

App center iOS

- Index -



1. postman
2. Json
3. Alamofire
4. Get / Post



POSTMAN

http://117.16.231.66:7

117.16.231.66:7000/re

+

...

No Environment

⌵

👁

⚙

POST

http://117.16.231.66:7001/boardSelect/

Params

Send

Save

Authorization

Headers (1)

Body

Pre-request Script

Tests

Cookies

Code

☐ form-data

☒ x-www-form-urlencoded

☐ raw

☐ binary

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>	department	컴퓨터공학부			
<input checked="" type="checkbox"/>					
	Key	Value	Description		

Body

Cookies (1)

Headers (7)

Test Results

Status: 200 OK

Time: 39 ms

Size: 6 KB

Pretty

Raw

Preview

JSON

⌵

🔍

```
78      "content_serial_id": 98,
79      "title": "스브 왜 이래",
80      "content": "몇",
81      "department": "컴퓨터공학부",
82      "timeSave": "2018-11-08T08:07:58.000Z",
83      "fileName": [
84        "1541664477613",
85        "1541664477746",
86        "1541664477832",
87        "1541664477936",
88        "1541664478178",
89        "1541664478270",
90        "1541664478450"
91      ],
92    },
93    {
94      "content_serial_id": 97,
95      "title": "",
96      "content": "What ",
97      "department": "컴퓨터공학부",
98      "timeSave": "2018-11-08T08:05:11.000Z",
99      "fileName": [
100        "1541664311044"
101      ]
102    }
  ]
}
```

Build

Browse

💡

🗑

🔍

?



POSTMAN

중고나라

Introduction

- ▶ 물품관련
- ▼ 계정관련
 - POST 회원가입기능
 - POST 로그인기능
 - POST 비밀번호 변경 기능
 - POST 비밀번호 찾기
 - POST 회원탈퇴
 - POST 핸드폰 번호 변경
 - GET 상품이미지 링크
- ▶ 쪽지
- ▼ 배너
 - GET iosBanner 이미지 파일 접근경로
 - POST 배너 링크
- ▶ 문의 및 신고

POST 회원가입기능

117.16.231.66:7000/account

id = 학번 passwd = 비밀번호 name = 이름 tel = 폰번호

헤더값 (토큰)필요없음

HEADERS

Content-Type application/x-www-form-urlencoded

BODY

id	201301484
passwd	0
name	임동완
tel	01021675629

Example Request

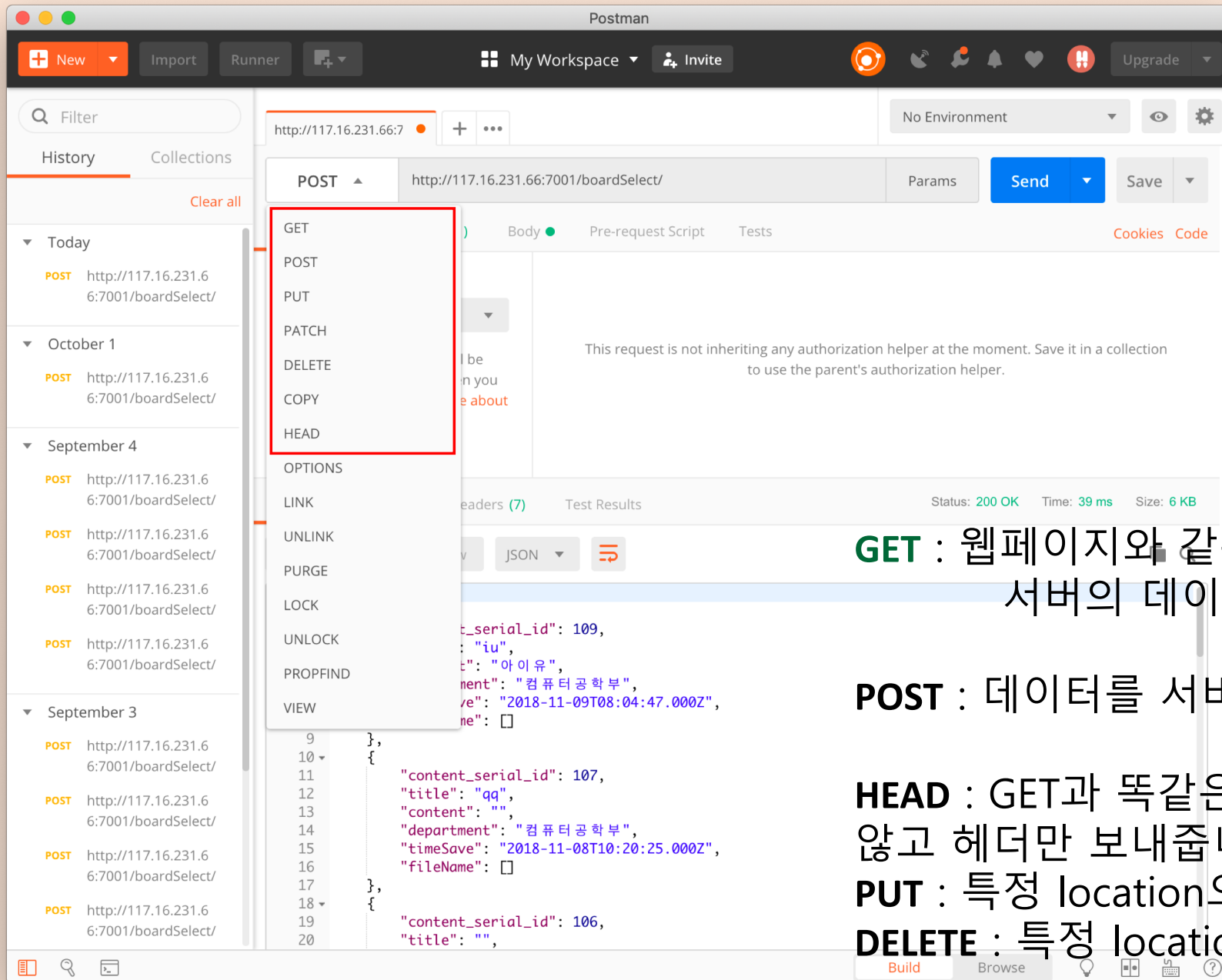
회원가입(성공) ▼

```
curl --request POST \  
  --url http://117.16.231.66:7000/account \  
  --header 'Content-Type: application/x-www-form-urlencoded' \  
  --data 'id=201301484&passwd=0&name=%EC%9E%84%EB%8F%99'
```

