

샘플사이트 화면정의서 + REST API

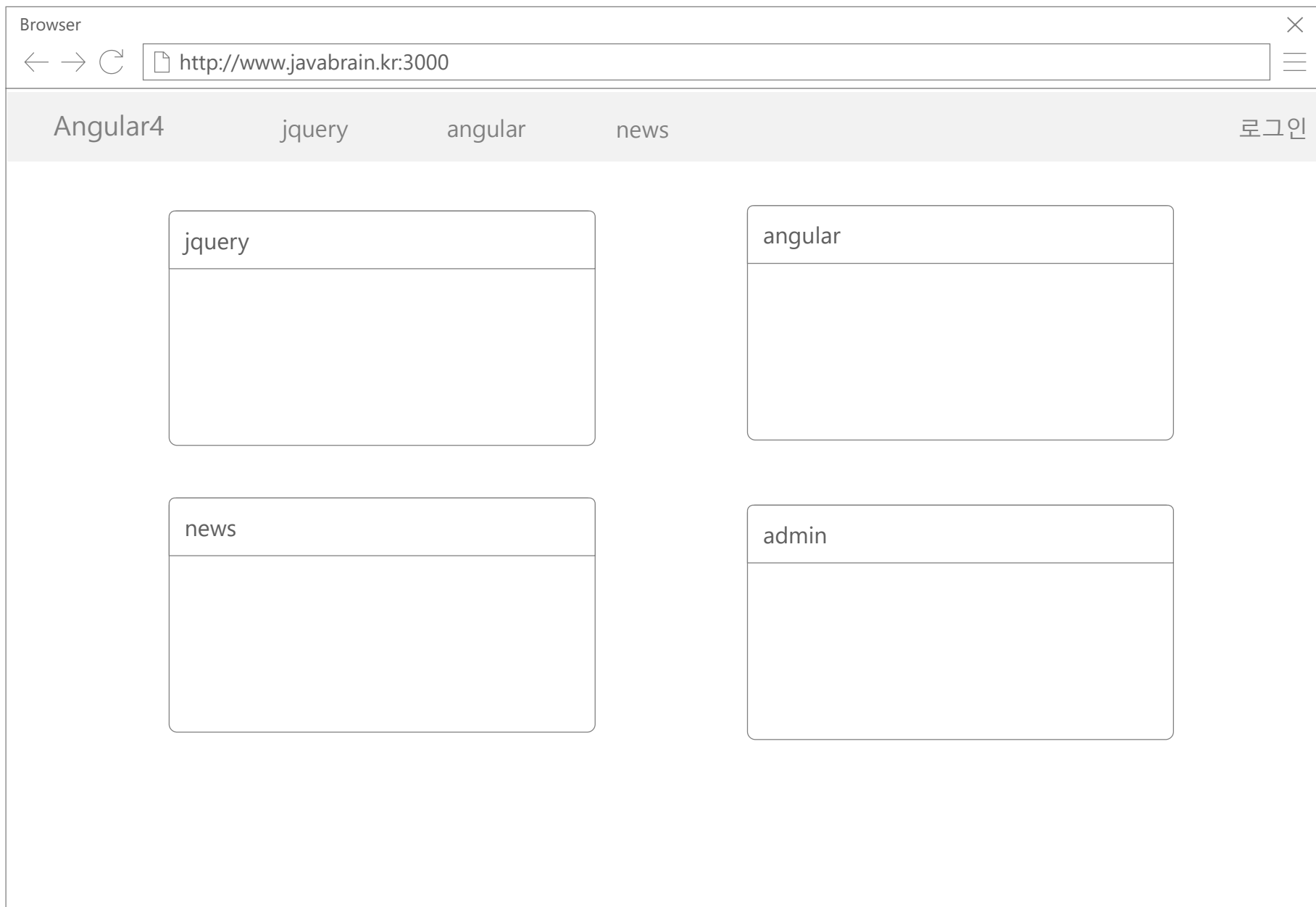
* 샘플사이트:

- 사용자 사이트: <http://www.javabrain.kr:3000>
- 관리자 사이트: <http://www.javabrain.kr:3000/admin>

eastflag@gmail.com

010-3010-1482

이동기



* 메인화면

* 최초에 bootstrap으로 작성

- * 나중에 material 디자인 적용
- 반응형 웹
- 화면이 좁아지면 메뉴가 우측으로 이동
- 판넬은 한줄에 한개씩

- 모바일

Angular4

로그인



Browser

← → ↻ http://www.javabrain.kr:3000

Angular4 jquery angular news 로그인

jquery Todo

완료	todo	생성일	수정일	삭제
<input type="checkbox"/>	청소하기	2017-04-05 17:12:12		<input type="button" value="삭제"/>
<input checked="" type="checkbox"/>	빨래하기	2017-04-04 17:12:12	2017-04-06 17:12:12	<input type="button" value="삭제"/>
<input type="checkbox"/>	설거지하기	2017-04-03 17:12:12		<input type="button" value="삭제"/>

* JQuery 화면
Bootstrap으로 화면 구성
Navbar를 사용하여 상단 메뉴
바 구성
form-group, form-control 등을
사용하여 입력 UI 구성
Table 사용하여 기본 테이블 작
성
Jquery 기능 사용
Ajax 사용
Javascript dom 과 비교

1. 할일을 입력하고 추가 버튼
을 누르면 하단 리스트의 **최상
단**에 추가된다.

생성일은 new Date()를 사용하
여 현재 날짜를 자동으로 기록
한다.

2. 체크박스를 클릭하면 완료처
리가 되어야 하고 todo에 취소
선이 그으져야 하고 수정일에
현재 날짜가 기록되어야 한다.

3. 삭제 버튼을 누르면 리스트
에서 사라진다.

Browser

← → ↻ http://www.javabrain.kr:3000

Angular4 jquery **angular** news 로그인

Angular4 Todo App

완료	todo	생성일	수정일	삭제
<input type="checkbox"/>	청소하기	2017-04-05 17:12:12		<input type="button" value="삭제"/>
<input checked="" type="checkbox"/>	빨래하기	2017-04-04 17:12:12	2017-04-06 17:12:12	<input type="button" value="삭제"/>
<input type="checkbox"/>	설거지하기	2017-04-03 17:12:12		<input type="button" value="삭제"/>

Alert

⚠ 삭제하시겠습니까?

Abort Cancel OK

* 기능구현 : angular2 + Array

MVC 프레임워크를 제공해주는 angular2를 사용하여 구현 Model은 여전히 Array를 사용

1. 할일을 입력하고 추가 버튼을 누르면 하단 리스트의 최상단에 추가된다.
생성일은 new Date()를 사용하여 현재 날짜를 자동으로 기록한다.

2. 체크박스를 클릭하면 완료처리가 되어야 하고 todo에 취소선이 그어져야 하고 수정일에 현재 날짜가 기록되어야 한다.

3. 삭제 버튼을 누르면 삭제를 할지 물어보는 alert 박스가 뜨고 확인 버튼을 클릭하면 삭제되고 리스트에서 사라진다.

