

# 조명 제어 API

*version 1.2b*

2023/12/05

## 목 차 (Table of Contents)

<b>1.</b>	개요 .....	<b>4</b>
1.1	개념 .....	5
1.1.1	ID .....	5
<b>2.</b>	<b>공통 .....</b>	<b>6</b>
2.1	기본 형식 .....	6
2.1.1	Base URL .....	6
2.1.2	Header .....	6
2.1.3	응답 메시지 .....	7
2.2	API .....	8
2.2.1	조명 제어 맵 .....	8
2.2.2	조명 모듈 구성 .....	12
2.2.3	조명 모듈 설정 .....	16
2.2.4	OTA (펌웨어 관리) .....	21

## ■ 문서 이력

버전	일자	작성자	문서 변경 내용
1.0	2023.10	Gil	최초 작성
1.1	2023.10	Gil	OTA 파일 삭제 추가
1.2	2023.11	Gil	OTA 취소 추가

## 1. 개요

주차 통합 시스템의 관리자 프로그램에서 조명 제어 API 를 정리한다.

## 1.1 개념

### 1.1.1 ID

API에서 사용하는 ID는 크게 3가지가 있다.

CCTV ID와 조명 모듈 ID, NID이다.

#### CCTV ID

CCTV ID는 주차장 번호 + 주차장 층 + 주차장내 구역 + CCTV 번호를 포함한다.

Naming 규칙은 아래와 같이 (underline)를 4등분되어 있다.

P<n>\_B<n>\_<n>\_<n>

예)

P1\_B2\_1\_1 (1주차장 지하 2층 1구역 CCTV #1)

#### 조명 모듈 ID

조명 모듈 ID는 CCTV 내에서 유일한 ID이며, 1~128까지 사용 가능하다.

128은 Root 모듈로 고정되어 있으며, 나머지 번호는 임의로 설정한다.

#### NID

NID는 노드 ID로 조명 모듈을 전체 리스트에서 명명할 때 사용한다.

따라서 CCTV와 조명 모듈 타입, 아이디를 포함한다.

Naming 규칙은 아래와 같다.

<CCTV\_ID>@<Type>.<ID>

Type A(통로등), P(만공차등), C(충돌방지등), R(루트로드), L(경광등)

예)

<a href="#">P1_B2_1_1@A.1</a>	CCTV P1_B2_1_1에 연결된 통로등 타입의 조명 모듈 아이디 1
<a href="#">P1_B2_1_1@P.11</a>	CCTV P1_B2_1_1에 연결된 만공차등 타입의 조명 모듈 아이디 11
<a href="#">P1_B3_1_1@C.21</a>	CCTV P1_B3_1_1에 연결된 충돌감지등 타입의 조명 모듈 아이디 21
<a href="#">P1_B3_1_1@R.128</a>	CCTV P1_B3_1_1에 연결된 루트로드 타입의 조명 모듈 아이디 128

## 2. 공통

### 2.1 기본 형식

#### 2.1.1 Base URL

[http://<server\\_ip>:8082/api](http://<server_ip>:8082/api)

ex) <http://192.168.3.207:8082/api>

#### 2.1.2 Header

API 헤더는 기본적으로 아래 형식을 사용한다.

```
Content-Type: application/json
accept: application/json
Authorization: c0f0de79-55f3-419b-88ea-6dde435acb35
```

\* Authorization API Key: c0f0de79-55f3-419b-88ea-6dde435acb35

### 2.1.3 응답 메시지

응답은 json 형식을 사용

성공 시)

```
{  
  "result": "success",  
  "data": {}  
}
```

실패 시)

```
{  
  "result": "fail",  
  "errCode": 100,  
  "errMsg": "Invalid Parameter"  
}
```

\* errCode 정의 필요

## 2.2 API

### 2.2.1 조명 제어 맵

#### 2.2.1.1 맵 조회

Request)

Method	GET		
URL	/lights		
Parameters	location	String	주차장 위치/ 층 단위 예) "P1_B2": 1 주차장 지하 2 층
Description	주차장의 각층 조명 상태를 일괄 조회한다.		

