

Practical Assignment – Web Designing – 2(405)

AJAX programs list

1. Write JavaScript program that uses AJAX to read JSON file and display its contents.

```
<html lang="en">
<head>
  <title>Document</title>
  <script src="jquery-3.6.0.min.js"></script>
</head>
<body>
  <h2 id="demo">hello</h2>
  <button onclick="readtxt()">click me</button>
  <script>
function readtxt()
{
  var xhr = new XMLHttpRequest();
  xhr.open("GET","cars.json",true);
  xhr.onreadystatechange = function()
  {
    if (this.readyState == 4 && this.status == 200)
    {
      res = JSON.parse(this.responseText);
      alert(res);
      var t = "<table border=3> <th> Cars </thead>";
      for(var i in res)
      {
        t += "<tr><td>" + res[i] + "</td></tr>"
      }
      t += "</table>";
      document.getElementById("demo").innerHTML = t;
    }
  };
  xhr.send();
}
</script>
</body> </html>
-----cars.json-----
["Ford", "BMW", "Fiat", "Kia", "Hyundai", "Tata"]
```

Practical Assignment – Web Designing – 2(405)

2. Write JavaScript program that uses AJAX to read XML file and display its contents.

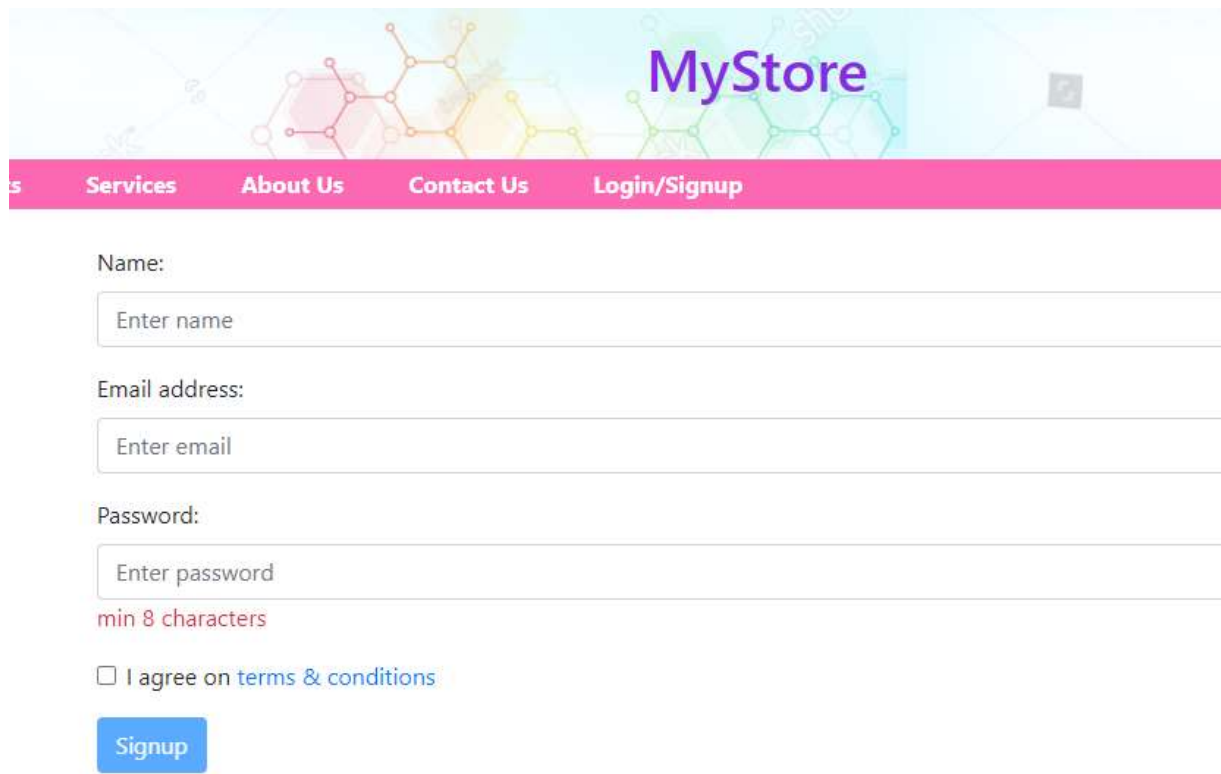
```
<html lang="en">
<head>
  <title>Document</title>
  <script src="jquery-3.6.0.min.js"></script>
  <script>
    function readtxt()
    {
      var xhr = new XMLHttpRequest();
      xhr.open("GET", "student.xml",true);
      xhr.onreadystatechange = function()
      {
        alert("readystate = " + xhr.readyState);
        if (this.readyState == 4 && this.status == 200)
        {
          alert(this.responseText);
          document.getElementById("demo").innerHTML = this.responseText;
        }
      };
      xhr.send();
    }
  </script>
</head>
<body>
  <h1 id="demo">hello</h1>
  <button onclick="readtxt()">click me</button>
</body>
</html>
```

```
-----student.xml-----
<students>
  <student>
    <rno> 101 </rno>
    <name> Aman </name>
  </student>
  <student>
    <rno> 102 </rno>
    <name> Priyank </name>
  </student>
```

Practical Assignment – Web Designing – 2(405)

jQuery programs list

1. Create a registration form as shown in following figure and use jQuery to
 - i) enable / disable the Signup button on selection of checkbox of “I agree on terms and conditions”.
 - ii) display number of characters left for “password” field.
 - iii) on focus and on blur, change the color of all three input boxes.



