

## ➔ Image upload & update

### Index.js

```
let express = require('express');
let mongoose = require('mongoose');
//point 1--> multer import
let multer = require('multer');
let methodOverride = require('method-override')

let session = require('express-session');
let flash = require('connect-flash')

let app = express();
app.set('view engine', 'ejs')
app.use(methodOverride('_method'))
app.use(express.static(__dirname + '/public/'))

app.use(session({
  secret: 'nodejs',
  resave: true,
  saveUninitialized: true
}))
app.use(flash())

app.use((req, res, next) => {
  res.locals.sucess = req.flash('sucess')
  res.locals.err = req.flash('err')
  next()
})

mongoose.connect('mongodb://localhost:27017/testingdb')
let myschema = mongoose.Schema({
  single_image: String
})
let mymodel = mongoose.model('table', myschema)

//point 2--> multer sorage setting
let storage = multer.diskStorage({
  destination: 'public/images/',
  filename: (req, file, cb) => {
    //cb(null, Date.now() + file.originalname) // file name setting with current -date
    cb(null, file.originalname)
  }
})

let upload = multer({
```

```

storage: storage,
// here image validation
fileFilter: (req, file, cb) => {
  if (file.mimetype == "image/jpeg" || file.mimetype == "image/jpg" || file.mimetype ==
"image/png") {
    cb(null, true)
  }
  else {
    cb(null, false)
    return cb(new Error('Only Jpg, png, jpeg, allow'))
  }
}
}))

app.get('/', (req, res) => {
  mymodel.find({})
  .then((x)=>{
    res.render('index', {x})
  })
  .catch((y)=>{
    console.log(y)
  })
}))

app.get('/add-image', (req, res) => {
  res.render('image-upload-file')
}))

app.post('/add-image', upload.single('single_image'), (req, res) => {
  req.file.filename
  let data = {
    single_image: req.file.filename
  }
  mymodel.findOne(data)
  .then((x)=>{
    if(x){
      req.flash('sucess', 'this image hass been already uploaded, please try angother
image')
      res.redirect('/')
    }
    else{
      mymodel.create(data)
      .then((x) => {
        req.flash('sucess', 'Your Image uploaded Sucessfully on data base')
        res.redirect('/')
      }).catch((y) => {
        console.log(y)
      })
    }
  })
}))

```

```

}))

app.get('/edit/:id', (req, res)=>{
  mymodel.findOne({single_image:req.params.id})
    .then((x)=>{
      res.render('edit-image-file', {x})
    })
    .catch((y)=>{
      console.log(y)
    })
})

app.put('/edit/:id', upload.single('single_image'),(req, res)=>{
  mymodel.updateOne({single_image:req.params.id},{ $set:{
    single_image:req.file.filename
  }} )
    .then((a)=>{
      req.flash('sucess', 'Your Image has been sucessfully edited')
      res.redirect('/')
    })
    .catch((b)=>{
      console.log(b)
    })
})

app.listen(200, () => {
  console.log('200 Port Working')
})

```

## image-upload-file.ejs

```

<form method="post" action="/add-image" enctype="multipart/form-data">
  <input type="file" class="form-control" name="single_image"><br><br>
  <button>Submit</button>
</form>

```

## edit-image-file.ejs

```

<form method="post" action="/add-image" enctype="multipart/form-data">
  <input type="file" class="form-control" name="single_image"><br><br>
  <button>Submit</button>
</form>

```

## index.ejs

```
<%if(sucess && sucess != ''){%>
  <div class="alert alert-success" role="alert">      <%= sucess%>    </div>
<%}else if(err && err != ''){%>
  <div class="alert alert-danger" role="alert">      <%= err%>      <%}%>

<a href='/add-image'>Add or Upload Images</a><br><br>

<%x.forEach((y)=>{%>
  <style>
    ul{
      display:flex;
      flex-wrap:wrap;
    }
    ul li{
      list-style:none;
      width:200px;
    }
    ul li img{
      width:100%;
    }
    ul li a{
      width:150;
      display:inline-block;
      padding:7px 10px;
      border:solid 1px #ccc;
      width: 50px;
      text-align:center;
      font-family:Verdana, Geneva, Tahoma, sans-serif;
    }
  </style>
  <ul>
    <li>
      
      <a href="/edit/<%=y.single_image%>">Edit</a>
    </li>
  </ul>
<%})%>
```

