# Web Development 2
## Semester 5 - Bics

KOCH Elliot
0190432505 / elliot.koch.001@student.uni.lu

## Homework 5

**Question 1:** Add the necessary documentation blocks to your endpoints, including the expected URI parameters, request body, success response, response errors and examples of successful and successful responses.

The goal of this first question is to add the lines of code needed to create an apidoc document for the index.js.

To be able to create a api document, I have to write comment lines of code that look like this:

/**

*  Action to do

* Action to do

* Action to do

*/

I have to start a commented part with "/**" and end it with "*/". And between these delimiters, I can perform some actions that will allow the creation of the apidoc.

So here is a picture of the get-part of the apidoc:

```
/**
 * @api {get} /city/:Name CityInfo
 * @apiName GetCityInfo
 * @apiDescription Request information for a cityGet
 * @apiGroup GET
 *
 * @apiParam {String} Name unique city name.
 *
 * @apiSuccess {String} ID Unique ID of the city inside the table.
 * @apiSuccess {String} Name  Name of the city.
 * @apiSuccess {String} CountryCode  Country code of the city.
 * @apiSuccess {String} District  District of the city.
 * @apiSuccess {String} Population  Population number of the city.
 *
 * @apiParamExample {Input Example to call the GetcityInfo}
 *      Example: curl -X GET localhost:3000/city/Kabul
 *
 * @apiSuccessExample Success-Response:
 * [
 *      {
 *       "ID": 1,
 *       "Name":"Kabul",
 *       "CountryCode":"AFG",
 *       "District":"Kabol",
 *       "Population":1780000
 *      }
 *]
 */
```

Here is a picture of the post-part of the apidoc:

```
/**
 * @api {post} //city/:Name/:CountryCode/:District/:Population PostCityInfo
 * @apiName PostCityInfo
 * @apiDescription tadd a city in the table
 * @apiGroup POST
 *
 * @apiParam {String} Name city name.
 * @apiParam {String} CountryCode  Country code of the city.
 * @apiParam {String} District District of the city.
 * @apiParam {String} Population Population number of the city.
 *
 * @apiSuccess {String} ID Unique ID of the city inside the table.
 * @apiSuccess {String} Name  Name of the city.
 * @apiSuccess {String} CountryCode  Country code of the city.
 * @apiSuccess {String} District  District of the city.
 * @apiSuccess {String} Population  Population number of the city.
 *
 * @apiParamExample {Input Example to call the PostCityInfo}
 *      Example: curl -X POST localhost:3000/city/Belval/LUX/Esh-sur-Alzette/10000
 *
 * @apiSuccessExample Success-Response:
 * [
 *      {
 *       "ID": NULL,
 *       "Name":"Belval",
 *       "CountryCode":"LUX",
 *       "District":"Esh-sur-Alzette",
 *       "Population":10000
 *      }
 *]
 */
```

Here is a picture of the put-part of the apidoc:

```
/**
 * @api {put} /city/:Name/:NewName/:CountryCode/:District/:Population PutCityInfo
 * @apiName PutCityInfo
 * @apiDescription Update information for a city
 * @apiGroup PUT
 *
 * @apiParam {String} Name unique city name (this param is used to find the city in the table).
 * @apiParam {String} NewName  updated Name of the city.
 * @apiParam {String} CountryCode  updated Country code of the city.
 * @apiParam {String} District  updated District of the city.
 * @apiParam {String} Population  updated Population number of the city.
 *
 * @apiSuccess {String} ID Unique ID of the city inside the table.
 * @apiSuccess {String} Name  Name of the city.
 * @apiSuccess {String} CountryCode  Country code of the city.
 * @apiSuccess {String} District  District of the city.
 * @apiSuccess {String} Population  Population number of the city.
 *
 * @apiParamExample {Input Example to call the PutCityInfo}
 *      Example: curl -X PUT localhost:3000/city/Kabul/Kabul/FRA/Kabol/2780000
 *
 * @apiSuccessExample Success-Response:
 * [
 *      {
 *       "ID": 1,
 *       "Name":"Kabul",
 *       "CountryCode":"FRA",
 *       "District":"Kabol",
 *       "Population":2780000
 *      }
 *]
 */
```

