

Progettazione di Servizi Web e Reti di Calcolatori

Politecnico di Torino – Prof. Antonio Lioy

AA 2019-2020, esercitazione di laboratorio n. 7

Per scrivere file HTML/CSS occorre usare un normale editor di testo (es. Notepad++). Per file HTML si può avere indifferentemente estensione “.htm” o “.html”. Per CSS estensione “.css”. Per i file javascript estensione “.js”

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.

Se le pagine sono di tipo dinamico (server-side) allora occorre installarle su un server web vero e proprio, cosa che verrà fatta in una prossima esercitazione.

Per tutti gli esercizi proposti, verificare la qualità grafica della soluzione realizzata: a) visualizzare sempre le pagine con almeno due browser diversi (es. IE e Firefox); b) allargare e restringere la finestra del browser; c) cambiare la risoluzione dello schermo (es. 800x600, 1024x768).

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:

- introducendo dati errati (es. “5mila”, “cinquemila”, “5 mila”, “2,3”)
- premendo OK nel pop-up senza aver introdotto dati
- chiudendo il pop-up di input tramite il pulsante Cancel
- chiudendo il pop-up di input tramite il pulsante di chiusura finestra (X in alto a destra)

Si suggerisce di sviluppare ogni esercizio prima in forma base (ossia senza particolari controlli di errore) e quindi facendo le necessarie modifiche 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” disponibile (per Mozilla Firefox e Google Chrome) alla seguente URL:

<http://users.skynet.be/mgueury/mozilla/>

Prima di usare tale plugin, leggere attentamente le istruzioni sia per configurarlo sia per usarlo correttamente (soprattutto quando le pagine contengono codice Javascript):

http://users.skynet.be/mgueury/mozilla/user_guide.html (solo per Mozilla Firefox)

Su Google Chrome, è possibile accedere al plugin con i seguenti passi: effettuare click destro sulla pagina da validare, cliccare su “Ispeziona” e da qui accedere al tab “HTML validator”.

Esercizio 7.1

Creare una pagina HTML che mostri la data ed ora attuale sia secondo il fuso orario del computer su cui viene visualizzata sia come UTC.

Esercizio 7.2

Creare una pagina HTML che richieda tramite pop-up il diametro D di un cerchio e poi scriva nel corpo della pagina “Un cerchio di diametro D ha area A e circonferenza C” ove A e C sono i valori calcolati tramite un programma Javascript. Tutti i valori devono essere presentati con tre cifre frazionarie. Suggerimento: per il valore di π si usi la costante fornita dall'oggetto Math.

Esercizio 7.3

Creare una pagina HTML che contenga un titolo di primo livello “Esercizio” e poi la frase “Tanto va la gatta al lardo che ci lascia lo zampino” seguita da due paragrafi, contenenti rispettivamente le parole GIORNO e NOTTE. Quando si fa clic su GIORNO la pagina deve essere mostrata con sfondo bianco e testo nero ed il titolo di primo livello deve essere “Adesso è giorno” mentre quando si fa clic su NOTTE la pagina deve essere mostrata con sfondo nero e testo bianco ed il titolo di primo livello deve essere “Adesso è notte”.

Esercizio 7.4

Creare una pagina HTML che contenga un campo di testo per l'inserimento di una stringa da parte dell'utente e quindi – quando viene premuto un apposito pulsante – visualizzi in una finestra di pop-up la stringa introdotta convertita in minuscolo.

Esercizio 7.5

Realizzare una pagina HTML con un menù a scelta multipla mutuamente esclusiva, in cui a ciascuna scelta sia associato un colore (rosso, verde, giallo, bianco). Effettuando una delle scelte disponibili, il colore di sfondo della pagina deve cambiare di conseguenza (es. diventare rosso).

Esercizio 7.6

Modificare l'esercizio precedente, permettendo all'utente la scelta di un colore arbitrario per lo sfondo.

Esercizio 7.7

Creare una pagina HTML con un form in grado di accettare due numeri naturali come input e quindi – quando viene premuto un apposito pulsante – mostri in un campo di testo (non modificabile dall'utente) la somma dei due numeri.

Esercizio 7.8

Modificare la pagina sviluppata nell'esercizio 7.7 eliminando il pulsante e facendo in modo che la somma dei numeri cambi automaticamente non appena cambia uno dei due numeri.

Esercizio 7.9

Creare una pagina HTML con un form che permetta l'introduzione della data di nascita di un utente, in formato (dd/mm/yyyy), e che indichi in output in quale giorno della settimana cadde il Natale in quell'anno.

Esercizio 7.10

Sviluppare una funzione Javascript chiamata “Maggiore” che accetti una quantità imprecisata di numeri come parametri e restituisca il maggiore di essi. Memorizzare la funzione in un file “funzioni.js”.