Response) – JSON

항목	타입	설명	예
nid	String	조명 노드 아이디	P1_B2_1_1@A.1
type	String	조명 타입 aisle(통로등) parking(만공차등), caution(충돌방지등), lamp(경광등)	"aisle"
value	String	조명 값 "20", "80" - 통로등(디밍) "0", "100" - 그외 (0=off,100=on)	"80"
name	String	조명 이름	"1 구역-지하 2 층-통로등#1"

Request Example)

GET /lights?location=P1\_B3

Response Example)

```
{
  "result": "success",
  "data": [
    {
      "nid": "P1_B3_1_1@A.1",
      "type": "aisle",
      "value": "20",
      "name": "1 구역-지하 3 층-통로등#1"
    },
    {
      "nid": "P1_B3_1_1@A.2",
      "type": "aisle",
      "value": "20",
      "name": "1 구역-지하 3 층-통로등#2"
    },
    {
      "nid": "P1_B3_1_1@P.11",
      "type": "parking",
      "value": "0",
      "name": "1 구역-지하 3 층-만공차등#1"
    },
    {
      "nid": "P1_B3_1_1@P.12",
      "type": "parking",
      "value": "0",
      "name": "1 구역-지하 3 층-만공차등#2"
    },
    {
      "nid": "P1_B3_1_1@C.21",
      "type": "caution",
      "value": "100",
      "name": "1 구역-지하 3 층-충돌감지등#1"
    },
    {
      "nid": "P1_B3_1_1@R.128",
      "type": "lamp",
      "value": "100",
      "name": "1 구역-지하 3 층-경광등"
    }
  ]
}
```

## 2.2.1.2 일괄 조명 제어

Request)

Method	POST
URL	/lights/{location}
Description	주차장의 각층 조명 상태를 일괄 제어한다.

Body)

항목	타입	설명	예
location	String	주차장 위치 /층 단위	예) "P1_B3" : 1 주차장 지하 3 층
action	String	조명 설정 emergency manual	"emergency"
value	String	조명 설정 값 { "on", "off" }	"on"

Response) – JSON

기본 형식

Request Example)

POST /lights/P1\_B3

```
{  
  "action": "manual",  
  "value": "on"  
}
```

1 주차장 지하 2 층 모든 조명을 켜다.

Response Example)

```
{  
  "result": "success"  
}
```

Request Example)

POST /lights/P1\_B3

```
{  
  "action": "emergency",  
  "value": "on"  
}
```

1 주차장 지하 2 층 모든 조명이 깜빡임으로 동작한다.

Response Example)

```
{  
  "result": "success"  
}
```

## 2.2.2 조명 모듈 구성

### 2.2.2.1 모듈 조회

Request)

Method	GET		
URL	/nodes		
Parameters	cctvId	String	CCTV ID 예) "P1_B2_1_1"
Description	CCTV 에 연결된 노드 리스트를 조회한다.		

Response) – JSON

항목	타입	설명	예
id	Integer	조명 모듈 ID (1..128)	2
macAddr	String	조명 모듈 주소	"62:3E:79:12:F1:4C"
type	String	조명 타입 root(루트노드) aisle(통로등) parking(만공차등), caution(충돌방지등), lamp(경광등)	"parking"
name	String	조명 모듈 이름	"1 구역-지하 2 층-만공차#1"
description	String	조명 모듈 설명 (사용자 입력 정보)	"점검 필요"

Request Example)

