
 .createServer(function (request, response) {  
 response.writeHead(200, { "Content-Type": "text/html" });  
 var q = gow.parse(request.url, true).query;  
 ***console***.log("true");  
 let id = q.id;  
 let empName = q.name;  
 let qualification = q.qual;  
 var numbers = /^[0**-**9]\*$/;  
 var letters = /^[a**-**zA**-**Z]\*$/;  
 var email\_val = /^([a**-**zA**-**Z0**-**9\_\.\-])+\@(([a**-**zA**-**Z0**-**9\-])+\.)+([a**-**zA**-**Z0**-**9]{2,4})+$/;  
 var txt = "Welcome " + q.name;  
 var dummy = "Errors: ";  
 response.write('<h1 style="color:darkblue" align="center">Hi,' + txt+'</h1>');  
 response.write('<h1 style="color:red"> \t' + dummy+'</h1>');  
 if (!(id !== "" && id !== undefined && id.match(numbers))) {  
 response.write(  
 "<p>ID is incorrect must contain numbers only and not null</p>"  
 );  
 }  
 if (  
 !(empName !== "" && empName !== undefined && empName.match(letters))  
 ) {  
 response.write("<p>Name is incorrect must contain characters only</p>");  
 }  
 if (  
 !(qualification !== "" && qualification !== undefined && qualification.match(letters))  
 ) {  
 response.write("<p>Qualification must be in characters only</p>");  
 }  
 if (!(id !== "" && id !== undefined && id.match(numbers))) {  
 response.write(  
 "<p>Salary must contain numbers only and not null</p>"  
 );  
 }  
 response.end();  
 })  
 .listen(8081);  
// Console will print the message  
***console***.log('Server running at 8081');