Here is a picture of the delete-part of the apidoc:

```
/**
 * @api {delete} /city/:Name DeleteCity
 * @apiDescription delete a city in the tableD
 * @apiName DeleteCity
 * @apiGroup DELETE
 *
 * @apiParam {String} Name unique city name.
 *
 * @apiSuccess {Boolean} True when the city has been deleting in the table
 *
 * @apiParamExample {Input Example to call the DeleteCity}
 *      Example: curl -X DELETE localhost:3000/city/Kabul
 *
 * @apiSuccessExample Success-Response:
 * [
 *      {true}
 * ]
 *
 * @apiError {Boolean} False when the city cannot be deleting in the table
 *
 *
 * @apiErrorExample Error-Response:
 * [
 *      {false}
 * ]
 *
 */
```

In those pictures, we can see that I use different parameters that I found in the apidoc website to construct my apidoc:

“@api {method} path  title” I use this command to first indicate the title of the section that is going to be presented in the apidoc, and to show which method I use with the path that we have to use to call it.

“@apiDescription text” I use this command to describe what is going to be about each part of the apidoc.

“@apiName name” I use this command to specify the name of the part of the apidoc.

“@apiGroup name” I use this command to define to which group the method documentation block belongs.

“@apiParam {type} Fieldname description” I use this command to describe the parameters that I used to call a method. For example, in the get-method, the only parameter that I use is the Name: @apiParam Name name of a city.

“@apiSuccess {type} Fieldname description”  I use this command to describe the parameters that are returned when a method is called.

“@apiParamExample {type} description

example” I use this command to show how to call a method of the index.js file.

“@apiSuccessExample {type} description

example” I use this command to show the return of a method when it’s successful.

“@apiError {type} Fieldname description” I use this command to describe the error that may raise a method (in my case the delete one).

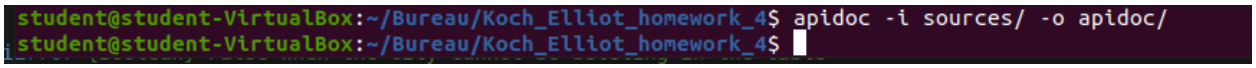
“@apiErrorExample {type} description

example” I use this command to show the return of a method when it’s unsuccessful.

**Question 2:** Generate a webpage using the apidoc command. All endpoints should appear properly documented in that webpage. *Hint*: See apiDoc example for the kind of output you should produce.

The goal of this question is to use an apidoc command to create the apidoc webpage that contains the documentation of index.js.

