



# 파일 업로드 및 게시물 등록

Web 개발 기초





1. 파일 업로드

2. 게시물 등록

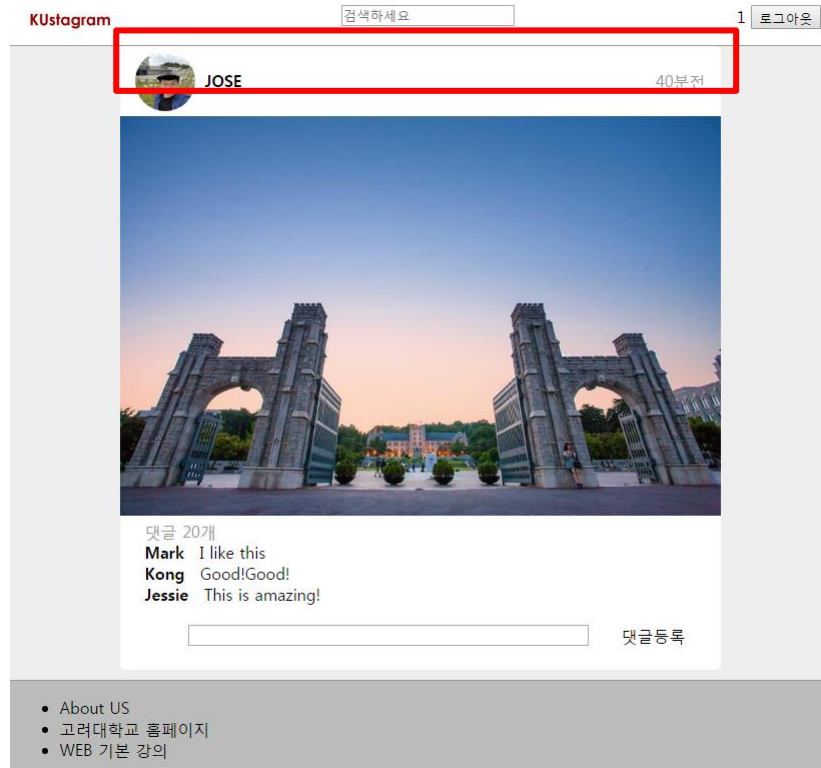


# 1. 파일 업로드

## 파일 업로드 & 게시물 작성 템플릿 구성

게시물 작성을 하고 사진을 업로드 하기 위해서 먼저 기본 템플릿을 구성하는 것이 필요하다.

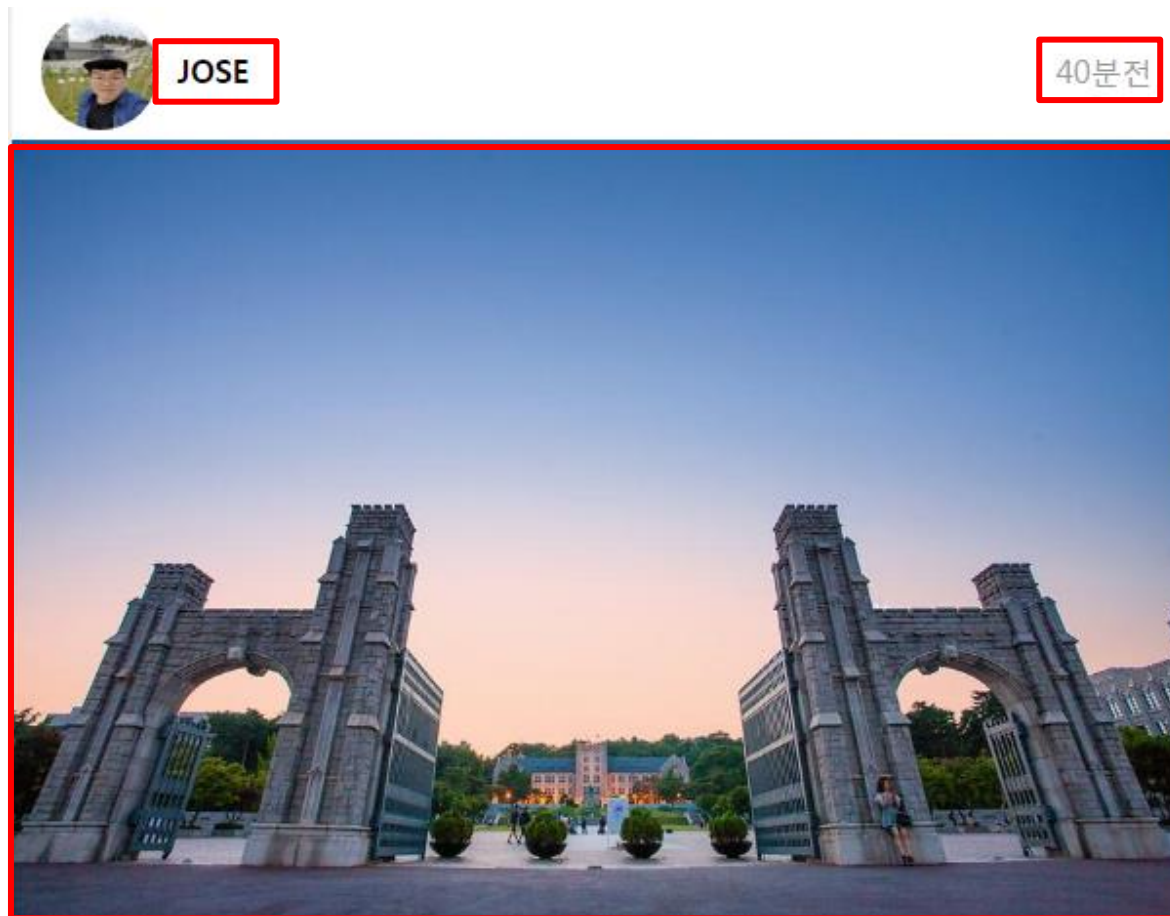
기본 템플릿은 아래 표시 부분(상단바와 게시물 사이 구간)에 구성한다.



