

Traccia 1 – Web Server + Sito Web Statico

Funzionamento del Server

Per la creazione del server HTTP ho utilizzato i moduli standard `http.server` e `socketserver`, per semplificare la gestione delle richieste HTTP.

Il server dopo essere stato avviato aspetta una richiesta su `localhost:8080` poi:

- Quando riceve una richiesta HTTP GET, viene analizzato l'URL per determinare il percorso del file
- Se il file richiesto esiste, il server invia una risposta HTTP 200 al client ed il contenuto del file richiesto
- Se il file non esiste allora il server invia una risposta HTTP 404 al client che indica che il file non è stato trovato
- In caso il file richiesto sia già presente nella cache del browser e il file non è cambiato dall'ultima richiesta, il server invia una risposta HTTP 304 al client che indica che il file non è cambiato quindi il server risparmia banda e non rinvia il file.
- In caso non venga richiesto nessun file e il client abbia semplicemente fatto una richiesta GET a `localhost:8080` il server invia una risposta HTTP 301 redirezionando a `index.html` di cui il cliente farà richiesta al server che risponderà con HTTP 200

Ho utilizzato la classe `socketserver.ThreadingTCPServer` per utilizzare la programmazione concorrente a livello di thread per gestire più richieste contemporaneamente.

Per quanto riguarda il log delle richieste ricevute ho utilizzato il log automatica fatto da `SimpleHTTPRequestHandler`.

Funzionamento pagine HTML

Ci sono 4 pagine HTML ognuna delle quali accede:

- ad un foglio di stile "`style.css`" con classi diverse per avere uno stile diverso per ogni pagina
- ad una icona personalizzata per ogni pagina

Ogni pagina presenta:

- un immagine `jpg`

- un bottone che viene utilizzato per fare richiesta di altre pagine o tornare alla pagina iniziale

Testing

Per testare il server ho utilizzato il modo più semplice, cioè ho aperto un browser (nel mio caso Edge) e ho inserito l'indirizzo di loopback con la porta 8080 nella barra degli URL. Poi si viene reindirizzati nella pagina "index.html" nella quale sono presenti dei bottoni per inviare le richieste GET delle altre pagine al server. Nel mentre tramite il terminale controllavo i log delle richieste ricevute dal server.

Avvio Server

```
PROBLEMI  OUTPUT  CONSOLE DI DEBUG  TERMINALE  PORTE
PS C:\Users\marcp\Documents\GitHub\WebServer-SitoWebStatico> & C:/Users/marcp/anaconda3/python.exe c:/Users/marcp/Documents/GitHub/WebServer-SitoWebStatico/TCP_Server.py
Server ready to serve on http://127.0.0.1:8080 ...
```

Visione del Server

Richiesta di "index.html"

```
127.0.0.1 - - [01/Jun/2025 18:01:43] "GET /www/index.html HTTP/1.1" 200 -
127.0.0.1 - - [01/Jun/2025 18:01:43] "GET /www/style.css HTTP/1.1" 200 -
127.0.0.1 - - [01/Jun/2025 18:01:44] "GET /resources/question_marks.jpg HTTP/1.1" 200 -
127.0.0.1 - - [01/Jun/2025 18:01:44] "GET /resources/favicon_index.ico HTTP/1.1" 200 -
```

Richiesta di "gundam.html"

```
127.0.0.1 - - [01/Jun/2025 18:02:12] "GET /www/gundam.html HTTP/1.1" 200 -
127.0.0.1 - - [01/Jun/2025 18:02:12] "GET /www/style.css HTTP/1.1" 200 -
127.0.0.1 - - [01/Jun/2025 18:02:13] "GET /resources/rx_78_2.jpg HTTP/1.1" 200 -
127.0.0.1 - - [01/Jun/2025 18:02:13] "GET /resources/favicon_gundam.ico HTTP/1.1" 200 -
```

Richiesta già fatta di "index.html"

```
127.0.0.1 - - [01/Jun/2025 18:03:44] "GET /www/index.html HTTP/1.1" 304 -
```

Richiesta di file inesistente

```
127.0.0.1 - - [01/Jun/2025 18:05:59] "GET /hey HTTP/1.1" 404 -
127.0.0.1 - - [01/Jun/2025 18:05:59] "GET /favicon.ico HTTP/1.1" 404 -
```