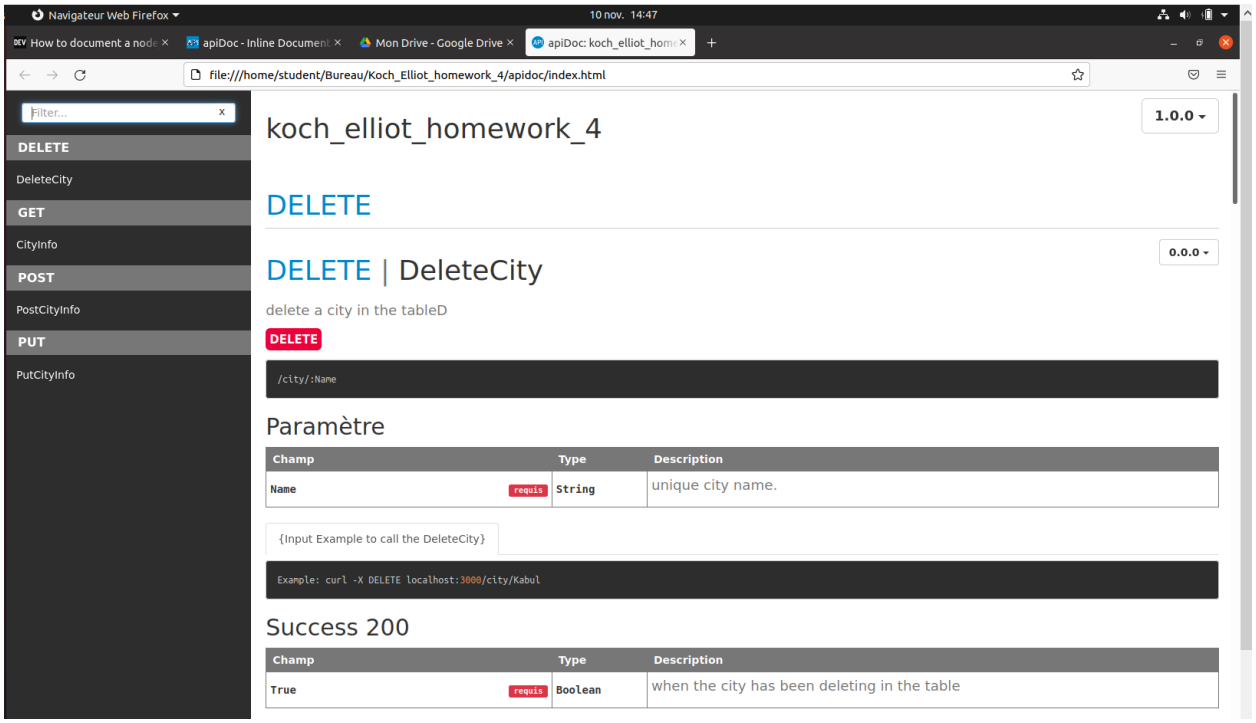
So to do so I use this simple line of code:

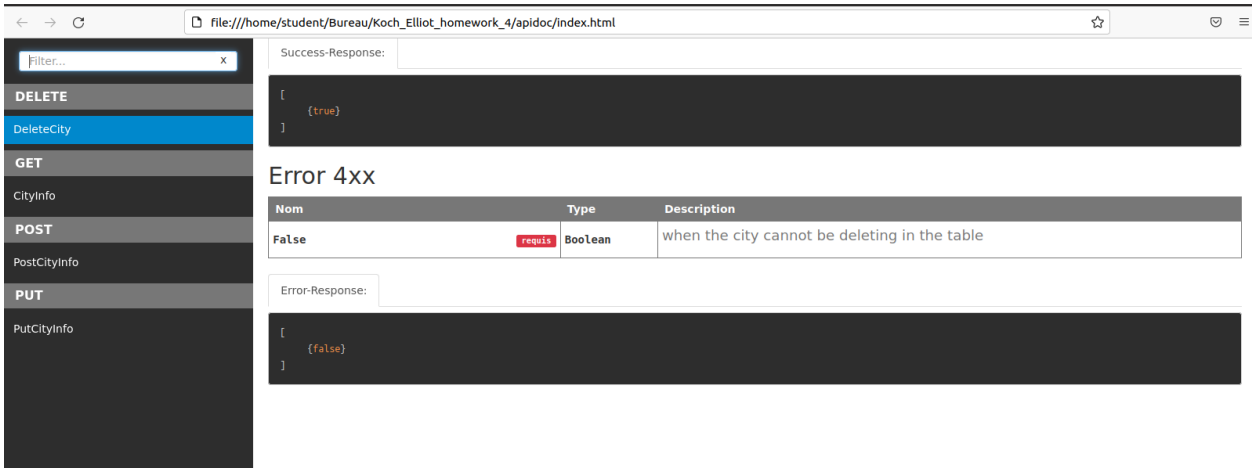


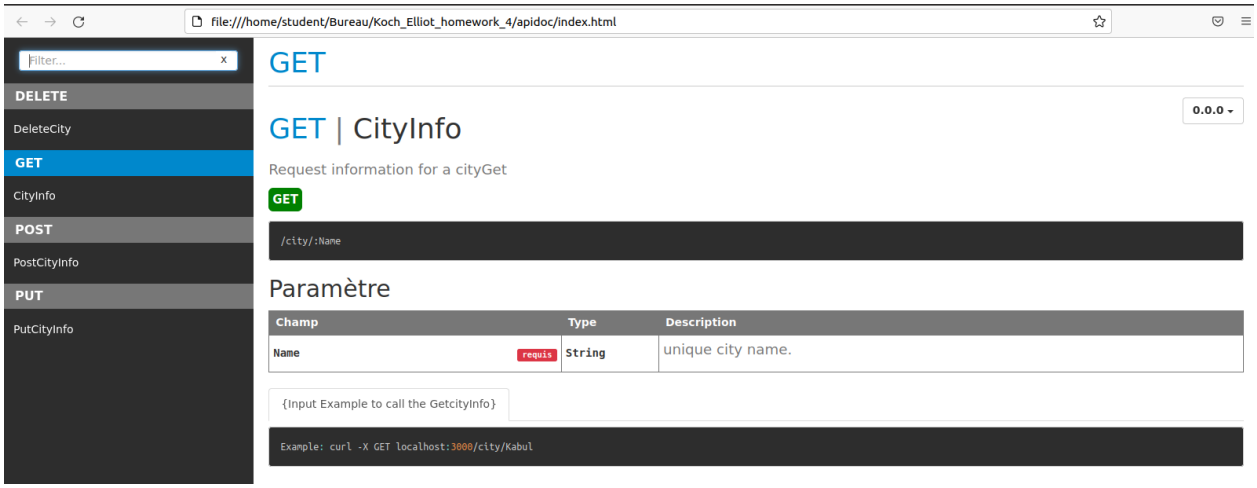

-i for Input / source directory name. (Location of my project files).

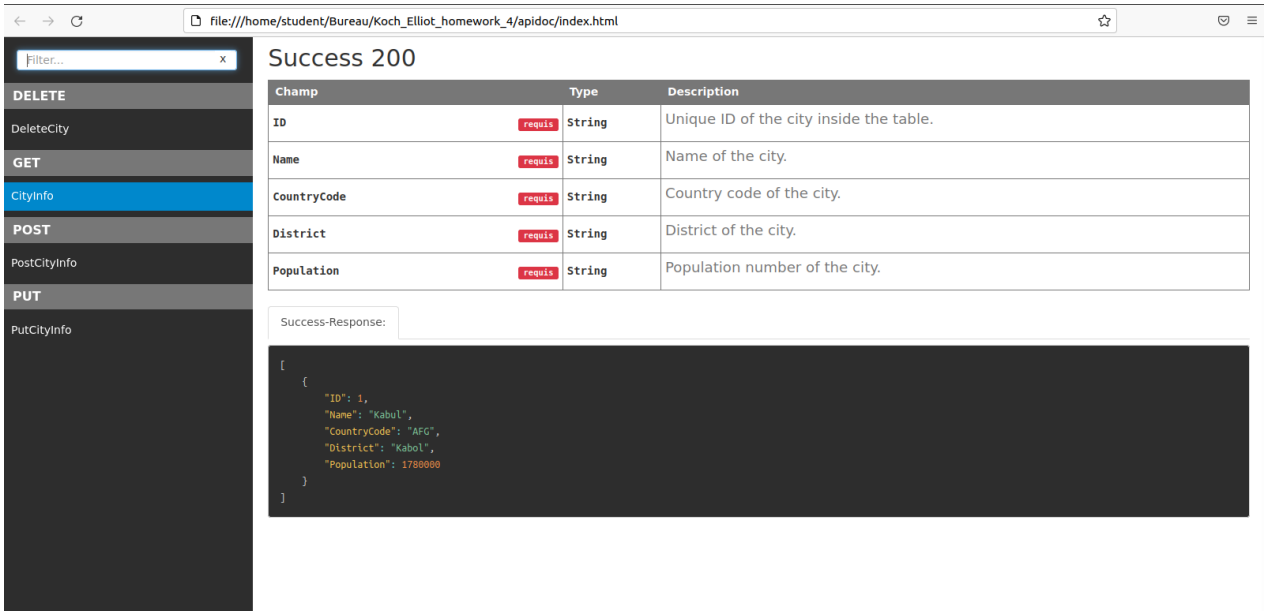
-o for Output directory name. (Location where to put the generated documentation).

