

## Python Flask how to get parameters from a URL?

[Ask Question](#)

In Flask, How do I extract parameters from a URL? How can I extract named parameters from a URL using flask and python?

When the user accesses this URL running on my flask app, I want the web service to be able to handle the parameters specified after the question mark:

```
http://10.1.1.1:5000/login?username=alex&password=pw1
```

```
#I just want to be able to manipulate the parameters
@app.route('/login', methods=['GET', 'POST'])
def login():
    username = request.form['username']
    print(username)
    password = request.form['password']
    print(password)
```

[python](#)[web-services](#)[flask](#)[url-parameters](#)

edited Jan 23 '15 at 21:40



[Eric Leschinski](#)

84k 36 316 268

asked Jul 22 '14 at 15:49



[Alex Stone](#)

19.3k 44 179 328

28 Just a small hint for security: Don't include passwords in GET requests. [security.stackexchange.com/question/s/147188/...](https://security.stackexchange.com/question/s/147188/...) – [palsch](#) Mar 13 '17 at 15:18

### 4 Answers

Use `request.args` to get parsed contents of query string:

```
username = request.args.get('user')
password = request.args.get('passw
```

edited Jul 28 '14 at 15:50



Alex Stone

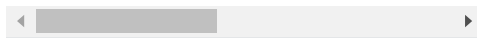
19.3k 44 179 328

answered Jul 22 '14 at 15:53



falsetru

241k 31 417 417



The URL parameters are available in [request.args](#), which is a [MultiDict](#) that has a `get` method, with optional parameters for default value (`default`) and type (`type`) - which is a callable that converts the input value to the desired format.

```
from flask import request

@app.route('/my-route')
def my_route():
    page = request.args.get('page', default)
    filter = request.args.get('filter',
```

Examples with the code above:

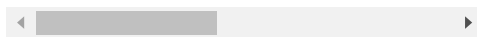
```
/my-route?page=34          -> page=34
/my-route                  -> page=default
/my-route?page=10&filter=test -> page=10, filter=test
/my-route?page=10&filter=10 -> page=10, filter=10
/my-route?page=*&filter=*   -> page=*, filter=*
```

answered Sep 20 '17 at 11:41



qqbenq

5,664 2 28 39



You can also use brackets `<>` on the URL of the view definition and this input will go into your view function arguments

```
@app.route('/<name>')
def my_view_func(name):
    return name
```

answered Mar 14 at 0:29



Inbar Cheffer

307 1 4

Flask doc reference:

[flask.pocoo.org/docs/1.0/quickstart/#variable-rules](https://flask.pocoo.org/docs/1.0/quickstart/#variable-rules) – markm Jul 10 at 0:54

If you have a single argument passed in the URL you can do it as follows

```
from flask import request
#url
http://10.1.1.1:5000/login/alex

from flask import request
@app.route('/login/<username>', method='GET')
def login(username):
    print(username)
```

In case you have multiple parameters:

```
#url
http://10.1.1.1:5000/login?username=alex&password=123456

from flask import request
@app.route('/login', methods=['GET'])
def login():
    username = request.args.get('username')
    print(username)
    password = request.args.get('password')
    print(password)
```

What you were trying to do works in case of POST requests where parameters are passed as form parameters and do not appear in the URL. In case you are actually developing a login API, it is advisable you use POST request rather than GET and expose the data to the user.

In case of post request, it would work as follows:

```
#url
http://10.1.1.1:5000/login
```

HTML snippet:

```
<form action="http://10.1.1.1:5000/login">
  Username : <input type="text" name="username">
  Password : <input type="password" name="password">
  <input type="submit" value="submit">
</form>
```

Route:

```
from flask import request
@app.route('/login', methods=['POST'])
def login():
    username = request.form.get('username')
    print(username)
    password = request.form.get('password')
    print(password)
```

answered Jul 17 at 15:18



[Suchita Mukherjee](#)

91 1 4