GET /nodes?ctvId=P1\_B2\_1\_1

Response Example)

```
{
  "result": "success",
  "data": [
    {
      "id": 1,
      "macAddr": "62:3E:79:12:F1:41",
      "type": "aisle",
      "name": "1 구역-지하 2 층-통로등#1",
      "description": ""
    },
    {
      "id": 11,
      "macAddr": "62:3E:79:12:F1:43",
      "type": "parking",
      "name": "1 구역-지하 2 층-만공차등#1",
      "description": ""
    },
    {
      "id": 12,
      "macAddr": "62:3E:79:12:F1:44",
      "type": "parking",
      "name": "1 구역-지하 2 층-만공차등#2",
      "description": ""
    },
    {
      "id": 128,
      "macAddr": "62:3E:79:12:F1:46",
      "type": "root",
      "name": "1 구역-지하 2 층",
      "description": ""
    }
  ]
}
```

## 2.2.2.2 모듈 설정

Request)

Method	POST
URL	/nodes/{cctvId}
Description	CCTV 에 연결된 노드 구성을 설정한다.

Body)

항목	타입	설명	예
id	Integer	조명 모듈 ID (1..128)	2
macAddr	String	조명 모듈 주소	“62:3E:79:12:F1:4C”
name	String	조명 모듈 이름	“1 구역-지하 2 층-만공차#1”
description	String	조명 모듈 설명 (사용자 입력 정보)	“점검 필요”

Response) – JSON

기본 형식

## Request Example)

POST /nodes/P1\_B2\_1\_1

```
[
  {
    "id": 1,
    "macAddr": "62:3E:79:12:F1:41",
    "type": "aisle",
    "name": "1 구역-지하 2 층-통로등#1",
    "description": ""
  },
  {
    "id": 2,
    "macAddr": "62:3E:79:12:F1:42",
    "type": "aisle",
    "name": "1 구역-지하 2 층-통로등#2",
    "description": ""
  },
  {
    "id": 11,
    "macAddr": "62:3E:79:12:F1:43",
    "type": "parking",
    "name": "1 구역-지하 2 층-만공차등#1",
    "description": ""
  },
  {
    "id": 12,
    "macAddr": "62:3E:79:12:F1:44",
    "type": "parking",
    "name": "1 구역-지하 2 층-만공차등#2",
    "description": ""
  },
  {
    "id": 21,
    "macAddr": "62:3E:79:12:F1:45",
    "type": "caution",
    "name": "1 구역-지하 2 층-충돌감지등",
    "description": ""
  },
  {
    "id": 128,
    "macAddr": "62:3E:79:12:F1:46",
    "type": "root",
    "name": "1 구역-지하 2 층",
    "description": ""
  }
]
```

## Response Example)

```
{
  "result": "success"
}
```

## 2.2.3 조명 모듈 설정

### 2.2.3.1 통로등(디밍) 조회

Request)

Method	GET		
URL	/dimming		
Parameters	cctvId	String	CCTV ID 예) "P1_B3_1_1"
Description	CCTV 에 연결된 노드의 디밍값을 조회한다.		

Response) – JSON

항목	타입	설명	예
dimDefault	Integer	조명 기본 밝기 (0..100)	30
dimMin	Integer	조명 밝기 최소 (0..100)	20
dimMax	Integer	조명 밝기 최대	90

## Request Example)

GET /dimming?cctvId=P1\_B2\_1\_1

## Response Example)

```
{
  "result": "success",
  "data": {
    "dimDefault": 30,
    "dimMin": 20,
    "dimMax": 90
  }
}
```

## 2.2.3.2 통로등(디밍) 설정

Request)

Method	POST
URL	/dimming/{cctvId}
Description	CCTV 에 연결된 노드의 디밍값을 설정한다.

Body)

항목	타입	설명	예
dimDefault	Integer	조명 기본 밝기 (0..100)	30
dimMin	Integer	조명 밝기 최소 (0..100)	10
dimMax	Integer	조명 밝기 최대	80

Response) – JSON

기본 형식

Request Example)

POST /dimming/P1\_B2\_1\_1

```
{
  "dimDefault": 30,
  "dimMin": 20,
  "dimMax": 90
}
```

Response Example)

```
{
  "result": "success"
}
```

## 2.2.3.3 BLE 조회

Request)

Method	GET		
URL	/ble		
Parameters	cctvId	String	CCTV ID 예) "P1_B2_1_1"
Description	CCTV 에 연결된 노드의 BLE 파라미터를 조회한다.		