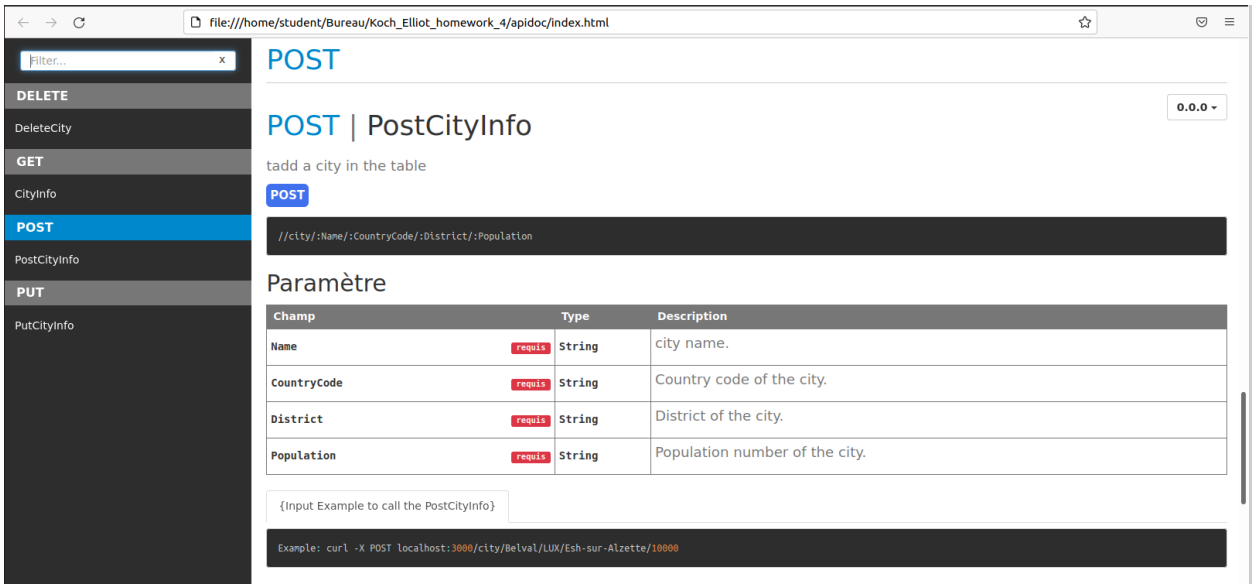
On the next pages you will see some screenshots of the apidoc that I have generated. You can also find it in the apidoc folder.

