

Simple HTTP Server 403

You are tasked with creating a simple HTTP server that can provide status codes 200, 404, and 403. The server should be able to generate a simple HTML page with the status message as the title and body content.

The server should have the following functionalities:

- If the server receives a request and the server can handle it, it should return a status code of 200 and the HTML page should display 'Hello world!'.
- If the server receives a request for a resource that does not exist, it should return a status code of 404 and the HTML page should display '404 Not found'.
- If the server receives a request for a resource that the client does not have access rights, it should return a status code of 403 and the HTML page should display '403 Forbidden'. The forbidden page is hello.html.

The serve function should start a server that listens for incoming connections and responds with the appropriate status code and HTML content. The server should be able to handle multiple connections simultaneously using select module. The server should be able to handle different types of HTTP requests. The server should be able to send the appropriate status code and HTML content based on the request.

Input: no input.

Output (without unit test):

```
data: b'GET index.html HTTP/1.1\r\nHost: localhost\r\n\r\n'
request header: ['GET index.html HTTP/1.1', 'Host: localhost', '', '']
data: b''
```

Output (with unit test):

```
Testing create_server ...
bind called with: call(('localhost', 8080))
listen called with: call(5)
```

```
Testing get_content ...
test attribute passed: Hello world! is in
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Hello world!</title>
</head>
<body>
Hello world!
</body>
</html>
```

test attribute passed: 404 Not found is in

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>404 Not found</title>
</head>
<body>
404 Not found
</body>
</html>
```

test attribute passed: 403 Forbidden is in

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>403 Forbidden</title>
</head>
<body>
403 Forbidden
</body>
</html>
```

Testing get_header ...

request header: ['GET /index.html HTTP/1.1', 'Host: localhost', '', '']

test attribute passed: /index.html is equal to /index.html

Testing process_requests ...

accept called with: call()

test attribute passed: call(('localhost', 8080)) is equal to call(('localhost', 8080))

test attribute passed: 5 is equal to 5

test attribute listen passed: True is True

test attribute accept passed: True is True

test attribute close passed: True is True