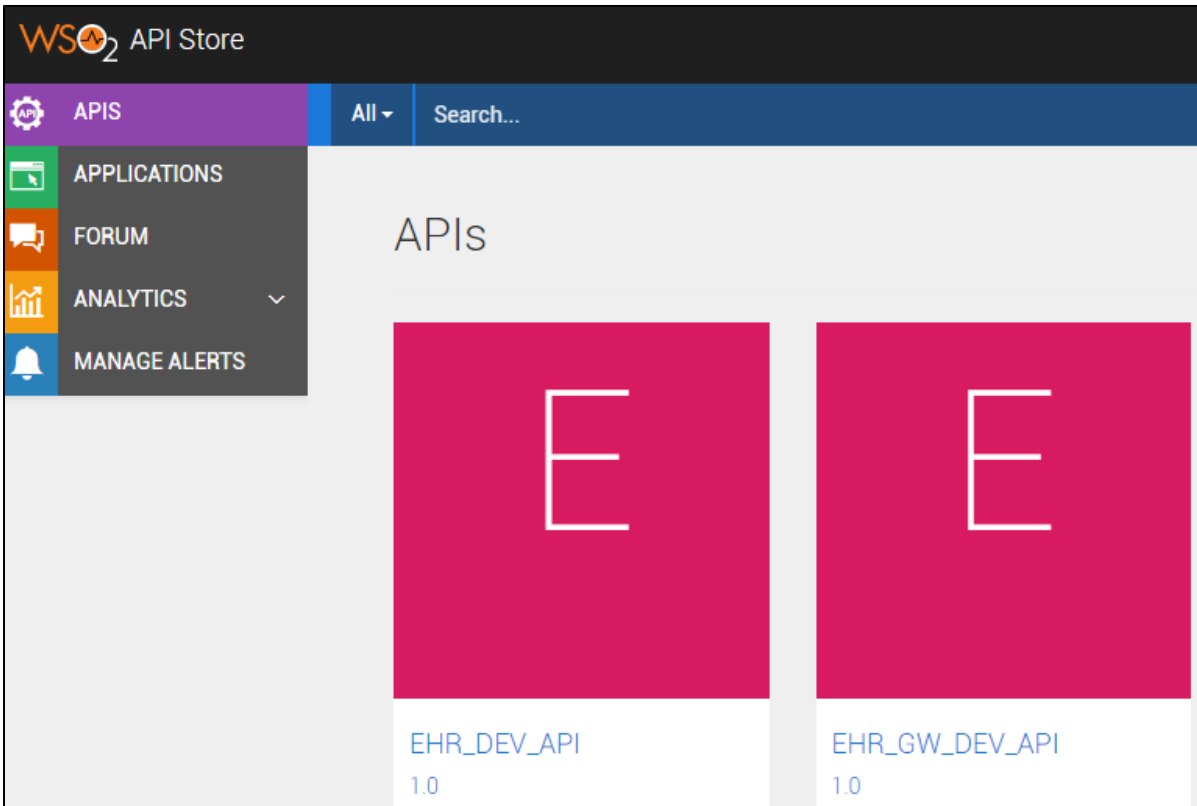


# EHR API 이용 가이드

## 1. EHR APIM Store 서버 접속

- <https://APIM주소:9443/store/>

APIM Store 주소는 병원 별로 제공되는 주소를 참조한다.



병원 별로 방화벽을 오픈해야하는 경우가 있으므로 사이트 접속이 안되는 경우 인터페이스 개발자에게 확인을 요청한다.

## 2. 접속 테스트

### 1) API호출을 위한 토큰 발급

- 테스트를 위해서 개발용 토큰을 발급하기 위해 [인증키](#)를 요청한다. (인터페이스 개발자에게 요청)

- 인증키를 이용해서 토큰을 발급받는다.

```
curl -k -d "grant_type=client_credentials" -H "Authorization: Basic 인증키" https://APIM주소:8243/token
```

토큰을 요청하면 Json 형식의 응답 메시지가 출력된다. access\_token 값을 읽어서 토큰을 가져온다.

```
{"access_token":"토큰","scope":"default","token_type":"Bearer","expires_in":남은유효시간(초)}
```