Browser

← → ↻ http://www.javabrain.kr:3000

Angular4 jquery **angular** news 로그인

Angular4 Todo App

할일

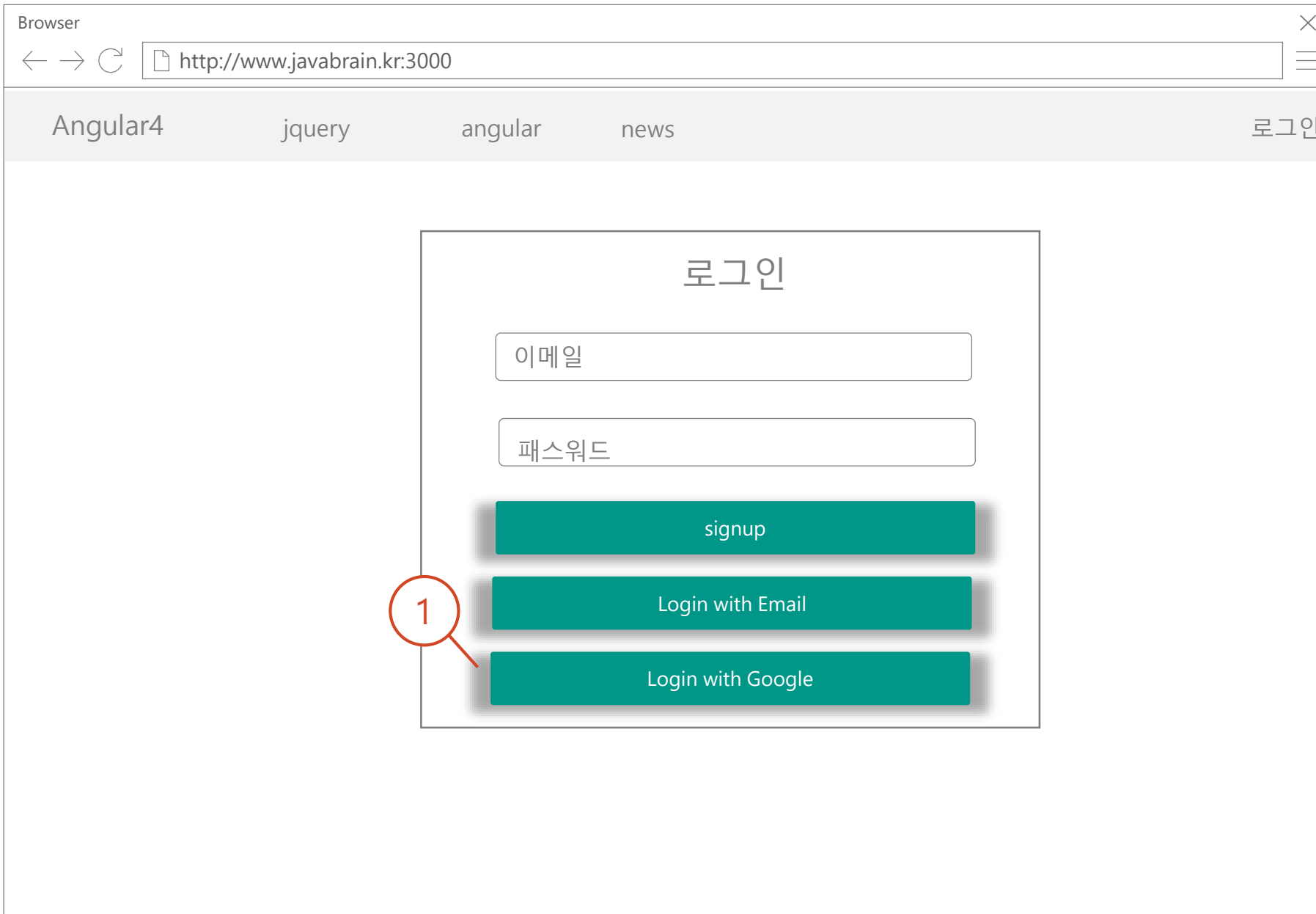
완료	todo	생성일	수정일	
미완료	청소하기	2017-04-05 17:12:12		<input type="button" value="수정"/> <input type="button" value="삭제"/>
완료	빨래하기	2017-04-04 17:12:12	2017-04-06 17:12:12	<input type="button" value="수정"/> <input type="button" value="삭제"/>
미완료	설것이하기	2017-04-03 17:12:12		<input type="button" value="수정"/> <input type="button" value="삭제"/>

완료	todo	생성일	수정일	
<input type="checkbox"/>	<input type="text" value="청소하기"/>	2017-04-05 17:12:12		<input type="button" value="저장"/> <input type="button" value="취소"/>
완료	빨래하기	2017-04-04 17:12:12	2017-04-06 17:12:12	<input type="button" value="저장"/> <input type="button" value="취소"/>
미완료	설것이하기	2017-04-03 17:12:12		<input type="button" value="저장"/> <input type="button" value="취소"/>

Diagram annotations: A red circle with '1' points to the '수정' button in the first table. A blue arrow points from the '수정' button in the first table to the '수정일' column header in the second table. A red circle with '2' points to the '저장' button in the second table. A red circle with '3' points to the '삭제' button in the second table.

* 기능구현 : angular2 + HTTP

1. 수정 버튼을 클릭하면 아래 화면으로 전환된다.
2. 완료, todo를 수정한뒤에 저장 버튼을 클릭하면 서버에 반영되고 다시 위 화면으로 전환된다.
3. 취소버튼을 누르면 이전 데이터로 복원된다.