Response) – JSON

항목	타입	설명	예
advMin	Integer	Advertising 최소 (ms) (100..10000)	300
advMax	Integer	Advertising 최대 (ms) (100..10000)	500
scanInt	Integer	Scanning 간격 (ms) (100..10000)	250
scanWin	Integer	Scanning 창 (ms) (100..10000)	200

Request Example)

GET /ble?cctvId=P1\_B2\_1\_1

Response Example)

```
{
  "result": "success",
  "data": {
    "advMin": 300,
    "advMax": 500,
    "scanInt": 250,
    "scanWin": 200
  }
}
```

## 2.2.3.4 BLE 설정

Request)

Method	POST
URL	/ble
Description	CCTV 에 연결된 노드의 BLE 파라미터를 설정한다.

Body)

항목	타입	설명	예
advMin	Integer	Advertising 최소 (ms) (100..10000)	300
advMax	Integer	Advertising 최대 (ms) (100..10000)	500
scanInt	Integer	Scanning 간격 (ms) (100..10000)	250
scanWin	Integer	Scanning 창 (ms) (100..10000)	200

Response) – JSON

기본 형식

Request Example)

POST /dimming/P1\_B2\_1\_1

```
{
  "advMin": 330,
  "advMax": 640,
  "scanInt": 300,
  "scanWin": 250
}
```

Response Example)

```
{
  "result": "success"
}
```

## 2.2.4 OTA (펌웨어 관리)

### 2.2.4.1 펌웨어 파일 조회

Request)

Method	GET
URL	/ota-files
Description	OTA 파일 리스트를 조회한다.

Response) – JSON

항목	타입	설명	예
file	String	파일명	CCTV_MESH_v1.1.9.bin
version	String	펌웨어 버전	v1.1.9
size	Integer	펌웨어 파일 크기(bytes)	993712
type	String	펌웨어 파일 타입 { "CCTV_MESH" "CCTV_BLE" }	"CCTV_MESH"
uploaded	String	업로드 시간	"2023-10-05 18:58:30"

Request Example)

GET /ota-files

Response Example)

```
{
  "result": "success",
  "data": [
    {
      "file": "CCTV_MESH_v1.1.9.bin",
      "version": "v1.1.9",
      "size": 798640,
      "type": "CCTV_MESH",
      "uploaded": "2023-10-05 18:53:20"
    },
    {
      "file": "CCTV_BLE_v1.1.9.bin",
      "version": "v1.1.9",
      "size": 910480,
      "type": "CCTV_BLE",
      "uploaded": "2023-10-05 18:52:16"
    },
    {
      "file": "CCTV_MESH_v1.1.2.bin",
      "version": "v1.1.2",
      "size": 993712,
      "type": "CCTV_MESH",
      "uploaded": "2023-09-17 11:13:09"
    },
    {
      "file": "CCTV_BLE_v1.1.2.bin",
      "version": "v1.1.2",
      "size": 989360,
      "type": "CCTV_BLE",
      "uploaded": "2023-09-17 11:11:38"
    }
  ]
}
```

## 2.2.4.2 펌웨어 파일 삭제

Request)

Method	DELETE
URL	/ota-files
Description	OTA 파일을 삭제한다.

Body)

항목	타입	설명	예
file	String	파일명	CCTV_MESH_v1.1.10.bin

Response) – JSON

기본 형식

Request Example)

DELETE /ota-files

```
{  
  "file": "CCTV_MESH_v1.1.9.bin"  
}
```

Response Example)

```
{  
  "result": "success"  
}
```

## 2.2.4.1 펌웨어 파일 업로드

Request)

Method	POST
URL	/ota-upload
Description	OTA 파일을 업로드한다.