# 1. 파일 업로드

## 파일 업로드 & 게시물 작성 템플릿 구성

게시물은 크게 userID, 작성시간, 사진, 글 내용 으로 구성된다.





# 1. 파일 업로드

## 파일 업로드 & 게시물 작성 템플릿 구성

Index.ejs 파일에서 main-body 클래스 위에 아래와 같이 작성해준다.

```
</div>
</div>
</div>

<div class="content-upload-container">
  <form action="/upload" method="post" enctype="multipart/form-data" style="text-align: center;">
    <textarea name="write-text" class="write-text-area" id="content" placeholder="내용을 입력해주세요"></textarea>
    <input type="file" id="getfile" name="userfile">
    <input type="submit" class="homework-submit-btn" value="과제 제출!"/>
  </form>
</div>

<div class="main-body">
  <div class="post">
    <div class="post-header">
```

# 1. 파일 업로드

## 파일 업로드 & 게시물 작성 템플릿 구성

브라우저에서 확인해보면 아래와 같이 나온다  
CSS를 수정해 스타일을 적용해주자



# 1. 파일 업로드

## 파일 업로드 & 게시물 작성 템플릿 구성

Style.css 의 하단에 내용을 추가해준다

```
.content-upload-container {  
  display: flex;  
  display: -webkit-flex;  
  align-items: center;  
  justify-content: center;  
  -webkit-align-items: center;  
  -webkit-justify-content: center;  
  padding: 10px;  
}  
  
.write-text-area {  
  width: 100%;  
  height: 60px;  
  padding: 10px;  
  margin-bottom: 10px;  
}  
  
.content-upload-container #getfile {  
  width: 100%;  
}
```

# 1. 파일 업로드

## 파일 업로드 & 게시물 작성 템플릿 구성

브라우저로 확인해보면 아래와 같은 화면을 볼 수 있다.

대충 배열된 스타일이지만 중앙으로 잘 정렬되었으니 만족하고 넘어간다.



The screenshot shows a web interface for Kustagram. At the top left is the logo 'Kustagram'. In the top center is a search bar with the placeholder text '검색하세요'. At the top right is a user profile icon with the number '1' and a '로그아웃' (Logout) button. The main content area is a light gray box. Inside this box, there is a large text input field with the placeholder text '내용을 입력해주세요'. Below this field is a button labeled '파일 선택' (File Selection) and a status message '선택된 파일 없음' (No files selected). To the right of these is a button labeled '과제 제출!' (Submit Assignment!). Below the main form area is a user profile card for 'JOSE', featuring a circular profile picture and the text '40분전' (40 minutes ago). The bottom of the page is a solid blue bar.





# 1. 파일 업로드

## 파일 업로드

파일 업로드를 위해 `multer`, `fs` 와 같은 모듈을 사용할 것이다.

`Multer` 는 파일 업로드 / 다운로드를 수행하기 위한 모듈이다.

`Multer`를 통해 파일이 업로드되는 경로, 파일명 등을 설정할 수 있다.

`fs` 는 이름 그대로 File system 이며, 파일의 I/O 에 관련된 기능을 수행한다.

`fs`를 이용해 업로드한 파일을 관리할 수 있다.

(이번 강좌에서는 업로드한 파일을 삭제할 때 사용할 것)



# 1. 파일 업로드

## 파일 업로드

먼저 모듈을 적용한다.