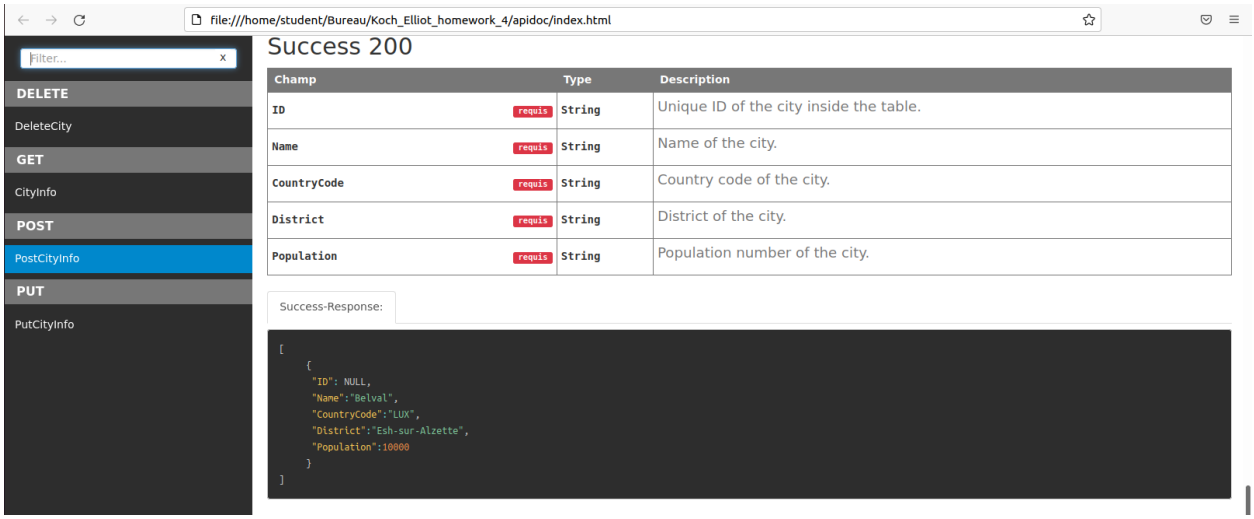

Navigateur Web Firefox
10 nov. 14:47
file:///home/student/Bureau/Koch_Elliot_homework_4/apidoc/index.html
Filter...
DELETE
DeleteCity
GET
CityInfo
POST
PostCityInfo
PUT
PutCityInfo
koch_elliot_homework_4
1.0.0
DELETE
DELETE | DeleteCity
0.0.0
delete a city in the tableD
DELETE
/city/:Name
Paramètre
Champ
Type
Description
Name
requis
String
unique city name.
{Input Example to call the DeleteCity}
Example: curl -X DELETE localhost:3000/city/Kabul
Success 200
Champ
Type
Description
True
requis
Boolean
when the city has been deleting in the table

