

# Table of Contents

1. API 개요	1
2. API 명세	2
3. API 사용법	5

v1.0.0, 2023.04.20

## 1. API 개요

Table 1. response-fields

Path	Type	Description
commentId	Number	댓글 ID
memberId	Number	회원 ID
body	String	댓글 내용
createdAt	String	생성 시간

response-body

```
{"commentId":1,"memberId":1,"body":"댓글 내용", "createdAt":"2023-04-26T13:20:04.379952"}
```

request-body

```
{"body":"댓글 내용"}
```

Table 2. request-headers

Name	Description
Authorization	Bearer (accessToken)

Table 3. request-fields

Path	Type	Description
body	String	댓글 내용

http-request

```
POST /board/answers/1/comments HTTP/1.1
Content-Type: application/json;charset=UTF-8
Authorization: Bearer (accessToken)
```

```
Accept: application/json
Content-Length: 53
Host: localhost:8080
```

```
{"body":"\u0000 \u0000 \u0000 \u0000\u0000"}
```

#### *httpie-request*

```
$ echo '{"body":"\u0000 \u0000 \u0000 \u0000\u0000"}' | http POST
'http://localhost:8080/board/answers/1/comments' \
'Content-Type:application/json;charset=UTF-8' \
'Authorization:Bearer (accessToken)' \
'Accept:application/json'
```

#### *curl-request*

```
$ curl 'http://localhost:8080/board/answers/1/comments' -i -X POST \
-H 'Content-Type: application/json;charset=UTF-8' \
-H 'Authorization: Bearer (accessToken)' \
-H 'Accept: application/json' \
-d '{"body":"\u0000 \u0000 \u0000 \u0000\u0000"}'
```

#### *http-response*

```
HTTP/1.1 201 Created
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers
Content-Type: application/json;charset=UTF-8
X-Content-Type-Options: nosniff
X-XSS-Protection: 1; mode=block
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
X-Frame-Options: SAMEORIGIN
Content-Length: 121

{"commentId":1,"memberId":1,"body":"\u0000 \u0000 \u0000 \u0000\u0000","createdAt":"2023-04-26T13:20:04.379952"}
```

Table 4. */board/answers/{answer-id}/comments*

Parameter	Description
<code>answer-id</code>	\u0000 \u0000

## 2. \u0000 \u0000

Table 5. *response-fields*

Path	Type	Description
commentId	Number	댓글 번호
memberId	Number	회원 번호
body	String	댓글 내용
createdAt	String	등록 시간

#### curl-request

```
$ curl 'http://localhost:8080/board/comments/1' -i -X PATCH \
  -H 'Content-Type: application/json;charset=UTF-8' \
  -H 'Authorization: Bearer (accessToken)' \
  -H 'Accept: application/json' \
  -d '{
  "commentId" : 1,
  "body" : "댓글 내용입니다"
}'
```

Table 6. /board/comments/{comment-id}

Parameter	Description
comment-id	댓글 번호

#### response-body

```
{
  "commentId" : 1,
  "memberId" : 1,
  "body" : "댓글 내용입니다",
  "createdAt" : "2023-04-26T13:20:04.475077"
}
```

#### http-response

```
HTTP/1.1 200 OK
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers
Content-Type: application/json;charset=UTF-8
X-Content-Type-Options: nosniff
X-XSS-Protection: 1; mode=block
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
X-Frame-Options: SAMEORIGIN
Content-Length: 131

{
  "commentId" : 1,
```

```

"memberId" : 1,
"body" : "ㅋㅋ ㅋㅋㅋㅋ",
"createdAt" : "2023-04-26T13:20:04.475077"
}

```

#### http-request

```

PATCH /board/comments/1 HTTP/1.1
Content-Type: application/json;charset=UTF-8
Authorization: Bearer (accessToken)
Accept: application/json
Content-Length: 67
Host: localhost:8080

{
  "commentId" : 1,
  "body" : "ㅋㅋ ㅋㅋㅋㅋ"
}

```

Table 7. request-fields

Path	Type	Description
commentId	Number	댓글 번호
body	String	댓글 내용

Table 8. request-headers

Name	Description
Authorization	Bearer (accessToken)

#### request-body

```

{
  "commentId" : 1,
  "body" : "ㅋㅋ ㅋㅋㅋㅋ"
}

```

#### httpie-request

```

$ echo '{
  "commentId" : 1,
  "body" : "ㅋㅋ ㅋㅋㅋㅋ"
}' | http PATCH 'http://localhost:8080/board/comments/1' \
  'Content-Type:application/json;charset=UTF-8' \
  'Authorization:Bearer (accessToken)' \
  'Accept:application/json'

```

### 3. □□ □□

*response-body*

*request-body*

Table 9. /board/comments/{comment-id}

Parameter	Description
comment-id	□□ □□□

*curl-request*

```
$ curl 'http://localhost:8080/board/comments/1' -i -X DELETE \  
-H 'Content-Type: application/json;charset=UTF-8' \  
-H 'Authorization: Bearer (accessToken)' \  
-H 'Accept: application/json'
```

Table 10. request-headers

Name	Description
Authorization	Bearer (accessToken)

*httpie-request*

```
$ http DELETE 'http://localhost:8080/board/comments/1' \  
'Content-Type:application/json;charset=UTF-8' \  
'Authorization:Bearer (accessToken)' \  
'Accept:application/json'
```

*http-request*

```
DELETE /board/comments/1 HTTP/1.1  
Content-Type: application/json;charset=UTF-8  
Authorization: Bearer (accessToken)  
Accept: application/json  
Host: localhost:8080
```

*http-response*

```
HTTP/1.1 204 No Content  
Vary: Origin  
Vary: Access-Control-Request-Method  
Vary: Access-Control-Request-Headers  
X-Content-Type-Options: nosniff
```

```
X-XSS-Protection: 1; mode=block  
Cache-Control: no-cache, no-store, max-age=0, must-revalidate  
Pragma: no-cache  
Expires: 0  
X-Frame-Options: SAMEORIGIN
```