

# SW 활용도 API 가이드 문서

버전	변경일	변경사유	변경내역
1.0	2024-01-23	최초작성	최초작성

Base Url: {-}

---

## 데이터 공통 응답형식

JSON 포맷의 데이터이며, 필드명은 소문자로 시작하는 camel notation을 사용.

다음 json object type으로 응답

```
{
  "code": 200,
  "message": "OK",
  "data": {
    ...
  }
}
```

Name	Type	Description	Mandatory	Note
code	Number	응답 코드	O	
message	String	응답 메시지	O	
data	-	응답 데이터	X	응답 데이터 형식에 따른 데이터 타입

## 통계 API

### | 월별 접속자 수

Description:

요청한 기간의 월별 평균 접속자 수 제공

GET

Endpoint:

/v1/access?begin\_month={YYYY-MM}&end\_month={YYYY-MM}

Request Parameters:

Name	Type	Description	Mandatory	Note
begin_month	String	요청할 첫번째 년.월	0	- YYYY-MM 형식
end_month	String	요청할 마지막 년.월	0	- YYYY-MM 형식

Response Body:

```
{
  "code": 200,
  "message": "OK",
  "data": {
    "result": [
      {
        "month": 2023-12,
        "accessor": 1937
      },
      {
        "month": 2024-01,
        "accessor": 1781
      }
    ]
  }
}
```

Name	Depth1	Type	Description	Mandatory	Note
result		Array		0	
	month	String	년.월	0	- YYYY-MM 형식

	accessor	Number	접속한 사용자 수	X	- DB 에러로 인 해 데이터 값이 없을 수 있음
--	----------	--------	-----------	---	-----------------------------------

## | 일자별 접속자 수

Description:

요청한 기간의 일별 평균 접속자 수 제공

Method:

GET

Endpoint:

/v1/access? begin\_day={YYYY-MM-DD}&end\_day={YYYY-MM-DD }

Request Parameters:

Name	Type	Description	Mandatory	Note
begin_day	String	요청할 첫번째 일자	0	- YYYY-MM-DD 형식
end_day	String	요청할 마지막 일자	0	- YYYY-MM-DD 형식

Response Body:

```
{
  "code": 200,
  "message": "OK",
  "data": {
    "result": [
      {
        "day": 2023-12-01,
        "accessor": 60
      },
      {
        "day ": 2024-12-02,
        "accessor": 70
      }
    ]
  }
}
```

Name	Depth1	Type	Description	Mandatory	Note
result		Array		0	-
	day	String	년.월.일	0	- YYYY-MM-DD 형식

	accessor	Number	접속한 사용자 수	X	- DB 에러로 인 해 데이터 값이 없을 수 있음
--	----------	--------	-----------	---	-----------------------------------

## | 평균 하루 로그인 수

Description:

요청한 일자의 평균 로그인 수 제공

Method:

GET

Endpoint:

/v1/login/{day}

Request Parameters:

Name	Type	Description	Mandatory	Note
day	String	요청 일자	O	- YYYY-MM-DD 형식

Response Body:

```
{
  "code": 200,
  "message": "OK",
  "data": {
    "loginAverage": 30
  }
}
```

Name	Type	Description	Mandatory	Note
loginAverage	Number	해당 일자의 평균 로그인 수	X	- 데이터가 없다면 Null 반환 - 데이터가 있는데 평균이 0 이라면 0 반환

## | 휴일을 제외한 로그인 수

Description:

요청한 기간에서 휴일을 제외한 일별 로그인 수 제공

Method:

GET

Endpoint:

/v1/login?begin\_day={YYYY-MM-DD}&end\_day={YYYY-MM-DD }

Request Parameters:

Name	Type	Description	Mandatory	Note
begin_day	String	요청할 첫번째 일자	0	- YYYY-MM-DD 형식
end_day	String	요청할 마지막 일자	0	- YYYY-MM-DD 형식

Response Body:

```
{
  "code": 200,
  "message": "OK",
  "data": {
    "result": [
      {
        "day": 2023-12-01,
        "login": 60
      },
      {
        "day ": 2024-12-02,
        "login": 70
      }
    ]
  }
}
```

Name	Depth1	Type	Description	Mandatory	Note
result		Array		0	-
	day	String	년.월.일	0	- YYYY-MM-DD 형식



	login	Number	로그인한 사용자 수	X	- DB 에러로 인해 데이터 값이 없을 수 있음
--	-------	--------	------------	---	----------------------------

### 질문1)

Quert Parameter로 받아야 하는 값들은 모두 일자별 접속자 수 api와 똑같은데 어떻게 구분을 하면 좋을지 고민하다가 질문이 생겼습니다.

각 api가 통계 api로 묶인다고 생각해서 처음에는 endpoint를 v1/average 로 만들었습니다. 그러니 로그인 수와 접속자 수에 대한 구분을 하는 것을 어떻게 해야 할지 이해가 잘 가지 않는 부분이 생겼던 것 같습니다.

