



Struttura della cartella

server_flask.py
static/
home.html
errore.html

Esempio di pagina home (static/home.html)

```
<!DOCTYPE html>
<html>
<body>
<h1>Home</h1>
<p>Se vedi questo, il file HTML è stato servito.</p>
<a href="/setcookie">Imposta cookie</a><br>
<a href="/showcookie">Mostra cookie</a><br>
<a href="/go/errore">Redirect 302 a errore.html</a>
</body>
</html>
```

Esempio di pagina di errore (static/errore.html)

```
<!DOCTYPE html>
<html>
<body>
<h1>Errore</h1>
<p>Questa è la pagina di errore servita via param.</p>
<a href="/html/home">Torna alla home</a>
</body>
</html>
```

Codice del server server_flask.py

```

from flask import Flask, make_response, redirect, request
from pathlib import Path

app = Flask(__name__)
STATIC_DIR = Path("static")

# -----
# Helper: serve file HTML
# -----
def serve_html(filename: str):
    path = STATIC_DIR / filename
    if not path.exists():
        return make_response("<h1>404 - File non trovato</h1>", 404)

    html = path.read_text(encoding="utf-8")
    resp = make_response(html, 200)
    resp.headers["Content-Type"] = "text/html; charset=utf-8"
    return resp

# -----
# 1) Pagina HTML via parametro
#    /html/home -> static/home.html
# -----
@app.route("/html/<page>")
def html_page(page):
    return serve_html(f"{page}.html")

# Shortcut: home
@app.route("/")
def root():
    return serve_html("home.html")

# -----
# 2) Redirect 302 verso una pagina HTML
#    /go/errore -> 302 -> /html/errore
# -----
@app.route("/go/<target>")
def go(target):
    return redirect(f"/html/{target}", code=302)

```

```

# -----
# 3) Set cookie
#   /setcookie -> imposta cookie "sessionid"
# -----
@app.route("/setcookie")
def setcookie():
    resp = serve_html("home.html")
    # Impostiamo un cookie semplice
    resp.set_cookie("sessionid", "abc123", path="/")
    return resp

# -----
# 4) Lettura cookie
#   /showcookie -> mostra tutti i cookie ricevuti
# -----
@app.route("/showcookie")
def showcookie():
    # Tutti i cookie parsati da Flask
    cookies = request.cookies

    html = "<h1>Cookie ricevuti</h1><pre>"
    for k, v in cookies.items():
        html += f"{k} = {v}\n"
    html += "</pre><a href='/'>Torna alla home</a>"

    resp = make_response(html, 200)
    resp.headers["Content-Type"] = "text/html; charset=utf-8"
    return resp

# -----
# Avvio server
# -----
if __name__ == "__main__":
    STATIC_DIR.mkdir(exist_ok=True)
    app.run(host="0.0.0.0", port=5000, debug=True)

```

Cosa fa esattamente

- GET / → serve static/home.html
- GET /html/home → serve static/home.html
- GET /html/errore → serve static/errore.html
- GET /go/errore → 302 → /html/errore
- GET /setcookie → serve home.html impostando Set-Cookie: sessionid=abc123
- GET /showcookie → legge request.cookies e mostra i cookie

Obiettivo

Fare due server:

- uno sulla porta 9000: serve pagine statiche (protette e non)
- uno sulla porta 10000: identity manager

Realizzare la gestione del login e dell'accesso a pagine protette