Server.js 파일을 열고 상단에 아래 내용을 추가해준다

```
var fs = require("fs");
var multer = require('multer');
var storage = multer.diskStorage({
  destination: function (req, file, cb) {
    cb(null, './static/img/uploads');
  },
  filename: function (req, file, cb) {
    var originalname = file.originalname;
    var type = originalname.split(".")[1];
    cb(null, file.fieldname + '-' + Date.now() + '.' + type);
  }
});

var upload = multer({ storage: storage });
```



# 1. 파일 업로드

## 파일 업로드

Server.js 의 하단의 router 에도 아래와 같이 추가해준다.

```
var router = require('./router/route')(app, passport, fs, upload);
```

추가 되었으면, router.js 의 상단 module.exports 에도 아래와 같이 추가한다.

```
module.exports = function(app, passport, fs, upload)
{
  app.get('/', function(req, res){
    // ...
  });
}
```



# 1. 파일 업로드

## 파일 업로드

파일 업로드를 위해 DB 작업을 해야 하므로, 테이블을 생성한다.

테이블 이름:  Add  column(s)

이름	종류	길이/값	기본값	데이터정렬방식	보기	Null	인덱스	A_I
<input type="text" value="idx"/> <small>Pick from Central Columns</small>	<input type="text" value="INT"/>	<input type="text"/>	<input type="text" value="없음"/>	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text" value="PRIMARY"/> <small>PRIMARY</small>	<input checked="" type="checkbox"/>
<input type="text" value="user_id"/> <small>Pick from Central Columns</small>	<input type="text" value="VARCHAR"/>	<input type="text" value="30"/>	<input type="text" value="없음"/>	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text" value="---"/>	<input type="checkbox"/>
<input type="text" value="img_src"/> <small>Pick from Central Columns</small>	<input type="text" value="VARCHAR"/>	<input type="text" value="50"/>	<input type="text" value="없음"/>	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text" value="---"/>	<input type="checkbox"/>
<input type="text" value="content"/> <small>Pick from Central Columns</small>	<input type="text" value="TEXT"/>	<input type="text"/>	<input type="text" value="없음"/>	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text" value="---"/>	<input type="checkbox"/>
<input type="text" value="reg_date"/> <small>Pick from Central Columns</small>	<input type="text" value="VARCHAR"/>	<input type="text" value="30"/>	<input type="text" value="없음"/>	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text" value="---"/>	<input type="checkbox"/>



# 1. 파일 업로드

## 파일 업로드

테이블 생성 완료

	#	이름	종류	데이터정렬방식	보기	Null	기본값	추가
<input type="checkbox"/>	1	<b>idx</b> 	int(11)			아니오	없음	AUTO_INCREMENT
<input type="checkbox"/>	2	<b>user_id</b>	varchar(30)	utf8_general_ci		아니오	없음	
<input type="checkbox"/>	3	<b>img_src</b>	varchar(50)	utf8_general_ci		아니오	없음	
<input type="checkbox"/>	4	<b>content</b>	text	utf8_general_ci		아니오	없음	
<input type="checkbox"/>	5	<b>reg_date</b>	varchar(30)	utf8_general_ci		아니오	없음	

## 2. 게시물 등록



이제 게시물을 등록하기 위한 준비가 끝났다.

게시물의 등록은 아래와 같은 과정을 따른다

1. 먼저 글을 적고 파일을 첨부해서 버튼을 누르면 route.js 로 Form 의 입력 값(글의 내용, 첨부 파일의 경로)을 넘긴다.
2. Route.js 에서 작성한 내용을 DB에 저장한다.



## 2. 게시물 등록

### Form 입력 값 넘기기

먼저 form 의 입력 값을 서버 측으로 넘긴다.

Index.ejs 에서 입력 값을 넘기는 form의 action을 보면 /upload 로 되어 있고, 방식은 post 방식을 따른다.

이에 맞게 route.js 파일을 작성하자.

```
<div class="content-upload-container">
  <form action="/upload" method="post" enctype="multipart/form-data" style="text-align: center;">
    <textarea name="write-text" class="write-text-area" id="content" placeholder="내용을 입력해주세요"></textarea>
    <input type="file" id="getfile" name="userfile">
    <input type="submit" class="homework-submit-btn" value="과제 제출!"/>
  </form>
</div>
```



## 2. 게시물 등록

### Form 입력 값 넘기기

Router.js 에 아래와 같이 작성한다.

표시된 부분이 form 입력 값을 서버 측에서 받는 부분이다

```
app.post('/upload', upload.single('userfile'), function(req, res){
  var db = mysql.createConnection(db_config);
  var username = req.user.user_id;
  var content = req.body.write_text;
  var filename = req.file.filename;
  var reg_date = new Date().toISOString().replace(/T/, ' ').replace(/\..+/, '');
  var sql = 'INSERT INTO `post`(`user_id`, `img_src`, `content`, `reg_date`)';
  sql += ' VALUES (?, ?, ?, ?)';
  db.query(sql, [username, filename, content, reg_date], function(err, results){
    if(err){
      console.error('mysql connection error');
      console.error(err);
      throw err;
    }
    console.log('homework input success');
  });
  db.end();
  res.redirect('/');
});
```





## 2. 게시물 등록

### Form 입력 값 넘기기

Index.ejs 에서 표시 부분을 수정!

