

Stack Overflow is a community of 4.7 million programmers, just like you, helping each other.

Join them; it only takes a minute:

[Sign up](#)

Join the Stack Overflow community to:



Ask programming questions



Answer and help your peers



Get recognized for your expertise

Unable to retrieve files from send_from_directory() in flask

Work on work you love. From home.

stackoverflow
CAREERS

I have a html file which references static object like this

```

<link href="css/bluestrap.css" rel="stylesheet">
```

Hence the browser tries to call this via and flask fails to do so

```
http://127.0.0.1:5000/img/snacks.png
```

There are lots of such references across multiple files hence changing the references is not possible. How do i serve these static files from FLASK

I have copied all these static files to the 'static' folder and tried this

```
@app.route('/<path:filename>')
def send_file(filename):
    return send_from_directory('/static', filename)
```

However this does not work, Is there any other way to do this ? or what am i doing wrong ?

python flask

asked Jul 16 '13 at 16:13



Rijul Jain

31 ● 1 ● 3

4 Answers

In production, you don't want to server static files using the flask server. I suggest you use a proper webserver to do that.

For dev, since you don't want to use url_for, you can try to initialize your flask app as below. This way, flask knows where your static files are.

```
app = Flask(__name__, static_folder='static')

@app.route('/<path:filename>')
def send_file(filename):
    return send_from_directory(app.static_folder, filename)
```

See this post with a lot of info [Static files in Flask - robot.txt, sitemap.xml \(mod_wsgi\)](#)

edited Jul 16 '13 at 16:50

answered Jul 16 '13 at 16:19



codegeek

7,575 ● 5 ● 26 ● 43

There are too many references of images / css files to change them using url_for function , that is why i don't want to change them . Is there no way to not change the html file and still serve them from python ? –

Rijul Jain Jul 16 '13 at 16:25

PC PyCharm

PYTHON IDE
FOR PROFESSIONAL
DEVELOPERS

Get it now

JET
BRAINS

Don't use Flask's built-in server in production. It is for development only! And don't use Flask to serve static assets. It's slow! In production use a webserver in front of Flask like apache2, nginx or lighttpd. These servers are able to rewrite a URL and serve static assets.

How to deploy Flask: <http://flask.pocoo.org/docs/deploying/>

How to rewrite a URL: [apache2](#), [nginx](#), [lighttpd](#).

answered Jul 16 '13 at 19:14



dAnjou

1,564 ● 2 ● 11 ● 28

If you look at the docs for `send_from_directory` you'll see that it takes the path to the directory in which the files are held *on disk*. Unless you have your image files saved in a root-level directory named `static`, you'll want to update your file path:

```
send_from_directory("/some/path/to/static", "my_image_file.jpg")
```

That being said, if you are using this for anything that will be under any load, it is better to ensure that your *web server* serves the files, rather than serving static files from your application.

answered Jul 16 '13 at 16:51



Sean Vieira

71.5k ● 12 ● 141 ● 161

My files are stored in a folder called 'img' inside a root level directory called 'static', so effectively its stored at static/img/snacks.png and browser calls 127.0.0.1:5000/img/snacks.png – Rijul Jain Jul 16 '13 at 16:56

@RijulJain - yes, but is that directory located at `/static` on disk? That is, is it next to `usr`, `etc` and so on under `/` in your *file system* (not your webserver root)? – Sean Vieira Jul 16 '13 at 16:59

That is in my webserver root, how do i give path relative to my webserver root? – Rijul Jain Jul 16 '13 at 17:22

I think a better way to do this would be:

```
import flask

# ...

@app.route('/img/<fname>')
def legacy_images(fname):
    return flask.redirect(flask.url_for('static', filename='img/' + fname), code=301)
```

Instead of sending the files on two different locations, this would do a permanent redirect to the proper URL. As others have said, it's also a good idea to serve static files directly with nginx or Apache.

answered Jul 17 '13 at 14:15



Markus Unterwaditzer

3,012 ● 8 ● 33