**Занятие jQuery ↔ Node.js**

**Тема: закачивание файлов на сервер (upload)**

На сервере Express + multer (+request)

На клиенте:

**Случай 1.** Примитивный уровень на клиенте. Форма.

http://kodaktor.ru/upload1\_b033b

|  |
| --- |
| **<**form method**="**post**"** action**="**http://kodaktor.ru:8886/api/fileupload**"** **enctype="multipart/form-data">**  **<**input type**="**file**"** **name="**fileUploaded**">**  **<**br**><**input type**="**submit**">** **<**/form**>** |

Этот уровень обеспечивает кнопку примитивного вида, позволяющую загрузить 1 файл за раз, при этом страница с формой заменяется в окне браузера на то, что пришло в ответе от сервера.

**Случай 2. Примитивный AJAX на основе современного HTML5**

[upload1\_1984e](http://kodaktor.ru/upload1_1984e)

Шаг 1. Учимся извлекать информацию из выбранного файла

|  |
| --- |
| **<**input id**=**"fileUploaded" type**=**"file" **name=**"fileUploaded" data**-**url**=**"http://kodaktor.ru:8886/api/fileupload"**>** **<**script**>**  **document**.querySelector('#fileUploaded').addEventListener('change',ev**=>**{  **console**.log( ev.target.files[0].name **+** ' ' **+** ev.target.dataset.url);  }); **<**/script**>** |

<http://kodaktor.ru/g/upload_ajax>

Шаг 2. Отправляем файл

|  |
| --- |
| <!DOCTYPE html> <**html**> <**head**><**title**>file ajax upload</**title**></**head**>  <**body**>  <**input** id="fileUploaded" name="fileUploaded" type="file" data-url="http://kodaktor.ru:8886**/api/fileupload**">  <**script**>  **console**.clear();  **document**.querySelector('#fileUploaded').addEventListener('change',ev**=>**{  **var** file **=** ev.target.files[0],  data **=** **new** **FormData**(),   xhr **=** **new** **XMLHttpRequest**();  data.append('fileUploaded', file, file.name);   xhr.open('POST', ev.target.dataset.url, **true**);  xhr.send(data);  xhr.onload **=** **function**(e){  **console**.log( (this.status **===** 200) ? 'ok ' **+** this.responseText : 'fail' );  };  });  </**script**>    </**body**> </**html**> |

На сервере:

|  |
| --- |
| **const** PORT **=** 8886; **var** express **=** require('express'),  app **=** express(),  request **=** require('request'),  url **=** require('url'),  multer **=** require('multer'); module.exports **=** (()**=>**{  **function** **inner**(){  this.start **=** whatToDo**=>**{  app.use(express.static(\_\_dirname **+** '/public'))  .use((req, res, next)**=>**next());   app.get('/', (req, res) **=>** {  //take client part from kodaktor  request(url.format({protocol:'http',host:'kodaktor.ru',pathname: '/g/upload\_ajax'})).pipe(res);  });    **var** storage **=** multer.diskStorage({  destination: (req, file, callback) **=>** callback(**null**, './uploads'),  filename: (req, file, callback) **=>** {  **console**.log(file.originalname);   callback(**null**, file.fieldname **+** '-' **+** **Date**.now());   }  });  **var** upload **=** multer({ storage : storage}).single('fileUploaded');   //<input type='file' name='fileUploaded'> or data.append('fileUploaded', file, file.name);    //route to accepting files  app.post(**'/api/fileupload'**,(req,res)**=>**{  res.set({'Access-Control-Allow-Origin': '\*', 'elias': 'goss'});  //without the header we can upload, but cannot get response    upload( req, res, err**=>**(err)?res.end('Error uploading file'):res.end('File well uploaded'));  });    app.listen(process.env.port**||**PORT,()**=>console**.log('--> Port %d listening!',PORT));  };   }  **return** **new** **inner**; })(); |

При записи ссылок на файлы в БД нам следует фиксировать не только file.originalname, но и дату и IP и другие доступные параметры (<https://github.com/expressjs/multer>):

* encoding
* mimetype
* size

| **Key** | **Description** | **Note** |
| --- | --- | --- |
| destination | The folder to which the file has been saved | DiskStorage |
| filename | The name of the file within the destination | DiskStorage |
| path | The full path to the uploaded file | DiskStorage |
| buffer | A Buffer of the entire file | MemoryStorage |