

# Esercizi su JS e DOM per verifica

## Esercizio 1

- Partendo da una pagina HTML che contiene diversi paragrafi, si vuole implementare una funzione da eseguire in caso di evento 'click' sul testo del paragrafo che mostri in una finestra di dialogo contenente il numero di occorrenze della lettera a minuscola per tale elemento  $< p >$ .
- Una volta svolto il punto precedente, si codifichi una seconda versione della funzione che permetta all'utente di scegliere il carattere da contare (es. mediante finestra di dialogo).
- *Cosa dovresti cambiare nel codice del primo punto per eseguire la funzione che dà scelta all'utente?*

## Esercizio 2

- Partendo da una pagina HTML che contiene del testo, si sviluppi uno script in grado di invertire il contenuto del paragrafo su cui l'utente ha cliccato.
- Si realizzi una seconda versione della funzione con un limite di 3 inversioni totali sulla pagina.
- *Anche qui per testare puoi "scollegare" l'esecuzione della prima funzione e far gestire l'evento dalla seconda.*

## Esercizio 3

- Si realizzi una pagina web che metta a disposizione dell'utente una sandbox HTML così organizzata: sulla sinistra è presente una textarea in cui viene digitato del codice HTML (che si presume corretto dal punto di vista sintattico), sotto è presente un bottone che chiama la funzione che "elabora" il codice HTML presente nella textarea, infine a destra è presente un div in cui viene visualizzato il risultato. Il contenuto di questo div dev'essere resettato a ogni chiamata della suddetta funzione.
- *Come potresti modificare lo script per far sì che il codice HTML inserito nella textarea venga ogni volta aggiunto a quello già presente?*

## Esercizio 4

Si vuole realizzare una libreria JS di funzioni per la rimozione di elementi numerici da un array in modi diversi (facendo finta che non esista il metodo *filter*). Tutte le funzioni ricevono come parametri l'array da controllare e il valore **lim** che assumiamo essere sempre correttamente passati (quindi, non facciamo controlli sul fatto che siano davvero numeri).

- **f1**: controlla l'array e, senza modificarlo, restituisce il riferimento a un nuovo array in cui non sono presenti elementi  $< \text{lim}$ ;
- **f2**: controlla l'array e rimuove da questo gli elementi  $< \text{lim}$  e restituisce **true** se ha eliminato almeno un valore, **false** altrimenti.
- **f3**: controlla l'array e, senza modificarlo, restituisce un nuovo array contenente le posizioni degli elementi  $< \text{lim}$  oppure array vuoto se non ce ne sono.

**N.B.** per le funzioni che modificano l'array originale, non dobbiamo preoccuparci se questo sia stato dichiarato come `let` o `const` nel codice chiamante.