Creare quindi una pagina HTML che richiami il file “funzioni.js” ed usi la funzione “Maggiore” per calcolare il massimo tra cinque numeri introdotti come campi distinti di un form. Il valore del massimo deve essere mostrato in una finestra di pop-up attivata quando si preme un apposito bottone etichettato “Trova il massimo”. Il form deve anche contenere un pulsante Cancella per eliminare il contenuto dei cinque campi di input.

Esercizio 7.11

Creare una pagina HTML che permetta tramite un form con tre menù a tendina l'inserimento delle quantità di tre diversi beni acquistabili (memoria SD 8GB, chiavetta USB 8GB, DVD-R), ciascuno variabile da 0 a 10.

Quando l'utente preme un apposito pulsante la pagina deve calcolare il costo totale dei beni. Il prezzo unitario dei tre beni è rispettivamente 15.50, 18.00 e 2.00 Euro. Il risultato deve essere visualizzato in un campo di testo (non modificabile dall'utente) presente nel form medesimo.

La pagina deve anche contenere un pulsante “Azzera” che permetta di riportare in un colpo solo a zero i valori di tutti i menù a tendina (e quindi anche del totale).

Esercizio 7.12

Creare una pagina HTML che permetta, tramite un form, di inserire un testo che deve essere visualizzato in colore rosso in un’apposita sezione della pagina intitolata “Testo inserito”. Tale sezione inizialmente deve contenere in colore nero il testo “Nessun testo inserito”.

Il testo inserito deve essere visualizzato quando viene premuto un pulsante OK (corrispondente all’azione Submit) mentre alla pressione di un pulsante Cancella (corrispondente all’azione Reset) tutta la pagina deve tornare al suo stato iniziale.

Esercizio 7.13

Creare una pagina HTML che contenga un form con:

- due campi di testo per l’inserimento di due numeri floating point contenenti esattamente due cifre dopo la virgola ed opzionalmente preceduti dal segno + o -;
- un gruppo di pulsanti per scegliere una sola operazione da svolgere tra somma, differenza, prodotto e quoziente;
- un pulsante per calcolare, tramite uno script JS, il risultato dell’operazione (che deve comparire in un apposito campo non editabile dall’utente);
- un pulsante per riportare il form allo stato iniziale.

Esercizio 7.14

Modificare la pagina realizzata nell’esercizio 7.14 controllando che i numeri introdotti siano interi compresi tra 0 e 99, eventualmente preceduti dal segno + o -.

Esercizio 7.15

Creare una pagina che permetta l’introduzione di uno *username* lungo almeno 5 caratteri e che contenga solo caratteri alfabetici, il punto o la virgola. Lo username deve inoltre contenere almeno un carattere maiuscolo ed uno minuscolo. Nel caso lo username introdotto non rispetti questa specifica, deve comparire un pop-up che segnala il problema specifico ed elenca gli eventuali caratteri non ammessi.

Esercizio 7.16

Creare una pagina che permetta l’introduzione di una temperatura scritta nel seguente formato: il segno + o -, massimo 3 cifre per la parte intera, il punto e quindi una sola cifra per la parte frazionaria. Esempi di temperature scritte correttamente: +3.2, -273.0, +27.4. Esempi di temperature indicate in modo errato: -273, 27.4, 0. Nel caso la temperatura introdotta non rispetti questa specifica, deve essere segnalato errore tramite una finestra di pop-up.

Esercizio 7.17

Creare una pagina con un form contenente i seguenti dati:

- un nome, costituito al più da 10 caratteri alfabetici, di cui il primo sia obbligatoriamente maiuscolo ed i restanti minuscoli;
- un’età (valore numerico compreso tra 0 e 199)
- un numero di telefono, costituito da un prefisso (due o tre cifre, di cui la prima sia zero) seguito dal carattere ‘-’ e quindi da un numero (sei o sette cifre) che può opzionalmente contenere un altro carattere ‘-’ dopo le prime tre cifre

ed un bottone *Verifica* alla cui pressione siano segnalati eventuali errori nella compilazione dei campi precedenti oppure compaia il messaggio “dati corretti”.

Esercizio 7.18

Creare una pagina con un campo di testo per inserire una data nel formato gg-mm-aaaa, dove il giorno ed il mese possono essere scritti con una o due cifre.

Quando la data viene inserita o modificata la pagina deve essere aggiornata in tre campi di testo (non editabili dall'utente) che indicano rispettivamente il giorno, il mese e l'anno introdotti, oppure un'indicazione di errore se il giorno non è compreso tra 1 e 31, o il mese non è compreso tra 1 e 12, oppure il giorno non è coerente con il mese (es. 31 febbraio).

Si suggerisce l'uso del metodo 'split' per estrarre giorno, mese ed anno dall'espressione regolare che controlla lessico e sintassi della data introdotta.