Body)

항목	타입	설명	예
file	String	파일명	CCTV_MESH_v1.1.10.bin

Response) – JSON

항목	타입	설명	예
file	String	파일명	CCTV_MESH_v1.1.9.bin
version	String	펌웨어 버전	v1.1.9
type	String	펌웨어 파일 타입 { "CCTV_MESH" "CCTV_BLE" }	"CCTV_MESH"
uploaded	String	업로드 시간	"2023-10-05 18:58:30"

Request Example)

POST /ota-upload

```
{
  "file": "CCTV_MESH_v1.1.9.bin"
}
```

Response Example)

```
{
  "result": "success",
  "data": {
    "file": "CCTV_MESH_v1.1.9.bin",
    "version": "v1.1.9",
    "type": "CCTV_MESH",
    "uploaded": "2023-10-05 18:53:20"
  }
}
```

## 2.2.4.1 펌웨어 진행 상태 조회

Request)

Method	GET	
URL	/ota	
Parameters	cctvId	String
Description	OTA 진행상태를 조회한다.	

Response) – JSON

항목	타입	설명	예
nid	String	노드 ID	P1_B2_1_1@A.1
id	Integer	조명 모듈 ID	1
file	String	파일명	CCTV_MESH_v1.1.10.bin
mode	String	업그레이드 모드 "single", "multi"	"single"
progress	Integer	진행률 (0~100 %)	30
result	String	결과 <sup>1)</sup> { "idle", "ready", "sending", "sent", "transferring", "completed", "timeout", "fail", "abort" }	"downloading"

1) completed, timeout, fail, abort 상태는 펌웨어 업데이트가 종료된 상태이다.

Request Example)

GET /ota?cctvId=P1\_B3\_1\_1

Response Example)

```
{
  "result": "success",
  "data": [
    {
      "nid": "P1_B3_1_1@A.1",
      "id": 1,
      "file": "CCTV_MESH_v1.1.9.bin",
      "mode": "single",
      "progress": "100",
      "result": "completed"
    }
  ]
}
```

## 2.2.4.1 펌웨어 업그레이드 (single)

Request)

Method	POST
URL	/ota-single/{cctvId}
Description	펌웨어 싱글 업그레이드를 실행 및 취소한다. (CCTV 단위)

Body)

항목	타입	설명	예
file	String	파일명	“CCTV MESH v1.1.10.bin”
Ids	Integer	조명 모듈 ID	[ 1, 2 ]
action	String	“run”: 실행 “cancel”: 취소	“run”

업그레이드 요청 이후 진행 상태는 /ota 를 API 를 통해서 조회할 수 있다.

Response) – JSON

기본 형식

Request Example) 업그레이드 실행

POST /ota-single/P1\_B3\_1\_1

```
{
  "file": "CCTV_MESH_v1.1.9.bin",
  "ids": [ 1, 2 ],
  "action": "run"
}
```

Response Example)

```
{
  "result": "success"
}
```

Request Example) 취소

POST /ota-single/P1\_B3\_1\_1

```
{  
  "action": "cancel"  
}
```

Response Example)

```
{  
  "result": "success"  
}
```

## 2.2.4.1 펌웨어 업그레이드 (multi)

Request)

Method	POST
URL	/ota-multi/{cctvId}
Description	펌웨어 멀티 업그레이드를 실행 및 취소한다. (CCTV 단위)

Body)

항목	타입	설명	예
file	String	파일명	“CCTV MESH v1.1.10.bin”
action	String	“run”: 실행 “cancel”: 취소	“run”

업그레이드 요청 이후 진행 상태는 /ota 를 API 를 통해서 조회할 수 있다.

Response) – JSON

기본 형식

Request Example)

POST /ota-multi/P1\_B2\_1\_1

```
{
  "file": "CCTV_MESH_v1.1.9.bin",
  "action": "run"
}
```

Response Example)

```
{
  "result": "success"
}
```

Request Example) 취소

POST /ota-multi/P1\_B2\_1\_1

```
{  
  "action": "cancel"  
}
```

Response Example)

```
{  
  "result": "success"  
}
```