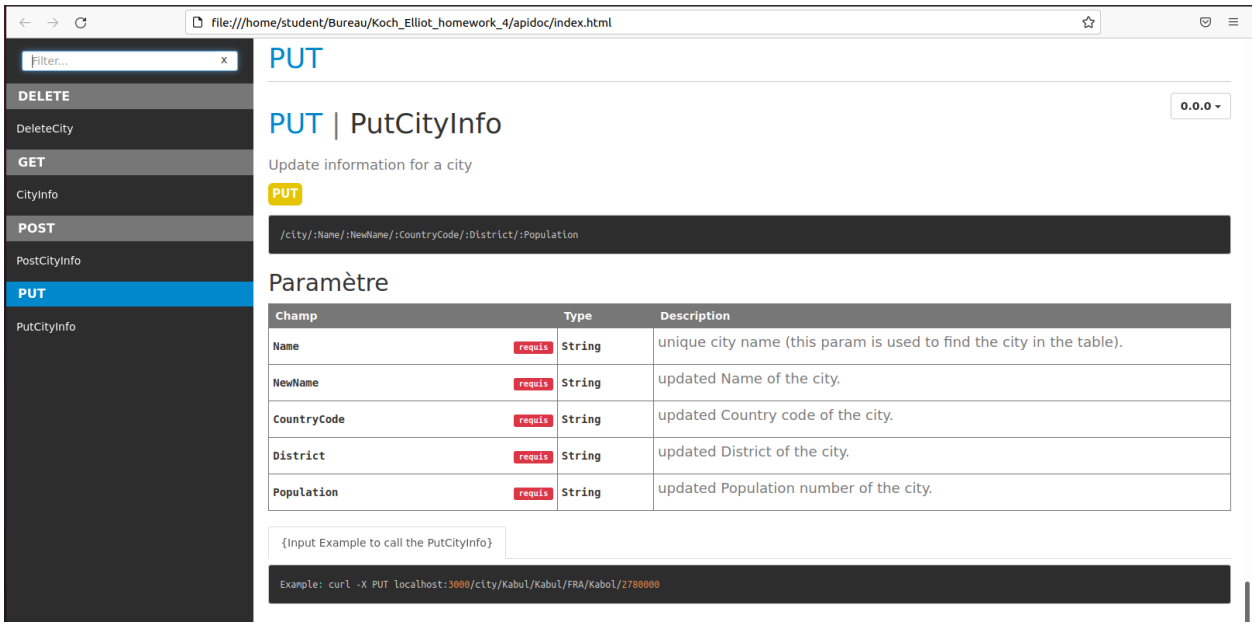

file:///home/student/Bureau/Koch_Elliot_homework_4/apidoc/index.html
Filter...
DELETE
DeleteCity
GET
CityInfo
POST
PostCityInfo
PUT
PutCityInfo
Success-Response:
[
{true}
]
Error 4xx
Nom
Type
Description
False
requis
Boolean
when the city cannot be deleting in the table
Error-Response:
[
{false}
]


file:///home/student/Bureau/Koch_Elliot_homework_4/apidoc/index.html
Filter...
DELETE
DeleteCity
GET
CityInfo
POST
PostCityInfo
PUT
PutCityInfo
GET
GET | CityInfo
0.0.0
Request information for a cityGet
GET
/city/:Name
Paramètre
Champ
Type
Description
Name
requis
String
unique city name.
{Input Example to call the GetcityInfo}
Example: curl -X GET localhost:3000/city/Kabul


file:///home/student/Bureau/Koch_Elliot_homework_4/apidoc/index.html
Filter...
DELETE
DeleteCity
GET
CityInfo
POST
PostCityInfo
PUT
PutCityInfo
Success 200
Champ
Type
Description
ID
requis
String
Unique ID of the city inside the table.
Name
requis
String
Name of the city.
CountryCode
requis
String
Country code of the city.
District
requis
String
District of the city.
Population
requis
String
Population number of the city.
Success-Response:
[
{
"ID": 1,
"Name": "Kabul",
"CountryCode": "AFG",
"District": "Kabol",
"Population": 1780000
}
]


file:///home/student/Bureau/Koch_Elliot_homework_4/apidoc/index.html
Filter...
DELETE
DeleteCity
GET
CityInfo
POST
PostCityInfo
PUT
PutCityInfo
POST
POST | PostCityInfo
0.0.0
tadd a city in the table
POST
//city/:Name/:CountryCode/:District/:Population
Paramètre
Champ
Type
Description
Name
requis
String
city name.
CountryCode
requis
String
Country code of the city.
District
requis
String
District of the city.
Population
requis
String
Population number of the city.
{Input Example to call the PostCityInfo}
Example: curl -X POST localhost:3000/city/Belval/LUX/Esh-sur-Alzette/10000


file:///home/student/Bureau/Koch_Elliot_homework_4/apidoc/index.html
Filter...
DELETE
DeleteCity
GET
CityInfo
POST
PostCityInfo
PUT
PutCityInfo
Success 200
Champ
Type
Description
ID
requis
String
Unique ID of the city inside the table.
Name
requis
String
Name of the city.
CountryCode
requis
String
Country code of the city.
District
requis
String
District of the city.
Population
requis
String
Population number of the city.
Success-Response:
[
{
"ID": NULL,
"Name":"Belval",
"CountryCode":"LUX",
"District":"Esh-sur-Alzette",
"Population":10000
}
]




## Success 200

| Champ | Type | Description |
|---|---|---|
| **ID** requis | **String** | Unique ID of the city inside the table. |
| **Name** requis | **String** | Name of the city. |
| **CountryCode** requis | **String** | Country code of the city. |
| **District** requis | **String** | District of the city. |
| **Population** requis | **String** | Population number of the city. |

Success-Response:

```
[
    {
        "ID": 1,
        "Name": "Kabul",
        "CountryCode": "FRA",
        "District": "Kabol",
        "Population": 2780000
    }
]
```