※ 토큰 발급 테스트를 위해서 CMD 창에서 실행한 예제

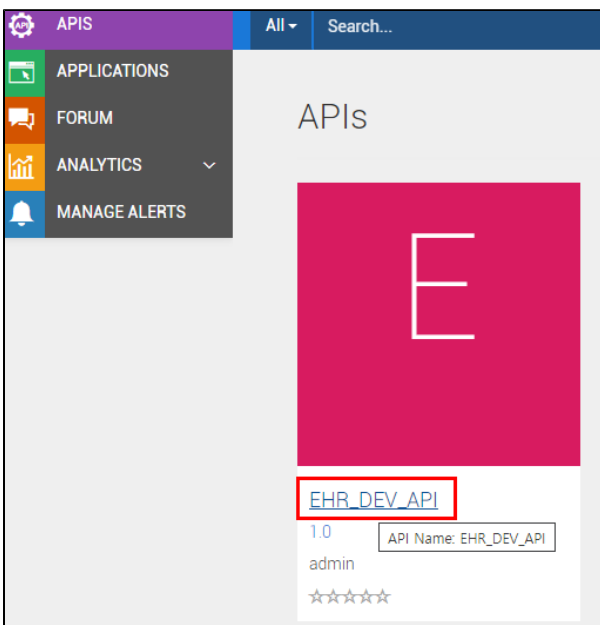
```
선택 명령 프롬프트
Microsoft Windows [Version 10.0.18363.836]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\WADSF>curl -k -d "grant_type=client_credentials" -H "Authorization: Basic UGJSMDAObGh1MGNNY21hTHFOWWdVZHg1VFZZYT
pTeV9iME9iSnZRbWdpNzJDWUFZR3VsWnZKS1lh" https://112.107.145.207:8243/token
{"access_token":"b9e8f954-51d7-3887-9a21-85f38345c0a0","scope":"am_application_scope default","token_type":"Bearer","exp
ires_in":3338}
C:\Users\WADSF>
```

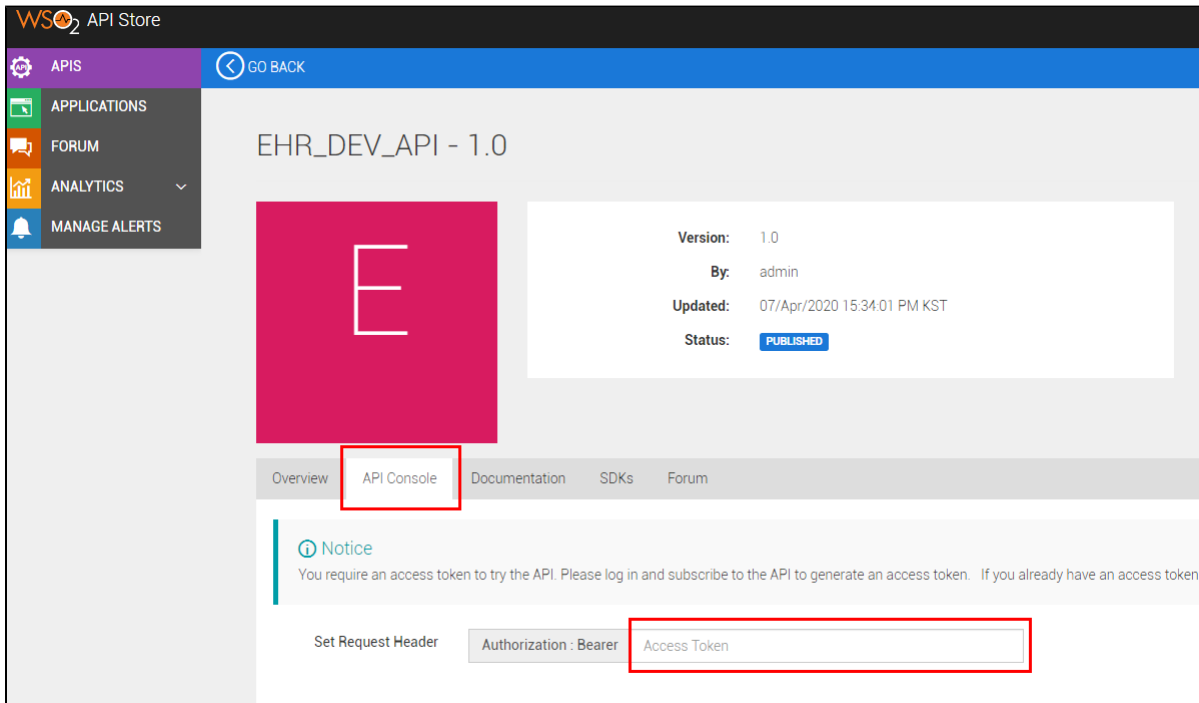
## 2) 토큰을 이용한 테스트 (Store 서버 이용)

store 사이트에 접속 후 이용하려는 API를 클릭한다. (크롬 브라우저를 이용)

여기서는 EHR 개발 API인 EHR\_DEV\_API 를 선택한다.

