# Streetlights API 1.0.0

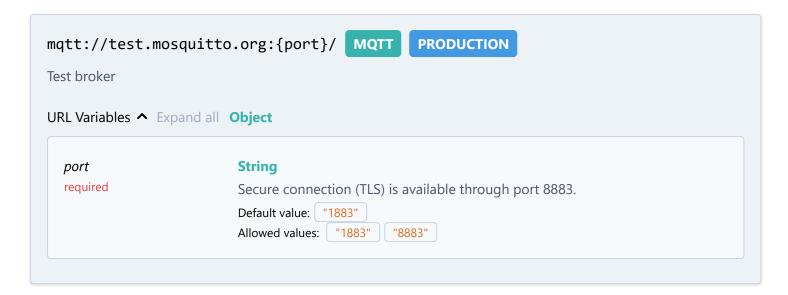
APACHE 2.0 APPLICATION/JSON

The Smartylighting Streetlights API allows you to remotely manage the city lights.

#### Check out its awesome features:

- Turn a specific streetlight on/off
- Dim a specific streetlight
- Receive real-time information about environmental lighting conditions

## Servers



## Operations

RECEIVE smartylighting/streetlights/1/0/event/{streetlightId}/lighting/measured

The topic on which measured values may be produced and consumed.

Inform about environmental lighting conditions of a particular streetlight.

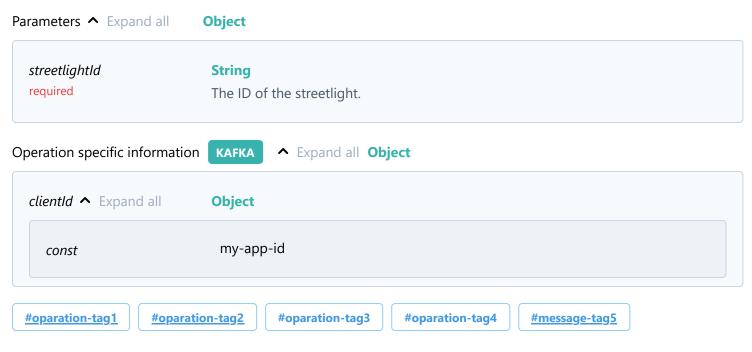
This is the description with **bold** text.

On multiple lines.

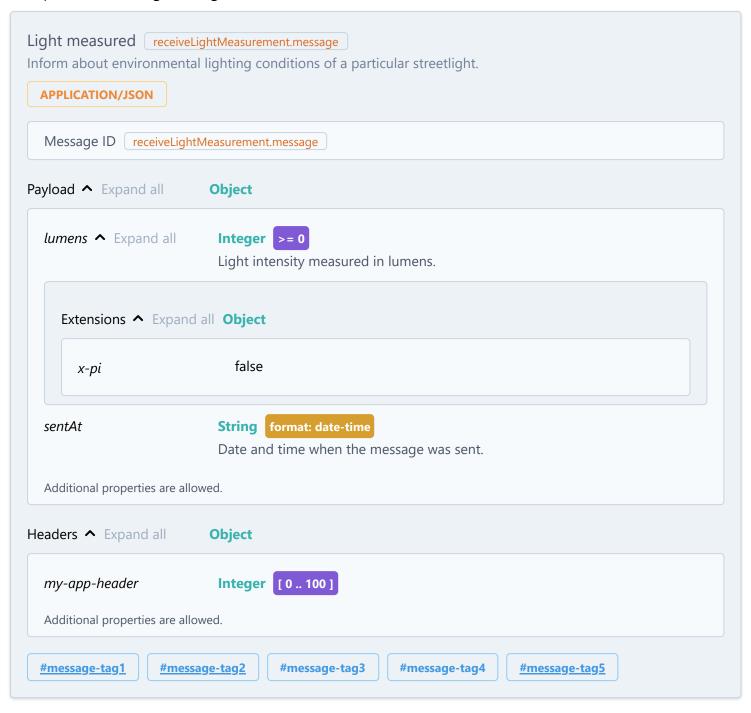
Operation ID receiveLightMeasurement

Available only on servers:

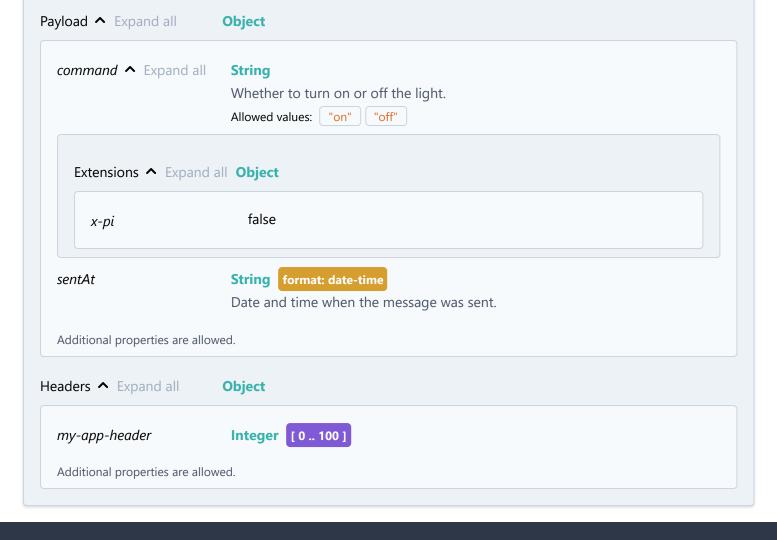
production



#### Accepts the following message:



```
Examples
  Payload
  {
    "lumens": ∅,
    "sentAt": "2019-08-24T14:15:22Z"
This example has been generated automatically.
  Headers
  {
    "my-app-header": 100
This example has been generated automatically.
SEND
        smartylighting/streetlights/1/0/action/{streetlightId}/turn/on
  Operation ID turnOn
Available only on servers:
 production
Parameters ^ Expand all
                            Object
  streetlightId
                              String
  required
                              The ID of the streetlight.
Operation specific information
                              KAFKA
                                        ▲ Expand all Object
  clientId ^ Expand all
                              Object
                               my-app-id
     const
Accepts the following message:
  Turn on/off turnOn.message
  Command a particular streetlight to turn the lights on or off.
    APPLICATION/JSON
     Message ID (turnOn.message)
```



```
Examples

Payload ^

{
    "command": "on",
    "sentAt": "2019-08-24T14:15:22Z"
}

This example has been generated automatically.

Headers ^

{
    "my-app-header": 100
}

This example has been generated automatically.
```

SEND smartylighting/streetlights/1/0/action/{streetlightId}/turn/off

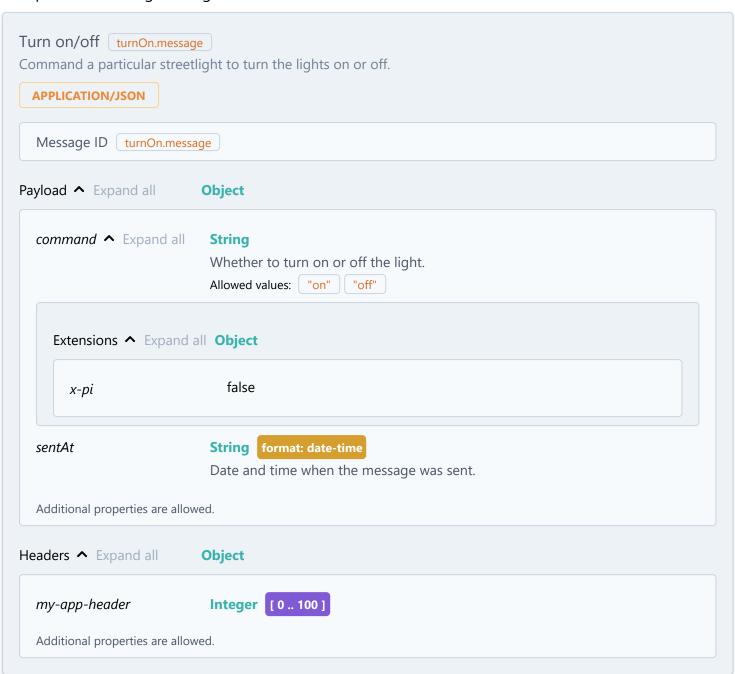
Operation ID turnOff

Available only on servers:



#### Accepts the following message:

production



```
Payload 

{
    "command": "on",
    "sentAt": "2019-08-24T14:15:22Z"
}

This example has been generated automatically.

Headers 

{
    "my-app-header": 100
}

This example has been generated automatically.
```



#### Accepts the following message:

```
Dim light dimLight.message

Command a particular streetlight to dim the lights.

APPLICATION/JSON

Message ID dimLight.message

Payload A Expand all Object
```

```
percentage

Integer [0..100]

Percentage to which the light should be dimmed to.

SentAt

String format: date-time

Date and time when the message was sent.

Additional properties are allowed.

Headers * Expand all Object

my-app-header Integer [0..100]

Additional properties are allowed.
```

```
Examples

Payload ^

{
    "percentage": 100,
    "sentAt": "2019-08-24T14:15:22Z"
    }

This example has been generated automatically.

Headers ^

{
    "my-app-header": 100
    }
```

```
SEND some.channel
```

this description shows in markdown

```
Operation ID some.channel.subscribe
```

Available only on servers:

```
<u>production</u>
```

Accepts **one of** the following messages:

```
#0 subscribe.message.0

APPLICATION/JSON
```

```
Payload  Expand all Object

result String
Examples values: "success"

Additional properties are allowed.
```

### Examples

```
Payload

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

This example has been generated automatically.