Textarea의 name 항목을

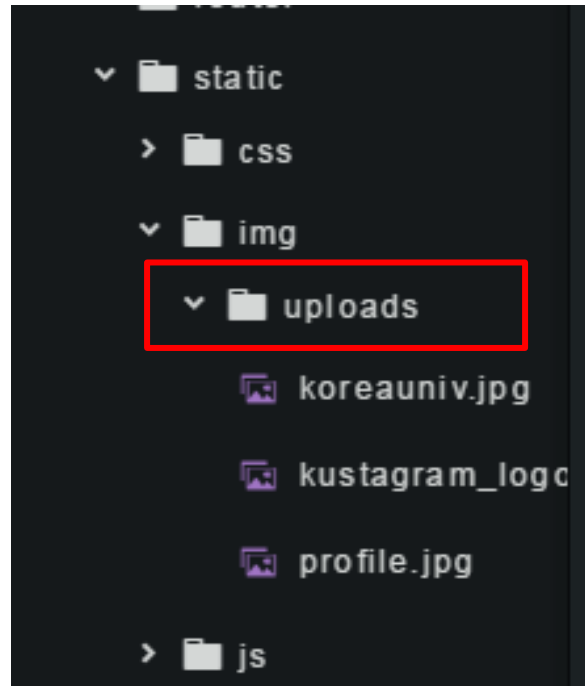
Write-text 에서 write\_text 로 수정 (언더바!)

```
<div class="content-upload-container">
  <form action="/upload" method="post" enctype="mu
    <textarea name="write_text" class="write-text-
    <input type="file" id="getfile" name="userfile
    <input type="submit" class="homework-submit-bt
  </form>
</div>
```

## 2. 게시물 등록

### Form 입력 값 넘기기

그리고 static > img 폴더에 uploads 폴더 추가





# 2. 게시물 등록

## DB 입력 하기

Router.js 에서 표시된 부분이 DB 에 입력하는 쿼리부분이다

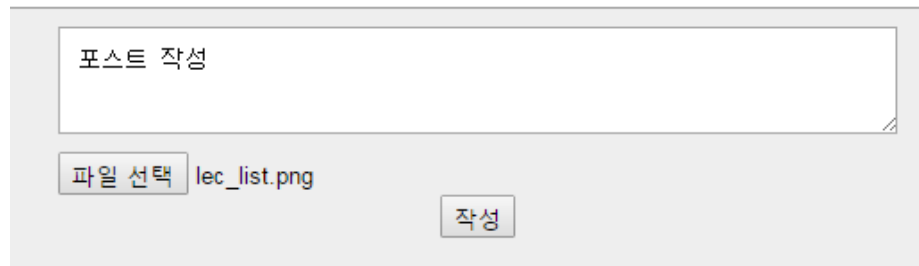
```
app.post('/upload', upload.single('userfile'), function(req, res){
  var db = mysql.createConnection(db_config);
  var username = req.user.user_id;
  var content = req.body.write_text;
  var filename = req.file.filename;
  var reg_date = new Date().toISOString().replace(/T/, ' ').replace(/\..+/, '');
  var sql = 'INSERT INTO `post`(`user_id`, `img_src`, `content`, `reg_date`)';
  sql += ' VALUES (?, ?, ?, ?)';
  db.query(sql, [username, filename, content, reg_date], function(err, results){
    if(err){
      console.error('mysql connection error');
      console.error(err);
      throw err;
    }
    console.log('homework input success');
  });
  db.end();
  res.redirect('/');
});
```



## 2. 게시물 등록

### DB 입력 하기

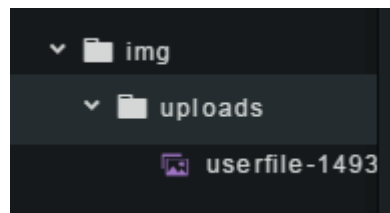
브라우저에서 임의의 내용과, 임의의 파일을 첨부하고 작성을 누르면



DB에 입력한 값이 저장된 것을 확인할 수 있다.

+ 옵션						
←T→	▼	idx	user_id	img_src	content	reg_date
<input type="checkbox"/>	수정	복사	삭제	2 1	userfile-1493726115067.png	포스트 작성 2017-05-02 11:55:15

파일 업로드 또한 제대로 된 것을 확인할 수 있다.





다음 시간에는...

1. 게시물 출력

2. 게시물 삭제