

# Robot API

http://127.0.0.1:8000/api

## Table of contents

- [/robot](#)
  - [GET /robot](#)
  - [POST /robot](#)
  - [GET /robot/{robotId}](#)
  - [PUT /robot/{robotId}](#)
  - [DELETE /robot/{robotId}](#)
  - [GET /robot/{robotId}/joint](#)
  - [GET /robot/{robotId}/joint/{jointId}](#)
  - [PUT /robot/{robotId}/joint/{jointId}](#)
  - [GET /robot/{robotId}/link](#)
  - [POST /robot/{robotId}/link](#)
  - [GET /robot/{robotId}/link/{linkId}](#)
  - [PUT /robot/{robotId}/link/{linkId}](#)
  - [DELETE /robot/{robotId}/link/{linkId}](#)
  - [GET /robot/{robotId}/link/{linkId}/geometry](#)
  - [POST /robot/{robotId}/link/{linkId}/geometry](#)
  - [GET /robot/{robotId}/link/{linkId}/geometry/{geometryId}](#)
  - [PUT /robot/{robotId}/link/{linkId}/geometry/{geometryId}](#)
  - [DELETE /robot/{robotId}/link/{linkId}/geometry/{geometryId}](#)

## /robot

### GET /robot

Retrieves the list of all existing robots

GET /robot

## Response 200

Returns the list of robots

### [Body](#)

## POST /robot

Creates a new robot given its name

POST /robot

### [Body](#)

## Response 201

The robot was successfully created. Returns the description of the created robot

### [Body](#)

## Response 400

Invalid request. The JSON arguments was invalid. The robot wasn't created.

## GET /robot/{robotId}

Retrieves the description of a robot

GET /robot/{robotId}

### URI Parameters

- **robotId**: *required (string)*  
The robot unique Id

## Response 200

Returns the description of the robot

### [Body](#)

## Response 404

If robotId is invalid

## PUT /robot/{robotId}

Sets the properproperties of a robot. All empty fields are left as-if. At least one field must be filled.

```
PUT /robot/{robotId}
```

### URI Parameters

- **robotId**: *required (string)*  
The robot unique Id

### Body

## Response 202

Returns the new description of the robot

### Body

## Response 400

Invalid request. The JSON arguments was invalid. The robot wasn't modified.

## Response 404

If robotId is invalid

## DELETE /robot/{robotId}

Deletes a robot

```
DELETE /robot/{robotId}
```

### URI Parameters

- **robotId**: *required (string)*  
The robot unique Id

## Response 204

The robot was successfuly deleted

## Response 404

If robotId is invalid

## GET /robot/{robotId}/joint

Lists all joints of a robot

```
GET /robot/{robotId}/joint
```

### URI Parameters

- **robotId**: *required (string)*  
The robot unique Id

## Response 200

Returns the list of joints of the robot

### Body

## Response 404

If robotId is invalid

## GET /robot/{robotId}/joint/{jointId}

Retrieves the description of a joint

```
GET /robot/{robotId}/joint/{jointId}
```

### URI Parameters

- **robotId**: *required (string)*  
The robot unique Id
- **jointId**: *required (string)*  
The joint unique Id

## Response 200

Returns the description of the joint

### Body

## Response 404

If robotId or jointId is invalid

## PUT /robot/{robotId}/joint/{jointId}

Sets the properproperties of a joint. All empty fields are left as-if. At least one field must be filled.

PUT /robot/{robotId}/joint/{jointId}

### URI Parameters

- **robotId**: *required (string)*  
The robot unique Id
- **jointId**: *required (string)*  
The joint unique Id

### Body

## Response 202

Returns the new description of the joint

### Body

## Response 400

Invalid request. The JSON arguments was invalid. The joint wasn't modified.

## Response 404

If robotId or jointId is invalid

## GET /robot/{robotId}/link

Lists all links of a robot

GET /robot/{robotId}/link

### URI Parameters

- **robotId**: *required (string)*  
The robot unique Id

## Response 200

Returns the list of links of the robot

### Body

## Response 404

If robotId is invalid

## POST /robot/{robotId}/link

Creates a new link in the robot

POST /robot/{robotId}/link

### URI Parameters

- **robotId**: *required (string)*  
The robot unique Id

### Body

## Response 201

The link was successfully created. Returns the description of the created link

### Body

## Response 400

Invalid request. The JSON arguments was invalid. The link wasn't created.

