

Node.js 멀티파트 처리



2016년도 2학기 2학년 방과후학교

멀티 파트 처리

Content-Type: multipart/form-data; boundary=XXXXYYYYZZZ

--XXXXYYYYZZZ

Content-Disposition: form-data; name="name"

value1

--XXXXYYYYZZZ

Content-Disposition: form-data; name="poster"; filename="IMG_4757.JPG"

Content-Type: image/jpeg

PGh0bWw+CiAgPGhIYWQ+CiAgPC9oZWfkPgogIDxib2R5PgogICAgaPHA+VGhpcyBpcyB0aGUgYm9keSBvZiB0aGUgbWVzc2FnZS48L3A+CiAgPC9ib2R5Pgo8L2h0bWw+Cg==

--XXXXYYYYZZZ--

파트1

파트2

```
<form method='post' action='/upload' enctype='multipart/form-data'>
  <input type="text" name="name">
  <input type="file" name="poster">
</form>
```

멀티 파트 처리

- 멀티 파트 처리를 위해 먼저 header의 content-type을 읽어 multipart/form-data인지를 확인하고, boundary 값을 추출한다.
- 메시지 바디 분석을 위해 전체 데이터를 boundary로 split하고, split된 결과에 따라 별도 처리를 한다.
- 멀티파트 메시지를 분석을 위해 멀티파트 메세지 분석용 모듈을 사용한다.
- 멀티파트 메세지 분석용 모듈 : formidable, multer
- npm install formidable

멀티 파트 분석 모듈 - formidable

- formidable 모듈이 제공하는 클래스
 - ✓ 요청 메시지를 분석하는 IncomingForm 클래스
 - ✓ 업로드된 파일의 정보를 다루는 File 클래스
- `form.parse(req, function (err, fields, files) { }`
 - ✓ `fields` : 이름-값 데이터
 - ✓ `files` : 업로드 파일 정보

멀티 파트 분석 모듈 - formidable

- 멀티파트로 전송되는 파일은 임의의 이름으로 변경되어 특정 위치에 저장된다.
- 기본적으로 OS의 임시 폴더 위치, uploadDir 프로퍼티로 설정할 수 있다.
- 같은 이름의 파일을 업로드해도 이름으로 겹쳐서 에러가 발생하지 않도록 파일명 뒤에 난수를 붙인다.

▲ upload

upload_42b24f7635fb20ec917438c8eaaed8f3

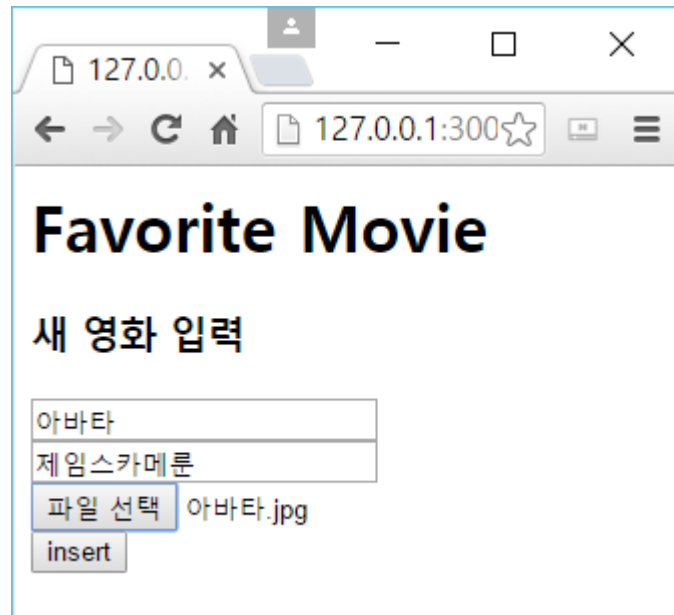
upload_4abc6c11e33d474d5356f49c97334afd

upload_4ff870f1a28461b06784f1a10f311dd4

upload_95bd841df36875a12e5a04dd2925d5fd

[실습] 영화 관리 어플리케이션 작성

- GET 방식으로 접근하면 전체 영화 목록과 이미지를 출력한다.
- 영화 정보 입력 폼을 통해 POST 방식으로 새로운 영화 정보를 등록한다.
- POST 요청 후 refresh하면 중복 Post 요청이 되므로 PRG 패턴을 추가한다.
- 파일명 : 15_multipartWmovie.js



A screenshot of a web browser window with the address bar showing '127.0.0.1:3000'. The page title is 'Favorite Movie'. Below the title, there is a section labeled '새 영화 입력' (New Movie Input). It contains two text input fields: the first contains '아바타' (Avatar) and the second contains '제임스카메론' (James Cameron). Below these fields is a '파일 선택' (File Select) button, which has '아바타.jpg' displayed next to it. At the bottom of the form is an 'insert' button.



A screenshot of a web browser window with the address bar showing '127.0.0.1:3000'. The page title is 'Favorite Movie'. Below the title, there is a list item '아바타 (제임스카메론)' (Avatar (James Cameron)). To the right of the text is a movie poster for Avatar. Below the poster, there is a section labeled '새 영화 입력' (New Movie Input). It contains two text input fields: the first is empty and labeled '영화제목' (Movie Title), and the second is empty and labeled '감독' (Director). Below these fields is a '파일 선택' (File Select) button, which has '선택된 파일 없음' (No file selected) displayed next to it. At the bottom of the form is an 'insert' button.