

API Reference

Tracking Endpoint

API Version: 1.0

INDEX

1. ENDPOINTS	3
1.1 DELETE /tracking/all	3
1.2 GET /tracking/cars	3
1.3 GET /tracking/cars/{vin}	3
1.4 GET /tracking/cars/{vin}/latest	4
1.5 GET /tracking/traffic-lights	4
1.6 GET /tracking/traffic-lights/{id}	5
1.7 GET /tracking/traffic-lights/{id}/latest	5

API

1. ENDPOINTS

1.1 DELETE /tracking/all

Delete all saved entities

Deletes all cars and traffic lights from the database

REQUEST

No request parameters

RESPONSE

STATUS CODE - 200: Default Response

RESPONSE MODEL - application/json

```
{
}
```

1.2 GET /tracking/cars

get all cars

Get information about all cars in database. Is empty when no car was found

REQUEST

No request parameters

RESPONSE

STATUS CODE - 200: Successful response

RESPONSE MODEL - application/json

```
[ {
  Array of object:
    vin      string  Vin of car
    location  number  horizontal location of car
    speed     number  speed of car
    timestamp string  timestamp of the change
}]
```

1.3 GET /tracking/cars/{vin}

get all information about specific car

Get all information about car with specific vin

REQUEST

PATH PARAMETERS

NAME	TYPE	DESCRIPTION
*vin	string	car vin

RESPONSE

STATUS CODE - 200: Successful response. Is empty when car with vin does not exist

RESPONSE MODEL - application/json

```
[{  
  Array of object:  
    vin      string  Vin of car  
    location  number  horizontal location of car  
    speed     number  speed of car  
    timestamp string  timestamp of the change  
}]
```

1.4 GET /tracking/cars/{vin}/latest

get latest specific car

Get latest information about a car with specified vin

REQUEST

PATH PARAMETERS

NAME	TYPE	DESCRIPTION
*vin	string	car vin

RESPONSE

STATUS CODE - 200: Successful response. Is empty when no car with vin was found

RESPONSE MODEL - application/json

```
{  
  vin      string  Vin of car  
  location  number  horizontal location of car  
  speed     number  speed of car  
  timestamp string  timestamp of the change  
}
```

1.5 GET /tracking/traffic-lights

get all traffic-lights

Get information about all traffic-lights in database. Is empty when no traffic-light was found

REQUEST

No request parameters

RESPONSE

STATUS CODE - 200: Successful response

RESPONSE MODEL - application/json

```
[{
  Array of object:
    id                string  ID of traffic light
    remainingMilliseconds number remaining time, the traffic light keeps its state
    color              string  PATTERN:^(green|red){1}$
    location            number  horizontal location of traffic light on the street
    timestamp           string  timestamp of the change
}]
```

1.6 GET /tracking/traffic-lights/{id}

get all information about specific traffic-light

Get all information about traffic-light with specific id

REQUEST

PATH PARAMETERS

NAME	TYPE	DESCRIPTION
*id	string	traffic-light id

RESPONSE

STATUS CODE - 200: Successful response. Is empty when traffic-lights with id does not exist

RESPONSE MODEL - application/json

```
[{
  Array of object:
    id                string  ID of traffic light
    remainingMilliseconds number remaining time, the traffic light keeps its state
    color              string  PATTERN:^(green|red){1}$
    location            number  horizontal location of traffic light on the street
    timestamp           string  timestamp of the change
}]
```

1.7 GET /tracking/traffic-lights/{id}/latest

get latest specific traffic-light

Get information about latest traffic-light with specified id

REQUEST

PATH PARAMETERS

NAME	TYPE	DESCRIPTION
*id	string	traffic-light id

RESPONSE

STATUS CODE - 200: Successful response. Is empty when no traffic-light with id was found

RESPONSE MODEL - application/json

{		
id	string	ID of traffic light
remainingMilliseconds	number	remaining time, the traffic light keeps its state
color	string	PATTERN:^(green red){1}\$
location	number	horizontal location of traffic light on the street
timestamp	string	timestamp of the change
}		