## Response 404

If robotId is invalid

## GET /robot/{robotId}/link/{linkId}

Retrieves the description of a link

GET /robot/{robotId}/link/{linkId}

### URI Parameters

- **robotId**: *required (string)*  
The robot unique Id
- **linkId**: *required (string)*  
The link unique Id

## Response 200

Returns the description of the link

### Body

## Response 404

If robotId or linkId is invalid

## PUT /robot/{robotId}/link/{linkId}

Sets the properties of a link. All empty fields are left as-is. At least one field must be filled.

```
PUT /robot/{robotId}/link/{linkId}
```

### URI Parameters

- **robotId**: *required (string)*  
The robot unique Id
- **linkId**: *required (string)*  
The link unique Id

### Body

## Response 202

Returns the new description of the robot

### Body

## Response 400

Invalid request. The JSON arguments was invalid. The link wasn't modified.

## Response 404

If robotId or linkId is invalid

## DELETE /robot/{robotId}/link/{linkId}

Deletes a link

```
DELETE /robot/{robotId}/link/{linkId}
```

### URI Parameters

- **robotId**: *required (string)*  
The robot unique Id
- **linkId**: *required (string)*  
The link unique Id

## Response 204

The link was successfully deleted

## Response 404

If robotId or linkId is invalid

## GET /robot/{robotId}/link/{linkId}/geometry

Lists all geometries of a link

```
GET /robot/{robotId}/link/{linkId}/geometry
```

### URI Parameters

- **robotId:** *required (string)*  
The robot unique Id
- **linkId:** *required (string)*  
The link unique Id

## Response 200

Returns the list of geometries of the link

### Body

## Response 404

If robotId or linkId is invalid

## POST /robot/{robotId}/link/{linkId}/geometry

Creates a new geometry in the link

```
POST /robot/{robotId}/link/{linkId}/geometry
```

### URI Parameters

- **robotId:** *required (string)*  
The robot unique Id
- **linkId:** *required (string)*  
The link unique Id

### Body

## Response 201



Returns the new description of the created geometry

## Body

## Response 400

Invalid request. The JSON arguments was invalid. The geometry wasn't created.

## Response 404

If robotId or linkId is invalid

## GET /robot/{robotId}/link/{linkId}/geometry/{geometryId}

Retrieves the properties of a geometry

```
GET /robot/{robotId}/link/{linkId}/geometry/{geometryId}
```

### URI Parameters

- **robotId**: *required (string)*  
The robot unique Id
- **linkId**: *required (string)*  
The link unique Id
- **geometryId**: *required (string)*  
The link unique Id

## Response 200

Returns the description of the geometry

## Body

## Response 404

If robotId, linkId or geometryId is invalid

## PUT /robot/{robotId}/link/{linkId}/geometry/{geometryId}

Sets the properproperties of a geometry

```
PUT /robot/{robotId}/link/{linkId}/geometry/{geometryId}
```

### URI Parameters

- **robotId**: *required (string)*  
The robot unique Id
- **linkId**: *required (string)*  
The link unique Id
- **geometryId**: *required (string)*  
The link unique Id

## Body

## Response 202

Returns the new description of the geometry

## Body

## Response 400

Invalid request. The JSON arguments was invalid. The geometry wasn't modified.

## Response 404

If robotId, linkId or geometryId is invalid

## **DELETE /robot/{robotId}/link/{linkId}/geometry/{geometryId}**

Deletes a geometry

```
DELETE /robot/{robotId}/link/{linkId}/geometry/{geometryId}
```

### **URI Parameters**

- **robotId**: *required (string)*  
The robot unique Id
- **linkId**: *required (string)*  
The link unique Id
- **geometryId**: *required (string)*  
The link unique Id

## Response 204

The geometry was successfully deleted

## Response 404

If robotId, linkId or geometryId is invalid