* 기능구현 : angular2 + HTTP

1. 구글 로그인을 한다.

Browser

← → ↻ http://www.javabrain.kr:3000

Angular4 jquery angular news 로그인

회원가입

☒ 이용약관동의

Textarea

0000

☒ 개인정보이용동의

Textarea

0000

- * 기능구현 : angular2 + HTTP
1. 서버에 회원정보가 없으면 회원가입을 시킨다.

Browser

← → ↻ http://www.javabrain.kr:3000

Angular4 jquery angular news 로그인

뉴스 목록

제목1	2017-10-13 12: 45
제목2	2017-10-13 12: 45
제목3 [4]	2017-10-13 12: 45
제목4	2017-10-13 12: 45

« 1 2 3 4 5 »

* 기능구현 : angular2 + HTTP

1. 댓글 개수를 표시
2. 현재페이지와 페이지 개수를 표시

Browser

← → ↻ http://www.javabrain.kr:3000

Angular4 jquery angular news 로그인

제목

내용

.

.

.

댓글 작성

댓글을 작성하세요 1

0/300

완료

홍길동
댓글1입니다.

2017-10-13 12: 45

홍길동
댓글2입니다.

2017-10-13 12: 45

삭제 2

* 기능구현 : angular2 + HTTP

1. 댓글을 작성한다. 만일 로그인이 되지 않은 상태라면 로그인창으로 튕겨낸다.
2. 댓글 목록 리스트가 댓글 작성 아래에 보여진다. 만일 자기 자신이 작성한 댓글이라면 삭제버튼이 보인다.

* 사용자 REST API 규격서

Server: www.javabrain.kr

Port: 8080

Request: application/json

Response: json

* Todo 메뉴

1) 할일 목록 가져오기

- protocol: /api/todo
- Method: get
- Query Parameter: start_index=0, page_size=5 (옵션)
- Response

```
[
  {
    "todo_id": 194,
    "isFinished": true,
    "todo": "수정완료3",
    "created": "2017-09-02 10:23:27.0",
    "updated": "2017-09-02 11:32:26.0"
  },
  {
    "todo_id": 190,
    "isFinished": false,
    "todo": "111",
    "created": "2017-09-01 21:23:32.0",
    "updated": "2017-10-18 16:33:27.0"
  }
]
```

2) 할일 추가하기

- protocol: /api/todo
- Method: post
- Request

```
{
  "todo": "빨래하기"
}
- Response
{
  "todo_id": 195,
  "isFinished": false,
  "todo": "빨래하기",
  "created": "2017-10-21 12:08:35.0",
  "updated": "2017-10-21 12:08:35.0"
}
```

3) 할일 수정하기

- protocol: /api/todo
- Method: put
- Request

```
{
  "todo_id": 195,
  "isFinished": true,
  "todo": "빨래 끝"
}
- Response
{
  "todo_id": 195,
  "isFinished": true,
  "todo": "빨래 끝",
  "created": "2017-10-21 12:08:35.0",
  "updated": "2017-10-21 12:10:07.0"
}
```

* 사용자 REST API 규격서

4) 할일 삭제하기

- protocol: /api/todo?todo_id=1
- Method: delete
- Response

```
{
  "result": 0,
  "value": "success"
}
```

* 로그인 메뉴

1) 회원 등록

- protocol: /api/signUp
- Method: POST
- request

```
{
  "email": "lee@test.com",
  "nickname": "강길동"
}
```

- response

```
{
  "result": 0,
  "value": "success",
  "data": {
    "member_id": 13,
    "email": "kang4@test.com",
    "nickname": "강길동",
    "role": null,
    "token": "eyJhbGciOiJIUzI1NiJ9.eyJqdGkiOiIxMyIsImIhdCI6MTUxMTIzMjI1NiwiOiJjoiliwiaXNzIjoia2FuZzRAdGVzdC5jb20iLCJleHAiOiE1MTEzMTg2NTZ9.pVMcZNqmi0R2cgJBjUDmos50HKhzxV2Hwdu1OpHtAjA"
  }
}
```