```
#1 subscribe.message.1

APPLICATION/JSON

Message ID subscribe.message.1

Payload  Expand all Object

Examples values: ("error":("errorCode":404, "errorMessage":"Something messed up")}

error  Expand all Object

errorCode Integer

errorMessage String

Additional properties are allowed.

Additional properties are allowed.
```

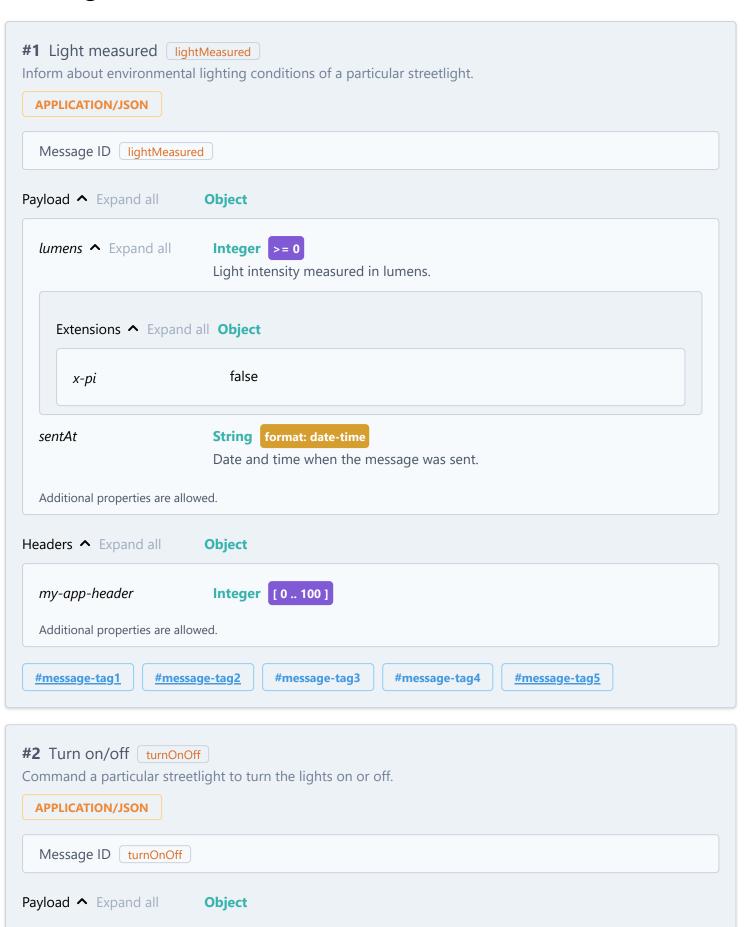
### **Examples**

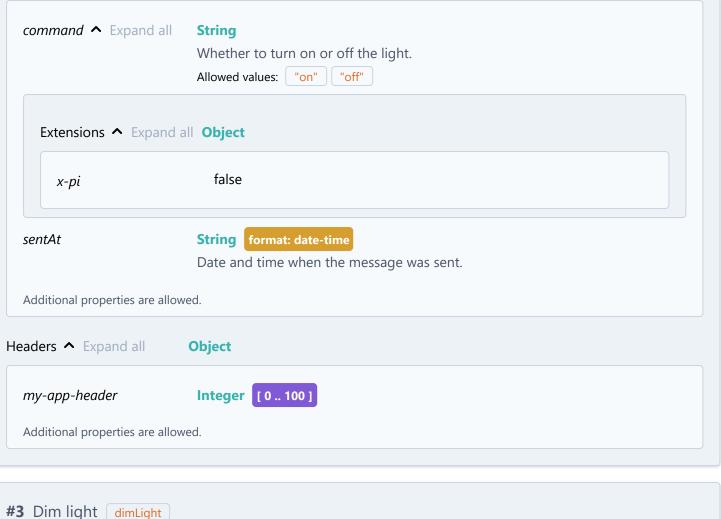
Payload ^

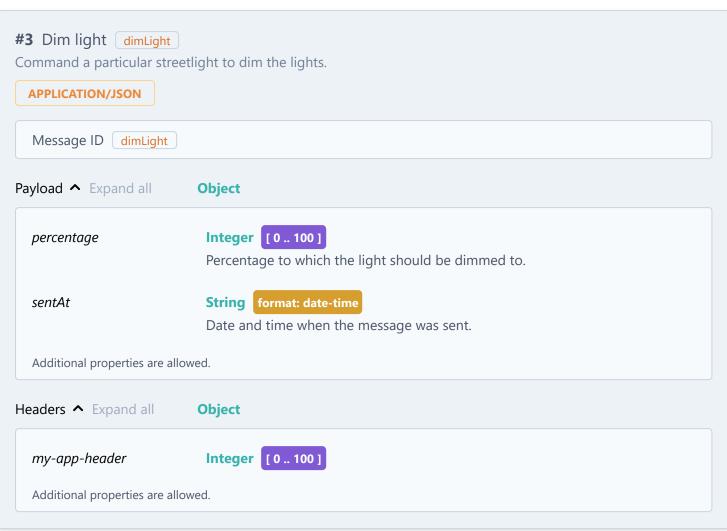