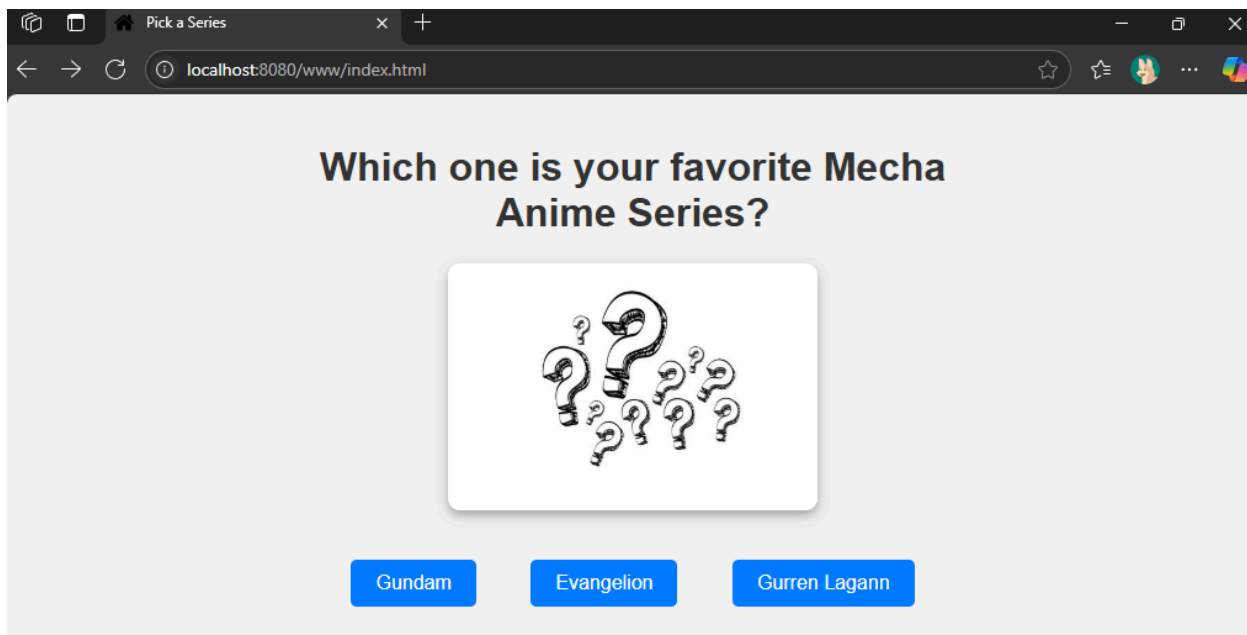
Nome: Marco

Cognome: Pertegato

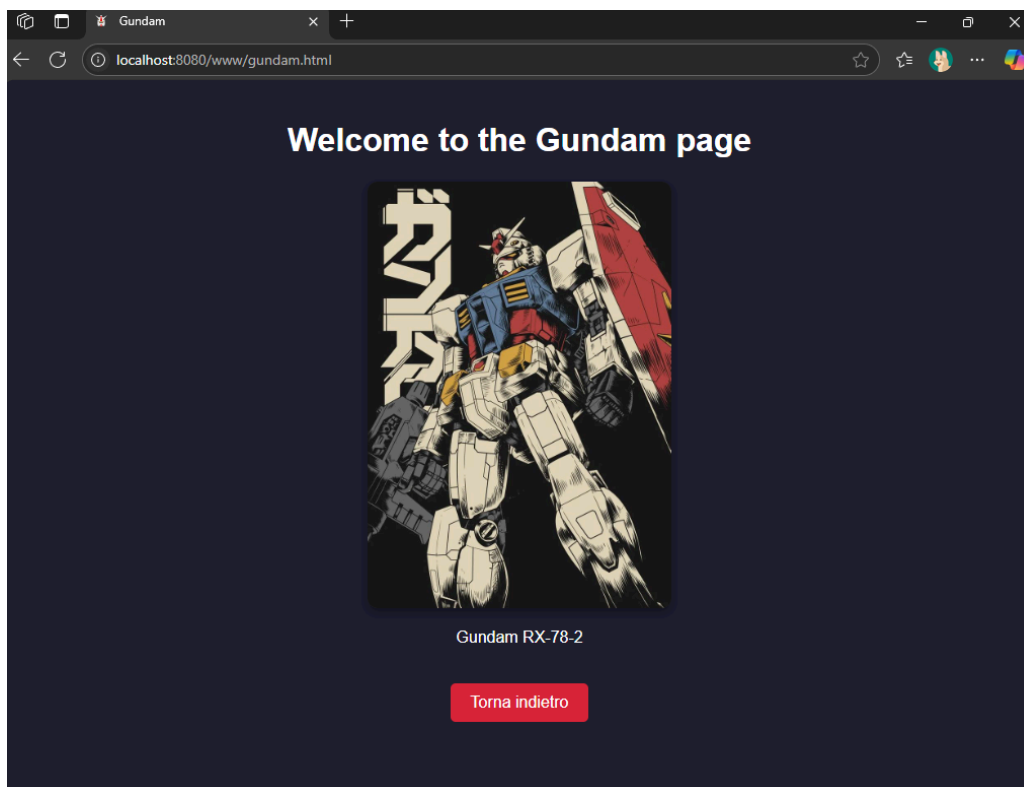
Matricola:0001123050

Visione del Server

Richiesta "index.html"



Richiesta "gundam.html"



Richiesta di file inesistente



Considerazioni Aggiuntive

Quando il client richiede una pagina html che è già salvata in cache il server risponde solo con il codice 304 solamente alla richiesta pagina html e il client quindi non richiede neanche la pagina "style.css" l'immagine jpg e l'icona.

Quando accedo semplicemente a localhost:8080 e il server redireziona a "index.html" nei log non viene mostrata la richiesta a "/" e la risposta con codice 301 ma direttamente la richiesta a "index.html"