2) 로그인

- protocol: /api/todo?todo_id=1
- Method: post
- request

```
{
  "email": "lee@test.com"
}
```

- Response

```
{
  "result": 0,
  "value": "success",
  "data": {
    "member_id": 13,
    "email": "kang4@test.com",
    "nickname": "강길동",
    "role": null,
    "token": "eyJhbGciOiJIUzI1NiJ9.eyJqdGkiOiIxMyIsImIhdCI6MTUxMTIzMjI1NiwiOiJjoiliwiaXNzIjoia2FuZzRAdGVzdC5jb20iLCJleHAiOiE1MTEzMTg2NTZ9.pVMcZNqmi0R2cgJBjUDmos50HKhzxV2Hwdu1OpHtAjA"
  }
}
```

* 사용자 REST API 규격서

* 뉴스 메뉴

1) 뉴스 목록 보기

- protocol: /api/newsList

- Method: POST

- request

```
{
  "start_index": 0,
  "page_size": 5
}
```

- response

```
{
  "result": 0,
  "value": "success",
  "data": [
    {
      "news_id": 13,
      "title": "1234",
      "created": "2017-11-12 22:54:38.0",
      "updated": "2017-11-13 02:22:45.0",
      "comment_count": 1
    },
    {
      "news_id": 12,
      "title": "1234",
      "created": "2017-11-12 22:54:38.0",
      "updated": "2017-11-13 02:22:45.0",
      "comment_count": 0
    }
  ]
}
```

2) 뉴스 상세보기

- protocol: /api/news?news_id=1

- Method: get

- Response

```
{
  "news_id": 13,
  "title": "1234",
  "content": "<p>1234</p>\n\n<p><img
src=\n"http://www.javabrain.kr/assets/upload/news/1510507360110.png\n"
style=\n"max-width:100%\n" /></p>\n",
  "created": "2017-11-12 22:54:38.0",
  "updated": "2017-11-13 02:22:45.0"
}
```

* 사용자 REST API 규격서

* 댓글 메뉴

1) 댓글 목록 보기

- protocol: /api/comment?news_id=???
- Method: GET
- response

```
[
  {
    "comment_id":3,
    "member_id":1,
    "news_id":7,
    "content":"test2",
    "updated":"2017-10-17 23:51:35.0",
    "nickname":"홍길동"
  },
  {
    "comment_id":2,
    "member_id":2,
    "news_id":7,
    "content":"test2",
    "updated":"2017-10-17 23:51:35.0",
    "nickname":"이몽룡"
  }
]
```

2) 댓글쓰기

- protocol: /api/comment
- Method: POST
- Request

```
{
  content: "1234",
  member_id: 13,
  news_id: 7
}
```

- Response

```
{
  "result": 0,
  "value": "success"
}
```

3) 댓글삭제

- protocol: /api/comment?comment_id
- Method: DELETE
- Response

```
{
  "result": 0,
  "value": "success"
}
```

관리자페이지

관리자사이트

뉴스관리

ADMIN

아이디

패스워드

로그인

* 로그인 페이지

1. 페이지 네이션
목록이 길어질 경우 하단 페이지 네이션
을 구현한다.
Material design의 페이징 이용

관리자페이지

Admin

뉴스등록

뉴스 목록

글쓰기

제목1	2017-10-13 12: 45
제목2	2017-10-13 12: 45
제목3	2017-10-13 12: 45
제목4	2017-10-13 12: 45

« 1 2 3 4 5 »

* 뉴스목록

1. 글쓰기 버튼을 누르면 글쓰기 화면으로 이동
2. 뉴스 목록이 많을 경우 페이지네이션을 한다.

관리자페이지

Admin

뉴스등록

뉴스 등록

제목

내용

업로드 이미지 선택

파일선택

선택된 파일 없음

1

저장

* 뉴스등록

1. 파일선택을 하여 이미지를 업로드 할수 있다.
이미지 업로드 api 참조.

관리자페이지

Admin

뉴스등록

제목1

내용

.
. .
. .
. .
. .