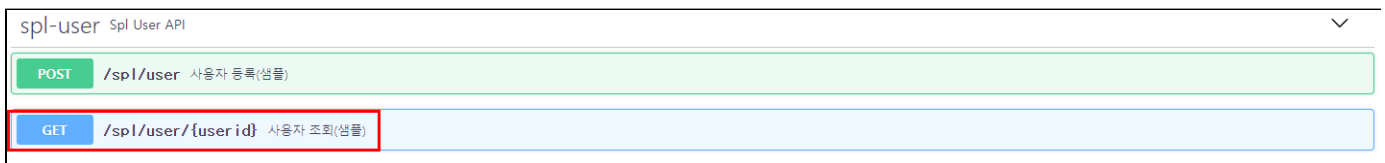


API Console 탭을 클릭한 후 토큰을 아래의 Access Token 입력창에 입력한다.

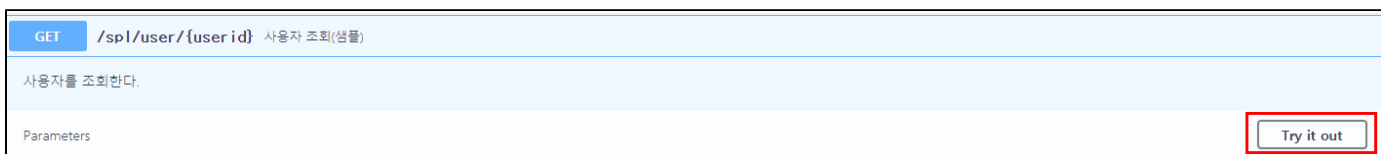


정상적으로 서비스가 가능한지 여부를 확인하기 위해서 샘플 API인 /spl/user/{userid} 사용자 조회(샘플)를 선택한다.

사용하려는 API가 존재한다면 해당 API를 이용해서 테스트를 할 수 있다.



Try it out 버튼을 클릭한다.



userid에 test를 입력한 후 Execute 버튼을 클릭한다.

The screenshot shows a REST client interface with a 'Parameters' tab. It contains a table with two columns: 'Name' and 'Description'. The first row has 'userid' (marked as required) in the Name column and '사용자 아이디' in the Description column. Below the table, the 'userid' parameter is set to the string value 'test'. At the bottom of the interface, there is a blue bar with an 'Execute' button.

성공한 경우 Response Body에 응답 데이터가 조회된다.

The screenshot shows the 'Responses' tab of a REST client. It displays a 200 status code. The 'Response body' is shown as a JSON object: 

```
{  "userList": [    {      "userId": "test",      "userName": "Kim",      "userAge": "20",      "userGender": "F"    }  ]}
```

. Below the response body, the 'Response headers' are listed, showing 'content-type: application/json; charset=UTF-8'. A 'Download' button is visible next to the response body.

토큰이 유효하지 않는 경우 아래와 같이 에러 메시지가 출력된다.

-> 재요청해서 유효한 토큰을 재발급 받는다.

The screenshot shows the 'Server response' tab of a REST client. It displays a 401 status code with the message 'Error: Unauthorized'. The 'Response body' is shown as a JSON object: 

```
{  "fault": {    "code": 900902,    "message": "Missing Credentials",    "description": "Required OAuth credentials not provided. Make sure your API invocation call has a header: <code>Authorization: Bearer ACCESS_TOKEN</code>"  }}
```

. Below the response body, the 'Response headers' are listed, showing 'content-type: application/json; charset=UTF-8'.

또한, EHR 서버에서 에러가 발생한 경우에는 500 에러가 발생하거나 기타 다른 에러가 발생할 수 있다.

-> 인터페이스 개발자에게 확인을 요청한다.

### 3) 토큰을 이용한 테스트 (RestClient 툴 이용)

RestClient 툴을 이용하여 테스트하는 방법을 설명한다.

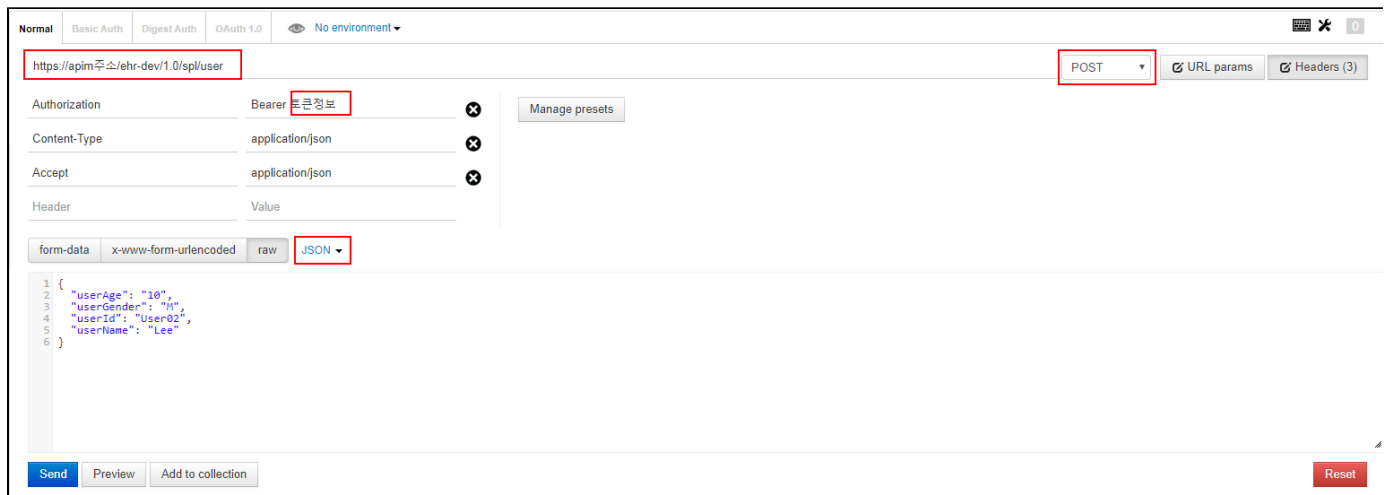
여기서는 크롬 앱인 Tabbed Postman - REST Client를 이용한 방법을 설명한다. (다른 프로그램을 사용해도 무관함)