Example Response

200 OK

```
{  
  "ans": true  
}
```



GET : 웹페이지와 같은 데이터를 서버로부터 받아온다.
서버의 데이터를 변경하지는 않습니다.

POST : 데이터를 서버로 보낼 때 사용.

HEAD : GET과 똑같은 작동을 하지만 데이터는 보내주지 않고 헤더만 보내줍니다.

PUT : 특정 location으로 데이터를 보낼 때 사용

DELETE : 특정 location의 데이터를 삭제할 때

- JSON은 경량(Lightweight)의 DATA-교환 형식
- Javascript에서 객체를 만들 때 사용하는 표현식을 의미
- JSON 표현식은 사람과 기계 모두 이해하기 쉬우며 용량이 작아서, 최근에는 JSON이 XML을 대체해서 데이터 전송 등에 많이 사용한다.
- 특정 언어에 종속되지 않으며, 대부분의 프로그래밍 언어에서 JSON 포맷의 데이터를 핸들링 할 수 있는 라이브러리를 제공한다.

```
    __v: 0
  },
  {
    "productImg": [
      "1534164044818_userfile"
    ],
    "productId": "9sSCz166W",
    "productName": "3",
    "productPrice": 2,
    "category": "책2호관",
    "productSelled": false,
    "updateDate": "2018-08-13T12:40:44.827Z",
    "__v": 0
  },
  {
    "productImg": [
      "1533797809756_1.PNG",
      "1533797809758_24a1542046c733bcf73ccd10e54c61e4.png"
    ],
    "productId": "b0KnrLPDz",
    "productName": "안대",
    "productPrice": 1232131,
    "category": "잡화기타",
    "productSelled": false,
    "updateDate": "2018-08-09T06:56:49.767Z",
    "__v": 0
  },
  {
    "productImg": [
      "1533626383384_1.PNG",
      "1533626383384_2.PNG"
    ],
    "productId": "cAX9XARFh",
    "productName": "note8",
    "productPrice": 654666,
    "productSelled": false,
    "category": "가전가구스마트폰",
    "updateDate": "2018-08-07T07:19:43.391Z",
    "__v": 0
  }
]
```

Aamofire

- 연결가능한(*Chainable*) Request/Response 메서드
- URL / JSON / plist 파라미터 인코딩
- 파일 / 데이터 / 스트림 / 멀티파트 폼 데이터 업로드
- Request 또는 Resume 데이터를 활용한 다운로드
- NSURLCredential을 통한 인증(*Authentication*)
- HTTP 리스폰스 검증(*Validation*)
- TLS 인증서와 공개 키 Pinning
- 진행 상태 클로저와 NSProgress
- cURL 디버깅 출력
- 광범위한 단위 테스트 보장
- 완벽한 문서화

Kingfisher

- 비동기 이미지 다운로드 및 캐싱
- URLSession 기반 네트워킹. 기본 이미지 프로세서 및 필터가 제공
- 메모리와 디스크를 위한 다중 계층 캐시 제공.
- 성능을 향상시키기 위해서 다운로드 작업을 취소하거나 처리할 수 있는 작업
- 필요에 따라 다운로더, 캐싱 시스템을 별도로 사용
- UIImageView, UIImage 및 UIButton에 대한 확장으로 URL에서 이미지를 직접 설정가능
- 이미지를 설정 시 내장된 애니메이션 사용가능.
- 확장 가능한 이미지 처리(Round, Blur등) 및 GIF이미지 형식 지원

```
Alamofire.request("\(self.appDelegate.serverURL)readBanner", method: .post, parameters:
    nil, headers: nil).responseJSON{ res in
    switch res.result{
    case .success(let item):
        //서버통신 성공시 내용
        print(item)
        break
    case .failure(let error):
        // 서버통신 실패시 내용
        print(error)
        break
    }
}
```