



# Daan Lenaerts

JUN3  
2015

by Daan Lenaerts

Python, Software  
development

http server,  
Tutorial, Web  
Server



## Create a simple http server with Python 3



Knowing how to create a simple http server comes in very handy, especially when working on projects where your application has to be accessed by a remote device. It's not very difficult to create an http server with Python, so let's dive straight into it.

Python is a very good language to do this, because it runs on Windows, Mac OS X and Linux, which makes your little web server very universal. Especially in a development environment.

To get started, create a new Python script with the following contents:

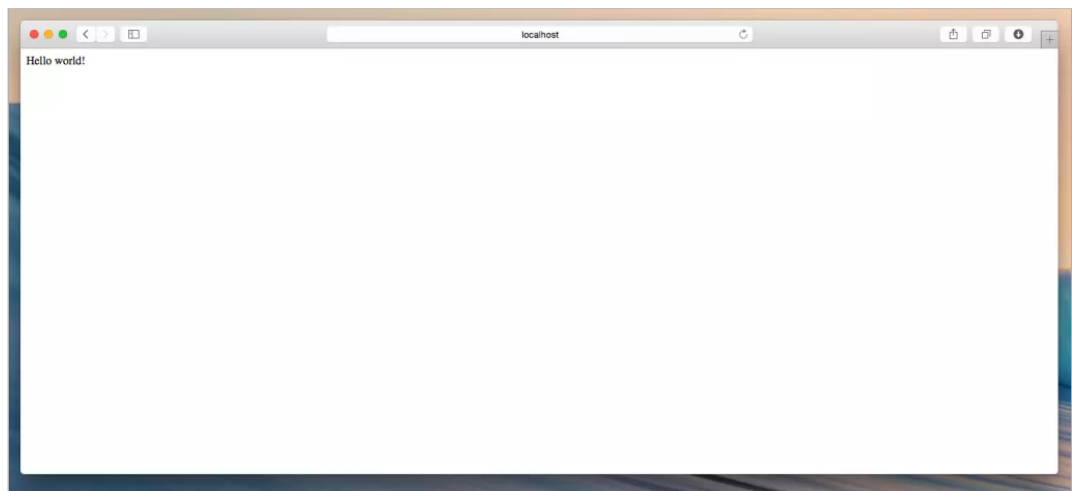
```
1  #!/usr/bin/env python
2
3  from http.server import BaseHTTPRequestHandler, HTTPServer
4
5  # HTTPRequestHandler class
6  class testHTTPServer_RequestHandler(BaseHTTPRequestHandler):
7
8      # GET
9      def do_GET(self):
10         # Send response status code
11         self.send_response(200)
12
13         # Send headers
14         self.send_header('Content-type', 'text/html')
15         self.end_headers()
16
17         # Send message back to client
18         message = "Hello world!"
19         # Write content as utf-8 data
```

```
20         self.wfile.write(bytes(message, "utf8"))
21     return
22
23     def run():
24         print('starting server...')
25
26         # Server settings
27         # Choose port 8080, for port 80, which is normally used for a http server, you need r
28         server_address = ('127.0.0.1', 8081)
29         httpd = HTTPServer(server_address, testHTTPServer_RequestHandler)
30         print('running server...')
31         httpd.serve_forever()
32
33
34     run()
```

When you run the script, the server will start on port 8081. If you want to use port 80, for example, you'll have to run your application with root permissions, so that's why I'm going with port 8081.

Whenever the server gets a GET request, it'll run the do\_GET function, and return a 200 (success) status code header, as well as the text/html content type header to the client system. Along with an hello world message.

To test the http server, open your web browser and type "http://localhost:8081", you'll see this:



---

Share this:



---

#### Related

Getting started with Python:  
functions

September 28, 2015

In "Python"

Python and SSH: sending  
commands over SSH using  
Paramiko

January 2, 2016

In "Python"

Python and SSH: Paramiko  
shell

July 1, 2016

In "Python"

13 Comments

---

**Eido**

July 25, 2016 at 7:49 am

Thank you very much for this straightforward example!

[□ Reply](#)

---

**Daan Lenaerts** Post author

July 25, 2016 at 7:59 am

You're welcome 😊

[□ Reply](#)

---

**Derek**

July 28, 2016 at 3:18 am

I'm curious how the instantiating of the `testHTTPServer_RequestHandler` class know's to run `do_GET`?

