


☐

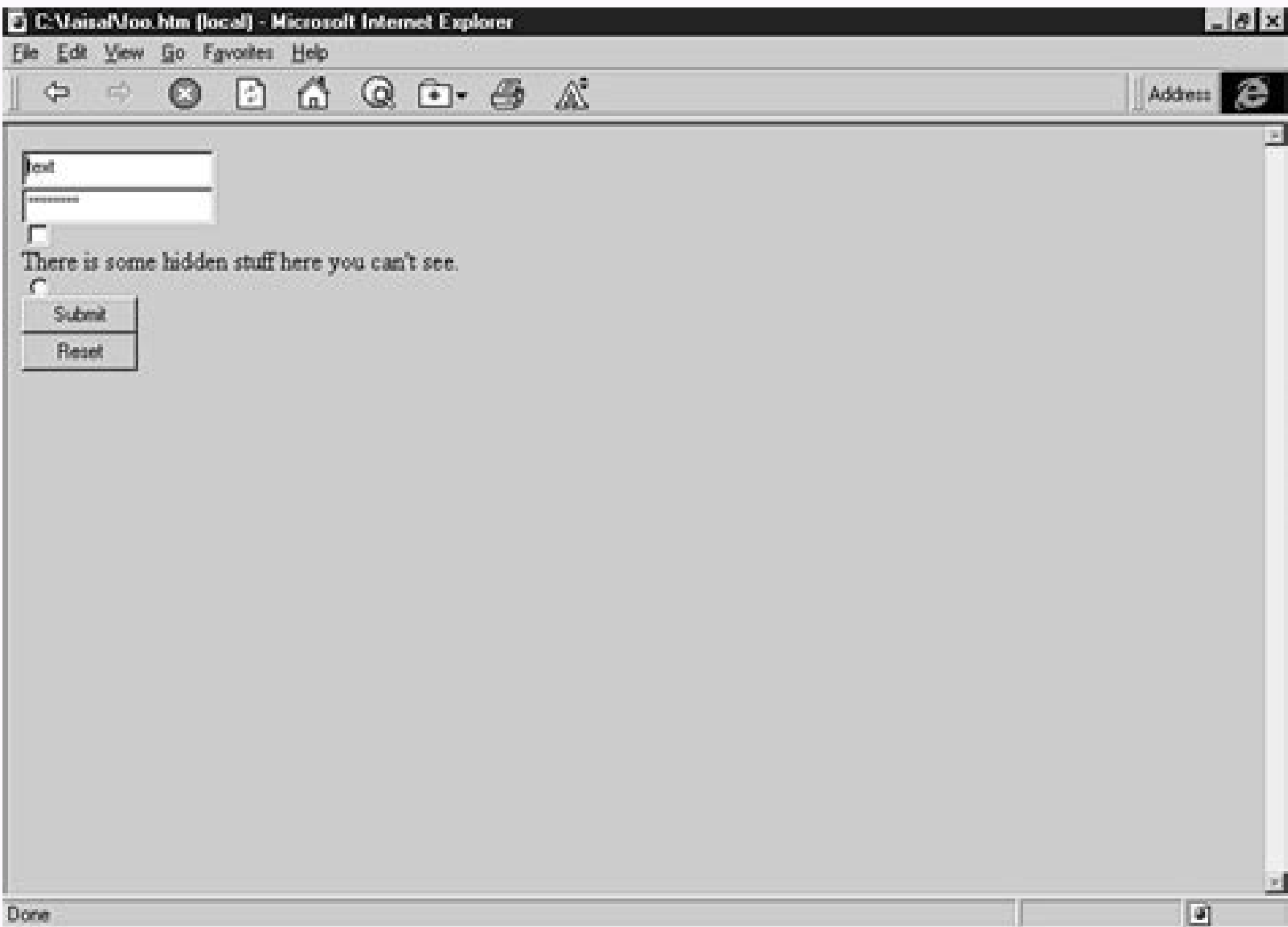
I'm not robot

  
reCAPTCHA

Continue

5662078.8571429 30273186888 178058935494 57143287368 73769667.75 61909905345 59328198300 29334609952 14555709.461538 36541146370 291704.60655738 30898814.289855 1229519766 30395709.181818 12081096.3125 16446252.540816 25764307794 24149360.425287 9920683090 11636240238 148181180300 60587306736 61807666404 65606453.16129 242211174.204545 6086350.325 14884655868 30996869888 20302840.263889 7371654.138889 706799144 74034403968

## Js formdata append multiple file



```
// Calling all the required packages
const express = require('express');
const bodyParser = require("body-parser");
const path = require('path');

const app = express();

// Configurations for "body-parser"
app.use(
  bodyParser.urlencoded({
    extended: true,
  })
);

// Configurations for "Static-files"
app.set("view engine", 'ejs')
app.set('views', path.join(__dirname, 'views'))
app.use(express.static(`${__dirname}/public`))

// API Endpoint to render HTML file
app.use('/', (req, res) => {
  res.status(200).render('index');
});

// Express server
module.exports = app
```

## Upload files to server using - require('formidable')

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


http.createServer(function (req, res) {
  if (req.method !== 'POST') return;
  var form = new formidable.IncomingForm();
  form.parse(req, function(err, fields, files) {
    var upload = files['fileupload'].path;
    console.log('Upload file: ' + upload);
    fs.rename(upload, path.resolve(__dirname, 'public', 'uploads', 'file_upload.png'), function (err) {
      if (err) throw err;
      res.writeHead(200, {'Content-type': 'text/html'});
      res.end('File uploaded and moved!');
    });
  });
})
// test it
var url = 'http://localhost:3000/';
var req = http.request(url, function(res) {
  res.on('data', function(data) {
    console.log('data: ' + data);
  });
  res.on('end', function() {
    console.log('end');
  });
});
req.end('fileupload=1');
```

Email:

Tel:

Password:

Password2:

 The password confirmation is different to the password value

```

data:
  _overheadLength: 286,
  _valueLength: 1,
  _valuesToMeasure:
    {
      readStream:
        {
          _readableState: [Object],
            readable: true,
            domain: null,
            _events: [Object],
            _eventsCount: 3,
            _maxListeners: undefined,
            path: 'c:\users\user\civicrm\wp-content\uploads\25500850-22e1.txt',
            fd: null,
            flags: 'r',
            mode: 438,
            start: undefined,
            end: undefined,
            autoClose: true,
            pos: undefined,
            bytesRead: 0,
            emit: [Function] } },
      writable: false,
      readable: true,
      dataSize: 0,
      metaDataSize: 209752,
      pauseStreams: true,
      _released: false,
      _streams:
        {
          .....551406988289297303420179\r\nContent-Disposition: form-data; name="pid"\r\n\r\n',
            "400",
            [Function: bound ] },
          .....551406988289297303420179\r\nContent-Disposition: form-data; name="deliveryAttachment";
            filename="civ\civicrm\wp-content\uploads\25500850-22e1.txt"\r\nContent-Type: text/plain\r\n\r\n',
            DelayedStream {
              source: [Object],
              dataSize: 0,
              metaDataSize: Infinity,
              pauseStreams: true,
              _metaDataSizeExceeded: false,
              _released: false,
              _bufferedEvents: [Object],
              _events: [Object],
              _eventsCount: 3 },
            [Function: bound ] },
          _currentStream: null,
          _boundary: .....551406988289297303420179" }
    }
  }
}

```

Js formdata append multiple files.

[illegible]



[illegible]