해당 프로그램은 크롬 웹 스토어에서 설치가 가능하며 [https://chrome.google.com/webstore/category/extensions?utm\\_source=chrome-ntp-icon](https://chrome.google.com/webstore/category/extensions?utm_source=chrome-ntp-icon)

해당 주소 접속 후 검색 항목에 Tabbed Postman 입력해서 설치할 수 있다.

Header에 토큰 및 Accept, Content-Type에 application/json을 입력한 후 알맞은 Method(GET, POST, PUT, DELETE)를 지정하여 API를 호출한다.

/spl/user - POST Method API 호출 예)



## 3. API 호출 방법

인터페이스 장비 및 시스템에서 API를 호출하는 방법을 설명한다.

REST API를 이용하는 방식으로 http, https 프로토콜 모두 가능하도록 개발하고 운영에서는 <https> 프로토콜을 사용한다.

### 1) 토큰을 이용한 API 호출

토큰을 이용하여 해당 API를 호출한다.

- 호출 예)

```
curl -k -X GET "https://API주소:8243/xxx/..." -H "accept: application/json" -H "Content-Type: application/json" -H "Authorization: Bearer 토큰"
```

```
curl -k -X POST "https://API주소:8243/xxx/..." -H "accept: application/json" -H "content-type: application/json" -H "Authorization: Bearer 토큰"
```

```
curl -k -X PUT "https://API주소:8243/xxx/..." -H "accept: application/json" -H "content-type: application/json" -H "Authorization: Bearer 토큰"
```

```
curl -k -X DELETE "https://API주소:8243/xxx/..." -H "accept: application/json" -H "content-type: application/json" -H "Authorization: Bearer 토큰"
```

토큰 유효하지 않아 권한이 없는 경우 Response.HttpCode는 401이 출력되고 아래와 같은 메시지가 출력된다. (해당 메시지의 에러코드가 900901 or 900902 인 경우)

#### 토큰이 유효하지 않은 경우

```
{
  "fault": {
    "code": 900901,
    "message": "Invalid Credentials",
    "description": "Access failure for API: /xxx/1.0, version: 1.0 status: (900901) - Invalid Credentials. Make sure you have given the correct access token"
  }
}
```

#### 토큰을 입력하지 않은 경우

```
{
  "fault": {
    "code": 900902,
    "message": "Missing Credentials",
    "description": "Required OAuth credentials not provided. Make sure your API invocation call has a header: \"Authorization: Bearer ACCESS_TOKEN\""
  }
}
```

Response.HttpCode가 401이 출력된 경우 인증키를 통한 토큰 발급을 신규로 요청한다.

```
curl -k -d "grant_type=client_credentials" -H "Authorization: Basic 인증키" https://APIM주소:8243/token
```

인증키가 유효하지 않는 경우에는 아래와 같은 메시지가 출력된다.

아래 메시지가 출력된 경우에는 인터페이스 개발자에게 인증키를 확인/재발급 받는다.

#### 인증키가 유효하지 않은 경우

```
{"error": "invalid_client", "error_description": "Client Authentication failed."}
```

정상적으로 토큰이 발급된 경우에는 아래와 같은 메시지가 출력된다.

```
{"access_token": "", "scope": "default", "token_type": "Bearer", "expires_in": ()}
```

access\_token 값을 읽어서 API호출을 위한 토큰을 가져온 후 다시 API를 요청한다.

재발급 받은 토큰과 함께 API 호출시 토큰 에러가 계속적으로 발생할 수 있으므로 재시도 횟수를 제한한다.

재시도 횟수를 초과하는 경우에는 인터페이스 개발자에게 확인을 요청한다.