

## Progettazione di Servizi Web e Reti di Calcolatori

Politecnico di Torino – Prof. Andrea Atzeni

### AA 2023-2024 esercitazione di laboratorio n. 7

Per scrivere file HTML/CSS/JS/PHP occorre usare un normale editor di testo, presente localmente o direttamente l'ambiente di sviluppo "remote Code Server", disponibile nel cloud del corso alla URL (specifica per ciascuno studente) avente la seguente forma generale:

`https://<nome_server>/<sMatricola>/dev`

I file HTML possono avere indifferentemente estensione ".htm" o ".html", i file CSS ".css", quelli Javascript ".js" e quelli PHP ".php".

Se le pagine sono di tipo statico e sono scritte correttamente con link relativi, allora è possibile posizionarle in una qualunque cartella del filesystem e navigarle direttamente aprendole con un qualunque browser. Questa esercitazione richiede invece la creazione anche di pagine dinamiche lato server in PHP. Per usarle correttamente, bisogna accedervi tramite un interprete PHP, che è integrato nel server HTTP presente nell'ambiente cloud del corso.

E' errato accedere direttamente al corrispondente file su disco locale, come fatto nelle esercitazioni precedenti per le pagine statiche HTML. Invece, è sempre necessario passare attraverso un server web che esegua il codice richiamando l'interprete PHP.

In alcuni ambienti di sviluppo moderni, come Visual Studio Code, questo può essere fatto usando un'estensione dell'ambiente di sviluppo che contenga un web server e l'interprete PHP (ad esempio, con VSC è possibile installare l'estensione "PHP Server").

Negli esercizi che prevedono l'introduzione di numeri, verificarne il corretto funzionamento non solo introducendo dati validi (es. "5", "-3", "+7", "2.3", "5e3") ma anche agendo volutamente in modo scorretto, es. "5mila", "cinquemila", "5 mila", "2,3", o inoltrando il form senza aver inserito tutti i dati richiesti.

Si suggerisce di sviluppare ogni esercizio prima in forma base (ossia senza particolari controlli di errore) e quindi modificarlo per trattare anche il caso di introduzione di dati errati.

Si ricorda che per validare il codice HTML delle pagine contenenti elementi Javascript client-side è necessario usare un validatore installato all'interno del browser. A questo scopo, si consiglia l'uso del plugin "HTML validator" (come nelle precedenti esercitazioni).

Lo stesso *modus operandi* si applica anche alla validazione delle pagine PHP: poiché il codice HTML viene generato dall'esecuzione dello script lato server, non è sensato validare con un validatore HTML direttamente la pagina PHP, in quanto il codice HTML finale non è disponibile sul server ma solo sul client (dopo l'esecuzione dello script server side appunto).

Per ciascun esercizio, si suggerisce di realizzare l'invio dei dati dei form prima tramite il metodo GET (particolarmente utile per fare il debug del sito perché visualizza all'interno della URL i parametri passati ed i relativi valori) e poi tramite il metodo POST (procedura più professionale perché nasconde i dati all'utente e non lascia traccia dei parametri e dei valori nel log del server HTTP).

Per ogni esercizio in cui ciò sia possibile e sensato, verificare lato client, prima dell'invio al server, che i dati inseriti rispettino le specifiche indicate nel testo dell'esercizio (es. siano numeri interi) ed in caso contrario i dati non siano inviati al server ma sia visualizzato un messaggio di errore.

**N.B.** Questo controllo "lato client" *non* implica la possibilità di non sviluppare gli opportuni controlli anche lato server. Infatti la pagina PHP potrebbe essere acceduta con intenti malevoli, ricevendo quindi valori errati o contraffatti; ciò rende quindi indispensabili i controlli lato server.

### Esercizio 7.1

Creare una pagina HTML che scelga in modo casuale un valore N intero, compreso tra 2 e 13 e quindi fornisca nel corpo della pagina in forma di tabella la tavola pitagorica di dimensione N.

**Suggerimento:** per scegliere il valore casuale usare la funzione `rand()` (<https://www.php.net/manual/it/function.rand.php>)

## Esercizio 7.2

Creare una pagina HTML contenente un form con un campo di input testuale e due pulsanti, rispettivamente *Cancella* e *Invia*.

Realizzare quindi una pagina PHP che riceva i dati dal form e crei una pagina avente come titolo di primo livello “Testo ricevuto” seguito da un paragrafo col contenuto del campo di input testuale.

## Esercizio 7.3

Realizzare una pagina HTML contenente un form con due campi di input testuale (per inserire due numeri interi) e due pulsanti, rispettivamente *Cancella* e *Somma*. Alla pressione del pulsante *Somma* i numeri introdotti vanno inviati ad una pagina PHP. La pagina PHP deve creare una pagina di risposta contenente la somma dei due numeri nella forma “A + B = S” ed un link per tornare alla pagina iniziale, associato al testo “Torna alla pagina iniziale”. Alla pressione del pulsante *Cancella* i campi devono essere ripuliti da eventuali numeri introdotti.

## Esercizio 7.4

Realizzare una pagina HTML contenente un form con un menù a tendina per scegliere un numero intero N compreso tra 1 e 10 ed un pulsante “Crea tabella”.

Realizzare quindi una pagina PHP che sia attivata alla pressione del pulsante, riceva i dati del form e crei una pagina di risposta contenente la tabella dei quadrati e dei cubi da 1 al valore N introdotto nel form (ogni riga della tabella deve contenere un numero K, il suo quadrato ed il suo cubo).

## Esercizio 7.5

Realizzare una pagina HTML contenente un form con:

- un campo per introdurre il nome di un prodotto (come stringa di testo)
- un campo per introdurre il prezzo netto di un prodotto (come numero intero oppure reale, con al massimo due cifre dopo la virgola)
- un radio button per scegliere tra tre diverse aliquote IVA (minima, ridotta, ordinaria)
- un pulsante per inviare i dati al server, e far calcolare il prezzo IVA inclusa, presentato con due cifre dopo la virgola

Realizzare quindi una pagina PHP che riceva i suddetti dati e quindi calcoli e visualizzi il nome del prodotto ed il prezzo, IVA inclusa, presentato con due cifre dopo la virgola. Il calcolo dell’IVA verrà fatto in base alla corrispondenza: minima=4%, ridotta=10%, ordinaria=22%.

NOTA: dopo aver realizzato (e verificato) il funzionamento della pagina con dati corretti, aggiungere controlli che comprendano quando un dato inserito non sia del tipo richiesto (es. una stringa alfabetica per un valore numerico) ed in questi casi segnalino all’utente il problema.

## Esercizio 7.6

Modificare il codice PHP dell’esercizio precedente aggiungendo una ulteriore checkbox nella pagina HTML per richiedere la “modalità debug” che – se selezionata – faccia sì che nella pagina presentata dal server, dopo il calcolo dell’IVA, sia contenuta anche una tabella con nome, tipo e valori dei campi inviati tramite il form.