[□ Reply](#)

---

**Daan Lenaerts** Post author

July 31, 2016 at 4:31 pm

Hi Derek

The `do_GET` method is a method that gets called automatically, through the `BaseHTTPRequestHandler` we're subclassing from, when a GET request arrives.

The same can be done with other HTTP methods like `do_POST` for example.

I hope that clarifies it a bit.

[□ Reply](#)

---

**Gary Kuipers**

August 26, 2016 at 1:52 pm

Holy Maloney! It actually works as seen on the web page! After about 15 "examples" that require lots more work to make work, this one actually did what it was supposed to! I am flabbergasted! Thanks for something that actually works! Clear and straightforward as well.

[□ Reply](#)

---

**Alexandr**

December 20, 2016 at 8:39 am

Hello!

How can I print the request string?

Thank you

[□ Reply](#)



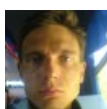
**Daan Lenaerts** Post author

December 25, 2016 at 3:34 pm

Hi Alexandr, thanks for your comment. You can do so by placing the following code in the do\_GET definition:

```
print(self.path)
```

[□ Reply](#)



**dagolinuxoid**

May 5, 2017 at 7:56 am

Make sure to use

```
#!/usr/bin/env python3
```

(if you don't use aliases) rather than just

```
#!/usr/bin/env python
```

[□ Reply](#)



**刀尖红叶**

July 7, 2017 at 2:19 am

When this server receive's a client's http get method, which method it use to response– http post, http put ?

[□ Reply](#)



**Daan Lenaerts** Post author

December 10, 2017 at 3:24 pm

Thanks for your comment. Those are all request methods. You can read more about it here: [https://www.w3schools.com/tags/ref\\_httpmethods.asp](https://www.w3schools.com/tags/ref_httpmethods.asp) .

[□ Reply](#)



**LUKE SHEPPARD**

October 30, 2017 at 6:07 am

Just wanted to say thanks for posting this python3 code. I'm going to be using your simple http server for a class tomorrow. It's an after-school programming class for a non-profit neighborhood center which serves underprivileged high school kids.

[□ Reply](#)

**Daan Lenaerts** Post author

December 10, 2017 at 3:22 pm



Hi Luke, thanks for your comment! I really appreciate it!  
It's amazing you're teaching programming to underprivileged kids. I'm glad I could help.  
Let me know if I can do more.

Kind regards

Daan

[□ Reply](#)



**Erik Aronesty**

November 17, 2017 at 6:03 pm

I use this:

<https://github.com/earonesty/dotfiles/blob/master/apiserver.py>

[□ Reply](#)

Leave a Reply

Your email address will not be published. Required fields are marked \*

Comment

Name \*

Email \*

Website

Protected with IP Blacklist Cloud

I'm not a robot

reCAPTCHA  
[Privacy](#) - [Terms](#)

POST COMMENT

☐ Notify me of follow-up comments by email.

☐ Notify me of new posts by email.

---

## CATEGORIES

[Apple](#)  
[CSS](#)  
[Design](#)  
[Google](#)  
[Linux](#)  
[Microsoft](#)  
[Opinion](#)  
[Photoshop](#)  
[PHP](#)  
[Python](#)  
[Software development](#)  
[Tools](#)  
[Uncategorized](#)  
[Version Control](#)  
[Visual Studio](#)

## TAGS

[Apache](#) [API](#) [Backup](#) [C#](#) [CLI](#) [Cloud](#) [Code](#) [communication](#) [connectivity](#)  
[CrashPlan](#) [Cron](#) [Cross-Platform](#) [database](#) [developer tool](#) [domain name](#) [email](#) [file](#)  
[storage](#) [free](#) [ftp](#) [GET](#) [GIT](#) [Github](#) [HTTP](#) [IDE](#) [Microsoft](#) [mysql](#) [Open source](#)  
[planning](#) [Python3](#) [request](#) [script](#) [server](#) [ssh](#) [subdomain](#) [Tasks](#) [To-do](#) [trello](#)  
[Tutorial](#) [Ubuntu](#) [Ubuntu server](#) [Virtual machines](#) [Visual Studio](#)  
[Web Server](#) [Windows](#) [Wunderlist](#)

## RECENT POSTS

[Use STATA at home \[for KU Leuven students\]](#)

[Deveo](#), a special version control solution for teams

[Pixabay](#): The best stock photos, totally free

[Google Keep](#): An amazing to-do list manager and note keeper

[Great self-hosted Github alternative: Gitlab](#)

Free WordPress Theme designed by [GavickPro](#), modified by [Daan Lenaerts](#)