The screenshot shows a web page for 'MyStore' with a pink navigation bar containing links: 'Services', 'About Us', 'Contact Us', and 'Login/Signup'. Below the navigation bar is a registration form. The form includes three input fields: 'Name:', 'Email address:', and 'Password:'. The 'Password:' field has a red text label 'min 8 characters' below it. There is a checkbox labeled 'I agree on terms & conditions' and a blue 'Signup' button.

```
<html lang="en">
<head>
  <title>MyStore - Register</title>
  <link href="css/style.css" rel="stylesheet">
  <link rel="stylesheet" href="css/bootstrap.min.css">
  <script src="jquery-3.6.0.min.js"></script>
  <script type="text/javascript">
    $(document).ready(function(){
      var maxLength = 8;
      $('#pwd').keyup(function() {
        var textlen = maxLength - $(this).val().length;
```

Practical Assignment – Web Designing – 2(405)

```
        $('#rchars').text(textlen + ' character(s) remaining');
    });
});
</script>
</head>
<body>
    <header style="background-image: url('banner.PNG');">
        <h1> MyStore</h1>
    </header>
    <div class="row">
        <div class="col-sm-2">

</div>
        <div class="col-sm-8">
            <form action="/action_page.php">
                <div class="form-group">
                    <label for="email">Name:</label>
                    <input type="text" class="form-control" placeholder="Enter name" id="name">
                </div>
                <div class="form-group">
                    <label for="email">Email address:</label>
                    <input type="email" class="form-control" placeholder="Enter email" id="email">
                </div>
                <div class="form-group">
                    <label for="pwd">Password:</label>
                    <input type="password" class="form-control" placeholder="Enter password" id="pwd">
                    <span id="rchars" class="text-danger">min 8 characters</span>
                </div>
                <div class="form-group form-check">
                    <label class="form-check-label">
                        <input id="check1" class="form-check-input" type="checkbox"> I agree </label>
                    </div>
                    <button type="submit" class="btn btn-primary" disabled="disabled">Signup</button>
                </form>
            </div>
            <div class="col-sm-2">
        </div>
    </div>
```

Practical Assignment – Web Designing – 2(405)

```
<script type="text/javascript">
$(document).ready(function(){
  $("#check1").click(function(){
    if($("#button").prop('disabled') == true)
    {
      $("#button").prop('disabled',false);
    }
    else
    {
      $("#button").prop('disabled',true);
    }
  });
});</script> </body> </html>
```

2. Create a webpage that uses jQuery to display different animation effects with different HTML elements, on click of respective buttons.

Practical Assignment – Web Designing – 2(405)

3. Create a form as shown in below diagram. On Click of “Add Language” add a new language to the languages selection box.

Personal Information:

Full Name

Email ID

Contact No

Gender ☐ Male ☐ Female

Address

Languages [Add Language](#)

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Document</title>
  <script src="jquery.min.js"></script>
  <script>
    $(document).ready(function(){
      $("#addLang").click(function(){
        $("#lang").show();
      });
      $("#newLang").blur(function(){
        var newLang = $("#newLang").val(); alert('Hello'+newLang);
        $("#langList").append("<option>" + newLang + "</option>");
        $("#lang").hide();
      });
    });
  </script>
  <style>
    fieldset{ display:block; margin-left:200px; margin-right:10px; border:10px groove;
width:800px; color:#00CC00;
    }
  </style>
</head>
<body>
```

Practical Assignment – Web Designing – 2(405)

```
<form method="post" enctype="multipart/form-data">
  <fieldset>
    <table align="center" width="70%">
      <legend>Personal Information:</legend>
      <tr>
        <td width="314">Full Name</td>
        <td width="276"><input type="text" id="name"></td> </tr>
      <tr>
        <td>Email ID</td>
        <td><input type="email" id="email"></td> </tr>
      <tr>
        <td>Contact No</td>
        <td><input type="text"></td> </tr>
      <tr>
        <td>Gender</td>
        <td>
          <input type="radio" value="Male"> Male
          <input type="radio" value="Female"> Female
        </td> </tr>
      <tr>
        <td>Address</td>
        <td><textarea rows="5" cols="20"></textarea></td>
      </tr>
      <tr>
        <td>Languages</td>
        <td>
          <select id="langList">
            <option>Gujarati</option>
            <option>Hindi</option>
            <option>English</option>
          </select>
          <a id="addLang" href="#"> Add Language </a>
        </td>
      </tr>
      <tr style="display: none; id="lang">
        <td>Add new Language</td>
        <td><input type="text" id="newLang"></td>
      </tr>
```

Practical Assignment – Web Designing – 2(405)

```
<tr>
<td colspan="2" align="center">
<input type="submit" value="Save">
</td>
</tr>
</table>
</fieldset>
</form>
</body>
</html>
```


Practical Assignment – Web Designing – 2(405)

Node JS programs list

1. Write nodeJS program to split query string from url.

