

Esercizi dalle pagine 29, 31, 34 e 36

Esercizio pagina 29

1. **Aprire un secondo terminale e rilanciare il server. Funziona? Perché?**

Risposta: No, non funziona perché il socket è già occupato dall'altro processo.

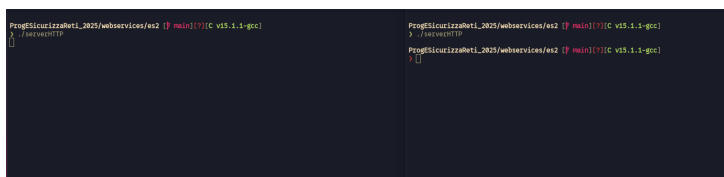


Figure 1: Tentativo di avviare due istanze del server

2. **Aprire il browser e impostare la URL `http://127.0.0.1:8000/`. Cosa si vede sul browser e sul terminale?**

Risposta: Sul browser viene mostrata la pagina di prova, mentre nel terminale compare il log della richiesta.

3. **Impostare la URL `http://localhost:8000/`. Cosa cambia?**

Risposta: Cambia solo il valore dell'header Host nella richiesta HTTP, che diventa localhost.

Esercizio pagina 31

1. **Eseguire il server web e aprire `form-get.html`. Cosa si vede?**

Risposta: Viene visualizzato sempre “Hello World, this is a very simple HTML document.”, cambia solo la pagina richiesta (`/form-get.html`).



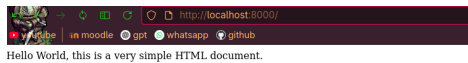
```
ProgSicurezza021_2025/webservices/es2 [P main][?]C v15.1.1-gcc
> ./serverHTTP
GET / HTTP/1.1
Host: 127.0.0.1:8000
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br, zstd
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: none
Sec-Fetch-User: ?1
Priority: u=0, i

[SERVER] sessione HTTP completata

GET /favicon.ico HTTP/1.1
Host: 127.0.0.1:8000
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
Accept: image/avif,image/jxl,image/webp,image/png,image/svg+xml,image/*;q=0.8,*/*;q=0.5
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br, zstd
Connection: keep-alive
Referer: http://127.0.0.1:8000/
Sec-Fetch-Dest: image
Sec-Fetch-Mode: no-cors
Sec-Fetch-Site: same-origin
Priority: u=6

[SERVER] sessione HTTP completata
```

Figure 2: Visualizzazione di 127.0.0.1:8000 sul browser (sinistra) e sul terminale (destra)



```
[SERVER] sessione HTTP completata

GET / HTTP/1.1
Host: localhost:8000
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br, zstd
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: none
Sec-Fetch-User: ?1
Priority: u=0, i

[SERVER] sessione HTTP completata

GET /favicon.ico HTTP/1.1
Host: localhost:8000
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
Accept: image/avif,image/jxl,image/webp,image/png,image/svg+xml,image/*;q=0.8,*/*;q=0.5
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br, zstd
Connection: keep-alive
Referer: http://localhost:8000/
Sec-Fetch-Dest: image
Sec-Fetch-Mode: no-cors
Sec-Fetch-Site: same-origin
Priority: u=6

[SERVER] sessione HTTP completata
```

Figure 3: Visualizzazione di localhost:8000 sul browser (sinistra) e sul terminale (destra)

Esercizio pagina 34

1. Eseguire il server web e aprire form-post.html. Cosa si vede?
Risposta: Uguae a prima, viene mostrata la pagina di prova, cambia solo l'URL (/form-post.html).

Esercizio pagina 36

1. Modificare serverHTTP.c per restituire le pagine richieste (form-get.html, form-post.html). Cosa succede?
Risposta: Le pagine corrispondenti vengono effettivamente restituite in base al path richiesto.

```
GET /form-get.html HTTP/1.1
Host: 127.0.0.1:8080
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br, zstd
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: none
Sec-Fetch-User: ?1
Priority: u=0, i

HTTP/1.1 200 OK

<html><head><title>An Example Page</title></head><body>Hello World, this is a very simple HTML document.</body></html>
```

Figure 4: Esecuzione con form-get.html

```
GET /form-post.html HTTP/1.1
Host: 127.0.0.1:8080
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br, zstd
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: none
Sec-Fetch-User: ?1
Priority: u=0, i

HTTP/1.1 200 OK

<html><head><title>An Example Page</title></head><body>Hello World, this is a very simple HTML document.</body></html>
```

Figure 5: Esecuzione con form-post.html

```
GET /form-get.html HTTP/1.1
Host: 127.0.0.1:8000
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br, zstd
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: none
Sec-Fetch-User: ?1
Priority: u=0, i

HTTP/1.1 200 OK
Content-Type: text/html; charset=UTF-8
Content-Length: 584

<!DOCTYPE html>
<html>
<body>

<h2>The method Attribute</h2>

<p>This form will be submitted using the GET method:</p>

<form action="http://localhost:8000/action" target="_blank" method="get">
  <label for="fname">First name:</label><br>
  <input type="text" id="fname" name="fname" value="John"><br>
  <label for="lname">Last name:</label><br>
  <input type="text" id="lname" name="lname" value="Doe"><br>
  <input type="submit" value="Invia">
</form>

<p>After you submit, notice that the form values is visible in the address bar of the new browser tab.</p>

</body>
</html>
```

Figure 6: Risultato dopo la modifica del server