#### #1 Example

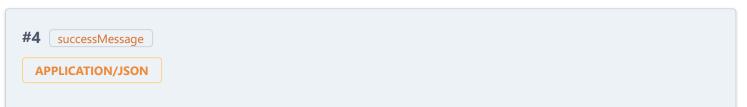
```
{
  "error": {
    "errorCode": 404,
    "errorMessage": "Something messed up"
```

# Messages

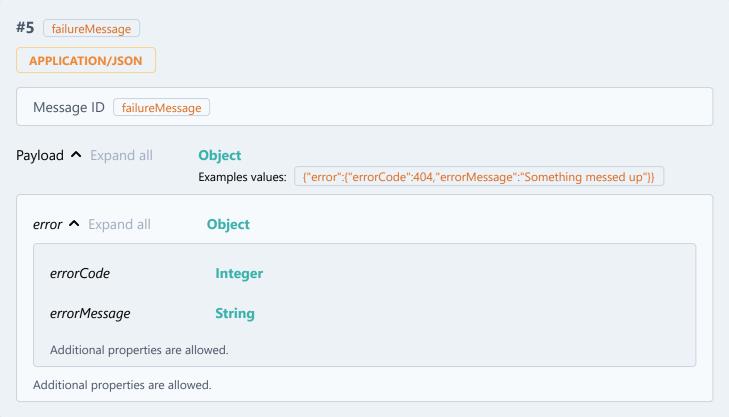




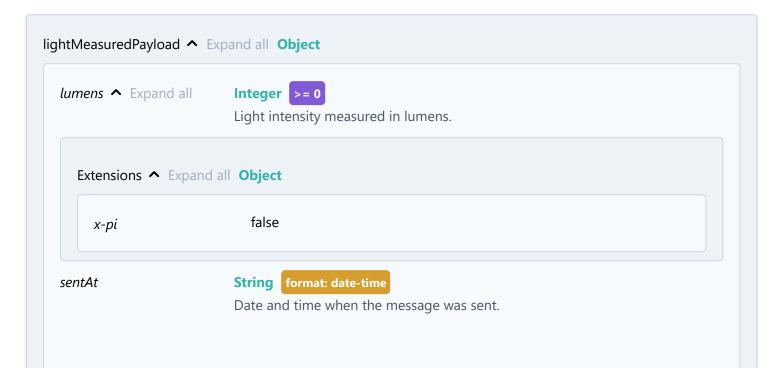




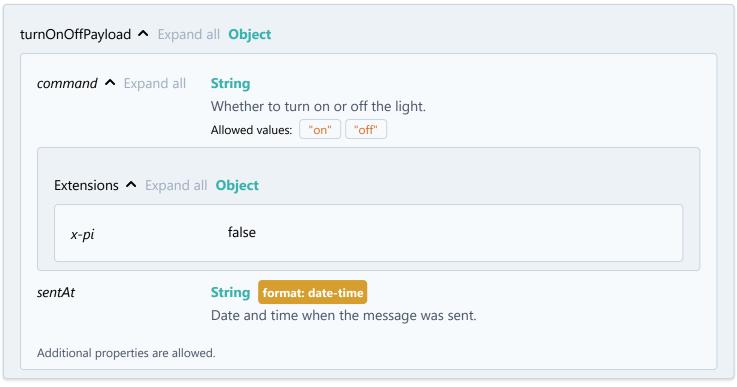


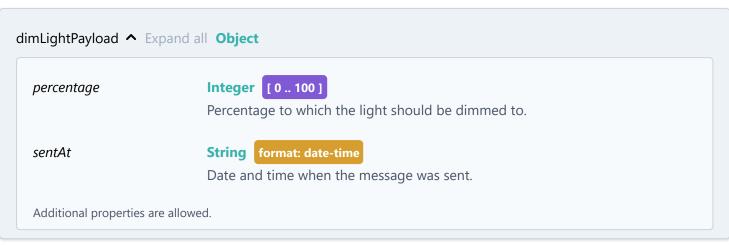


## Schemas



Additional properties are allowed.





sentAt String format: date-time

Date and time when the message was sent.