수정

삭제

뉴스 목록



글쓰기

제목1

2017-10-13 12: 45

제목2

2017-10-13 12: 45

제목3

2017-10-13 12: 45

제목4

2017-10-13 12: 45

* 뉴스보기

1. 뉴스보기를 누르면 뉴스항목이 상단에 보여지고 수정 삭제를 할수 있다.

관리자페이지

Admin

뉴스등록

뉴스 수정

제목

제목1

내용

내용

업로드 이미지 선택

파일선택

선택된 파일 없음

저장

1

* 뉴스보기

1. 수정할 항목이 표시되고 저장을누르면 다시 목록으로 돌아간다.

Admin

뉴스등록

제목1

내용
.
.
.
.
.

수정

삭제

Alert

!

삭제하시겠습니까?

Abort

Cancel

OK

글쓰기

뉴스 목록

제목1	2017-10-13 12: 45
제목2	2017-10-13 12: 45
제목3	2017-10-13 12: 45
제목4	2017-10-13 12: 45

- * 뉴스보기
1. 삭제 버튼을 누르면 삭제를 물어보는 팝업 다이얼로그가 뜨고 확인을 누르면 뉴스목록으로 돌아간다.

* 관리자 REST API 규격서

Server: www.javabrain.kr:8080

Request: application/json

Response: json

1) 뉴스목록 가져오기

- Protocol: /api/newsList
- Method: post
- Request

```
{ "start_index": 0, "page_size": 5}
```

- Response

```
{
  "result": 0,
  "value": "success",
  "data": [
    {
      "news_id": 9,
      "title": "씹으면 씹을수록 치매 예방에 도움된다",
      "content": "xxx",
      "created": "2017-10-20 21:34:26.0",
      "updated": "2017-10-20 21:34:26.0",
      "comment_count": 0
    },
    {
      "news_id": 9,
      "title": "씹으면 씹을수록 치매 예방에 도움된다",
      "content": "xxx",
      "created": "2017-10-20 21:34:26.0",
      "updated": "2017-10-20 21:34:26.0",
      "comment_count": 0
    }
  ],
  total: 6
}
```

2) 뉴스 등록

- protocol: /api/news
- Method: post
- Request

```
{
  "content": "<p>333</p>ㄹ",
  "title": "123"
}
```

- Response

```
{
  "result": 0,
  "value": "success"
}
```

3) 뉴스 보기

- protocol: /api/news?news_id=xxx
- Method: get
- Request
- Response

```
{
  "news_id": 9,
  "title": "씹으면 씹을수록 치매 예방에 도움된다",
  "content": "xxx",
  "created": "2017-10-20 21:34:26.0",
  "updated": "2017-10-20 21:34:26.0",
  "comment_count": 0
},
```

* 관리자 REST API 규격서

4) 뉴스 수정

- protocol: /api/news
- Method: put
- Request

```
{
  "content": "<p>333</p>←",
  "title": "123"
}
```
- Response

```
{
  "result": 0,
  "value": "success"
}
```

5) 뉴스 삭제

- protocol: /api/news?news_id=xxx
- Method: delete
- Response

```
{
  "result": 0,
  "value": "success"
}
```

6) 이미지 등록

- protocol: /api/imageUpload
- Method: post (Multipart-Upload)
- Request

```
-----WebKitFormBoundary6BT9FyGNqkOediiT
Content-Disposition: form-data; name="file"; filename="brain.png"
Content-Type: image/png

-----WebKitFormBoundary6BT9FyGNqkOediiT—
xxxxxx
```
- Response

```
{"result":0,"value":"/assets/upload/news/1507943619140.png"}
```

* 관리자 REST API 규격서

POST

http://www.javabrain.kr:8080/api/imageUpload

Params

Send

Save

Authorization

Headers

Body

Pre-request Script

Tests

Code

☒ form-data

☐ x-www-form-urlencoded

☐ raw

☐ binary

	Key	Value	Description	...	Bulk Edit
<input checked="" type="checkbox"/>	file	<div>파일 선택</div> 0002862421_001_2...020180723691.jpg			
	New key	Value	Description		

Body

Cookies

Headers (6)

Test Results

Status: 200 OK

Time: 77 ms

Pretty

Raw

Preview

JSON

1

2

3

4

{

"result": 0,

"value": "/assets/upload/news/1508557419527.jpg"

}