로그인 수에 대한 통계가 접속자 수에 대한 통계와 같은 데이터에서 나올 수 있는 부분이 아니라고 생각했습니다.(로그인을 하지 않더라도 접속이 가능하다는 전제 하에 그렇게 생각했습니다.)

따라서 endpoint를 아예 v1/login과 v1/access 로 나누어 진행을 하는 게 가장 좋은 방법이라고 생각하여 이렇게 진행했는데 이렇게 진행하는 것이 최선의 방법일지 여쭙보고 싶습니다.

그외로 제가 생각했던 방법은

#### 1) Query Parameter 추가

경우1) 로그인 수, 접속자 수를 구분해서 받는 Query

- ➔ 비교하는 방안을 그저 int값으로 받으면 코드 이해가 어려울 것이고, string으로 받으면 비교할 때 복잡도가 올라갈 것이라고 생각했습니다.
- ➔ 변경이 없을 값을 굳이 필드로 만들어 추가로 받는 것은 낭비라고 생각했습니다.

경우2) holiday에 대한 true로 받는 Query 추가

- ➔ 변경이 없을 값을 굳이 필드로 만들어 추가로 받는 것은 낭비라고 생각했습니다.

#### 2) Path Parameter로 각 api의 경로를 v1/average/login, v1/average/access 로 나누기

- ➔ 불필요한 파라미터를 URI Path에 추가하는 게 복잡함 유지, 리소스 사이의 관계 변화로 인한 유연성이 떨어진다는 점 때문에 좋지 않다는 원칙이 있어 최대한 줄이고자 했습니다.

### 질문2)

휴일인 경우와, 진짜 로그인 수가 없어서 0인 경우를 구분하기는 어렵다고 생각을 하게 되었습니다. 이를 구분하기 위해 따로 필드를 만들어 휴일임을 알려주는 것이 필요할지, 휴일인 경우에는 아예 제외해도 괜찮을지 클라이언트에게 여쭙보고 싶습니다.

## | 부서별 월별 로그인 수

Description:

요청한 기간의 부서별, 월별 로그인 수 제공

Method:

GET

Endpoint:

/v1/login?part={a,b,c}&begin\_month={YYYY-MM}&end\_month={YYYY-MM}

Request Parameters:

Name	Type	Description	Mandatory	Note
part	Array	요청할 부서들	X	- 해당 부분의 값이 없다면 전체 부서의 값 전달
begin_month	String	요청할 첫번째 년.월	O	- YYYY-MM 형식
end_month	String	요청할 마지막 년.월	O	- YYYY-MM 형식

Response Body:

```
{
  "code": 200,
  "message": "OK",
  "data": {
    "result": [
      {
        "part": "frontend",
        "list": [
          {
            "month": 2023-12,
            "login": 687
          },
          {
            "month": 2024-01,
            "login": 378
          }
        ]
      },
      {

```

```
    "part": "backend",
    "list": [
      {
        "month": 2023-12,
        "login": 456
      },
      {
        "month": 2024-01,
        "login": 243
      }
    ]
  }
]
```

Name	Depth1	Depth2	Type	Description	Mandatory	Note
result			Array		0	
	part		String	부서명	0	
	list		Array	월별 로그인 사용자 수	0	
		day	String	년.월.일	0	- YYYY-MM-DD 형식
		login	Number	로그인한 사용자 수	X	- DB 에러로 인해 데이터 값이 없을 수 있음

질문1)

get은 서버에서 데이터를 가져오고, post는 서버나 데이터 상태를 바꾸는 데 사용되지만, **쿼리로 하면 너무 복잡해질 경우에도 post를 통해 통신하는 것보다 목적에 맞게 get으로 통신하는 게 좋은지 궁금합니다.** 현재 Query Parameter로 부서 정보를 받고 있는데, 부서가 너무 많아질 경우 너무 url이 복잡해지는 문제 발생할 수 있을 것 같아 궁금했습니다.

---

질문)

시간의 값(예시 데이터: 21년 7월 16일 12시에 A 사용자 로그인)도 저장된다고 나와있는데 이는 어디에서 사용이 되는건지 궁금했습니다. 수만 응답하거나, 일별, 월별로 응답을 주는 것이라 시간까지 고려를 안 하는 것이라고 생각했습니다. 현재 api에서는 시간에 대해 사용해야 하는 부분이 없다고 판단했는데 이게 맞는지 궁금합니다.

질문)

로그인 수에 대한 데이터도 있는지 궁금합니다. 로그인 수와 접속자 수가 비례하지 않는 서비스라고 가정하고 api를 짜다보니 접속자 수에 대한 데이터는 어떤 형태로 저장되어 있는지 궁금했습니다.