```
// splitting query string demo program
// browser url = localhost:8081/?rollno=101&name=Vaibhav
var http = require('http');
var url = require('url');
console.log("Server running at port 8081");
http.createServer(function (req, res) {
  res.writeHead(200, {'Content-Type': 'text/html'});
  var q = url.parse(req.url, true).query;
  var txt = q.rollno + " " + q.name;
  res.end(txt);
}).listen(8081);
```

Practical Assignment – Web Designing – 2(405)

2. Write nodeJS program to reads query string from url using req.url and sends appropriate response.

// this program simply reads query string from url using req.url and sends appropriate response.

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

```
var server = http.createServer(function (req, res) {
```

```
  if (req.url == '/')
```

```
  {
```

```
    res.writeHead(200, { 'Content-Type': 'text/html' });
```

```
    res.write('<html><body><h1>This is home Page.</h1></body></html>');
```

```
    res.end();
```

```
  }
```

```
  else if (req.url == "/student")
```

```
  {
```

```
    res.writeHead(200, { 'Content-Type': 'text/html' });
```

```
    res.write('<html><body><h1>This is student Page.</h1></body></html>');
```

```
    res.end();
```

```
  }
```

```
  else if (req.url == "/admin")
```

```
  {
```

```
    res.writeHead(200, { 'Content-Type': 'text/html' });
```

```
    res.write('<html><body><p>This is admin Page.</p></body></html>');
```

```
    res.end();
```

```
  }
```

```
  else
```

```
  {
```

```
    res.end('<h1> Invalid request </h1>');
```

```
  }).listen(8081);
```

```
console.log("NodeJS Server is running, listening at port: 8081");
```

Practical Assignment – Web Designing – 2(405)

3. write nodeJS program to read URL, parse it to JS object and display it's properties.

//write nodeJS program to read URL, parse it to JS object and display it's properties.

//localhost:8081/hello.html?rollno=123&name=Vishal

```
var http = require('http');
var url = require('url');

http.createServer(function(req,res){
    res.write("<h1> req.url= " + req.url + "</h1>");

    q = url.parse(req.url,true).query;
    res.write("<h1> your rollno is " + q.rollno + "</h1>");
    res.write("<h1> your name is " + q.name + "</h1>");

    h = url.parse(req.url,true).host;
    res.write("<h1> host= " + h + "</h1>");

    p = url.parse(req.url,true).pathname;
    res.write("<h1> pathname= " + p + "</h1>");

    s = url.parse(req.url,true).search;
    res.write("<h1> search= " + s + "</h1>");
    res.end();
}).listen(8081);
console.log("Node JS server running at 8081...");
```

Practical Assignment – Web Designing – 2(405)

4. Create user defined module in nodeJS that exports an object.

```
//this module exports an object – udm3.js  
exports.student = {rno:"101",name:"Utsav"};
```

5. Create user defined module in nodeJS that exports a function.

```
// this module exports a function - udm2.js  
exports.print = function(str){console.log(str)};  
-----prg1.js-----  
  
// this program requires - imports above modules  
var udm2 = require('./udm2.js');  
var udm3 = require('./udm3.js');  
udm2.print("this msg is from UDM-2");  
console.log("Now printing data from UDM-3");  
console.log("Roll No=",udm3.student.rno);  
console.log("Name=",udm3.student.name);
```

Practical Assignment – Web Designing – 2(405)

6. Write nodeJS program that

- i. creates one file
- ii. writes one line to the file
- iii. appends one line to the file
- iv. read the file
- v. delete the file

→

-----create a file and write some text -----

```
var fs = require('fs');
fs.open('file2.txt', 'w', function (err, file) {
  if (err)
  {
    console.log("Error");
    throw err;
  }
  fs.write("hello");
  console.log('Saved!');
});
```

----- read data from file -----

```
var http = require('http');
var fs = require('fs');
console.log("Server running at 8081...");
http.createServer(function (req, res) {
  console.log("Request received");
  fs.readFile('file1.html',function(err, data) {
    if(err)
      throw(err);
    else
    {
      res.writeHead(200, {'Content-Type': 'text/html'});
      res.write(data);
      return res.end();
    }
  });
}).listen(8081);
```

Practical Assignment – Web Designing – 2(405)

-----append one line to the file -----

```
fs.appendFile('new.txt', ' Written using append file ', function (err) {  
  if (err)  
  {  
    console.log("Error");  
    throw err;  
  }  
  console.log('Saved!');  
});
```

-----delete the file -----

```
fs.unlink('file2.txt', function (err) {  
  if (err)  
  {  
    console.log("Error") ;  
    throw err;  
  }  
  console.log('File deleted!');  
});
```