

# Flask programmer documentation

This is web application that has an assistant database.

This application is written in python.

This application has crud endpoints:




- GET /assistants – displaying source,
- GET /assistants/create – displaying create form,
- POST /assistants – storing source,
- GET /assistants/update – displaying update form,
- PUT /assistants/{id} – updates source,
- DELETE /assistants/{id} – deletes source

I will test them in postman and describe them.

Get/assistants- this endpoints just display all assistants from database.

```
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <meta charset="UTF-8">
6   <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"
7     integrity="sha384-Vkoo8x4CGs034Hwv6GT/09PaXtkKtu6ug5TOeNV6gBiFeWPGFN9nUnOf23Q9Iffj" crossorigin="anonymous">
8
9 </head>
10
11 <body>
12   <script>
13     function deleteAssistant(id) {
14       fetch(document.baseURI + '/' + id, {
15         method: 'DELETE'
16       })
17       .then(res => res.json())
18       .then(data => location.replace(document.baseURI));
19     }
20   </script>
21
22   <table class="table">
23     <thead class="thead-dark">
24       <tr>
25         <th>Assistant Id</th>
26         <th>Job Id</th>
27         <th>First name</th>
28         <th>Last name</th>
29         <th>Email address</th>
30         <th>Picture</th>
31       </tr>
32     </thead>
33   </table>
34
```

Here is ss from web browser:

Assistant Id	Job Id	First name	Last name	Email address	Picture	Action
1	4th Grade Teacher	Iza	Piechowiak	iza@gmail.com		<div>Delete</div> <div>Update</div>
2	4th Grade Teacher	Janusz	Kowalski	janusz@gmail.com		<div>Delete</div> <div>Update</div>
3	4th Grade Teacher	Jacek	Kaczmarek	kaczmarek.jacek10@gmail.com		<div>Delete</div> <div>Update</div>

Get/assistants/create- this endpoint just shows page where you can enter new assistant data and then save them.

SS form postman:

The screenshot shows a Postman interface with a GET request to `http://127.0.0.1:5000/assistants/create`. The response is an HTML document. The body of the response is as follows:

```
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <meta charset="UTF-8">
6   <link rel="stylesheet" href="/static/css/main.css">
7   <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"
8     integrity="sha384-Vkoo8x4CGs034HmVx087/QSpatxkcu6ugSTOelV6g8BIfewP9fH9M9O7Z39q1fjhn" crossorigin="anonymous">
9
10  <head>
11    <h3>Assistant Registration Form</h3>
12  </head>
13
14
15 </head>
16
17 <body>
18
19   <form action="/assistants" method="POST" enctype="multipart/form-data">
20     <br>
21     <div class="form-group">
22       <h5>
23         <label for="exampleFormControlInput1">First name:</label>
24       </h5>
25       <input type="text" name="first_name" id="first_name"
26         class="form-control" id="exampleFormControlInput1" placeholder="insert first name" required>
27     <br>
28     <h5>
29       <label>Last name:</label>
30     </h5>
31     <input type="text" name="last_name" id="last_name" class="form-control" id="exampleFormControlInput1" placeholder="insert last name" required>
32     <br>
33   </form>
```

SS form web browser:

## Assistant Registration Forumlar

First name:

Last name:

Email:

Job name:

Upload picture:

Not choosing any file will casue file to generate automaticly

Here all data fields are required, so if u don't insert all needed data then you won't be able to save them and add to database.

First name and Last name are text fields.

Email is email field so u need type there [email@gmail.com](mailto:email@gmail.com) or another correct email.

Job name you can choose job. Jobs are from json file.

Last option, you can upload your own photo or photo will be automatically generated from you from website: <https://randomuser.me/api>

Post/assistants- this happens when you press save button and all data are correct.

Then new assistant is added to database.

SS from postman:

POST http://127.0.0.1:5000/assistants

Send Save

Params Authorization Headers (8) **Body** Pre-request Script Tests Settings Cookies Code

none form-data x-www-form-urlencoded raw binary GraphQL







KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> inputfile	tommy.jpg	
<input checked="" type="checkbox"/> first_name	Bartek	
<input checked="" type="checkbox"/> last_name	Kaczmarek	
<input checked="" type="checkbox"/> email_address	bartek@gmail.com	
<input checked="" type="checkbox"/> job_id	15	
<input checked="" type="checkbox"/> pic_name	tommy	
Key	Value	Description

Body Cookies Headers (4) Test Results Status: 200 OK Time: 39 ms Size: 57.76 KB Save Response

Pretty Raw Preview Visualize HTML

```
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <meta charset="UTF-8">
6   <link rel="stylesheet" href="/static/css/main.css">
7   <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"
8     integrity="sha384-Vkoo8x4CGs03+Hhvx8T/Q5PaXtkKtu6ug5TOeNV6gB1FeWPGFN9MuhOf23Q9Ifjh"
9     crossorigin="anonymous">
10
11
12 </head>
13
14 <body>
15
16 <script>
```

Result in browser:

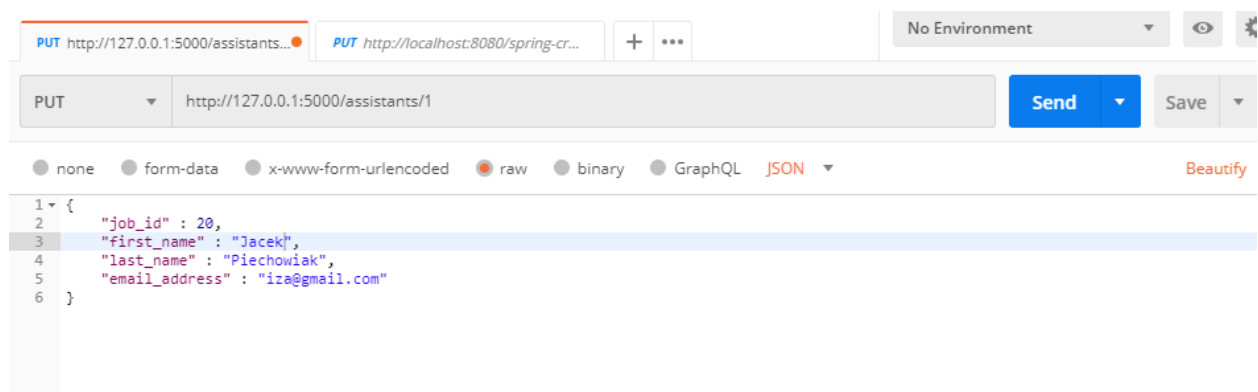
Assistant Id	Job Id	First name	Last name	Email address	Picture	Action
1	4th Grade Teacher	Iza	Piechowiak	iza@gmail.com		<div>Delete</div> <div>Update</div>
2	4th Grade Teacher	jacek	Kowalskaa	janusz@gmail.com		<div>Delete</div> <div>Update</div>
3	3D Animator	Jacek	Kaczmarek	kaczmarek.jacek10@gmail.com		<div>Delete</div> <div>Update</div>
4	2 Year Olds Preschool Teacher	Jacek	Kaczmarek	kaczmarek.jacek10@gmail.com		<div>Delete</div> <div>Update</div>
5	21 Dealer	Jacek	Kaczmarek	kaczmarek.jacek10@gmail.com		<div>Delete</div> <div>Update</div>
6	3rd Grade Teacher	Bartek	Kaczmarek	bartek@gmail.com		<div>Delete</div> <div>Update</div>

## Get/assistants/update


This endpoint just shows update form of selected assistant. It is same as create assistant but current data are already inserted and you can update them if you want.

PUT /assistants/{id} – this do all updating work and changes database data.

SS from postman:



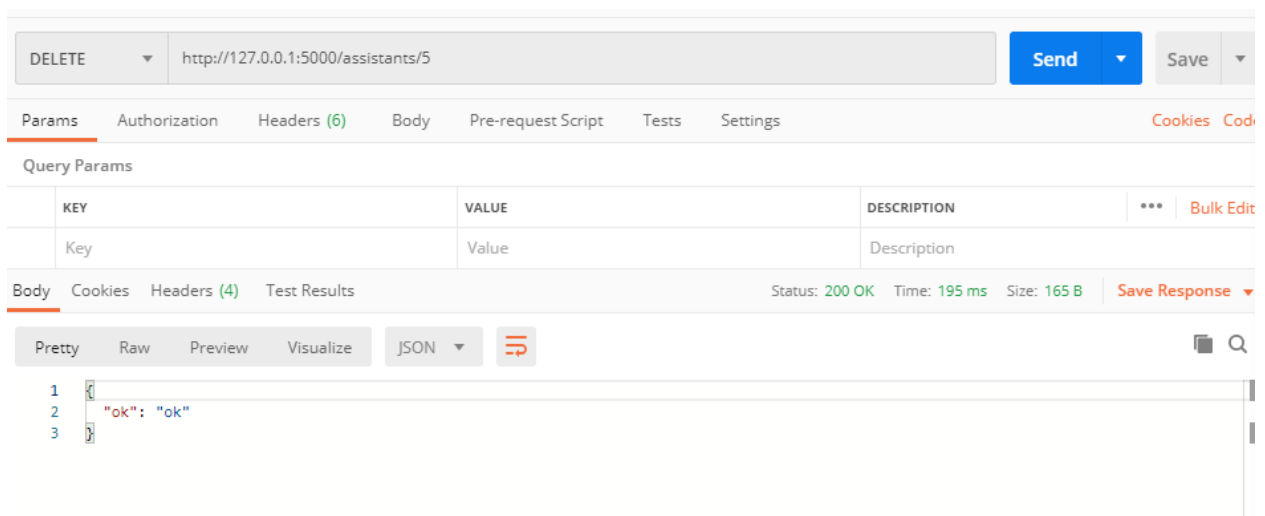
Result in web browser:

Assistant Id	Job Id	First name	Last name	Email address	Picture	Action
1	4th Grade Teacher	Jacek	Piechowiak	iza@gmail.com		<div>Delete</div> <div>Update</div>

DELETE /assistants/{id}- this endpoint removes one assistant data from database.

U can remove assistant by clicking red button 'Delete'.





SS from postman:



The image shows the Postman interface for a DELETE request. The request is sent to the URL `http://127.0.0.1:5000/assistants/5` with the method `DELETE`. The response is a `200 OK` status with a time of `195 ms` and a size of `165 B`. The response body is displayed in JSON format:

```
1 {
2   "ok": "ok"
3 }
```

SS form web browser:

Assistant Id	Job Id	First name	Last name	Email address	Picture	Action
1	4th Grade Teacher	Iza	Kaczmarek	iza@gmail.com		<div>Delete</div> <div>Update</div>
2	4th Grade Teacher	Janusz	Kowalski	janusz@gmail.com		<div>Delete</div> <div>Update</div>
3	4th Grade Teacher	Jacek	Kaczmarek	kaczmarek.jacek10@gmail.com		<div>Delete</div> <div>Update</div>
4	3D Artist	Jacek	Kaczmarek	kaczmarek.jacek10@gmail.com		<div>Delete</div> <div>Update</div>

As you can see there is no assistant with id number equal to 5. But remember to insert correct id number. If the assistant you want to delete does not exist, then the program won't remove anything from the database.