

HTML da usare per l'esercizio Javascript

Usare il seguente HTML per i primi tre esercizi (che, si noti, al momento restituisce un errore JavaScript):

```
<!DOCTYPE html>
<html>
  <head>
    <title>Prove Javascript</title>
    <script type="text/javascript">

      </script>
    </head>
    <body>
      <h1>Prove Javascript</h1>
      <p id="para1">Un paragrafo di prova.</p>
      <button type="button" onclick="funzione()">Clicca qui</button>
    </body>
</html>
```

Esercizio 1

Utilizzando il documento HTML specificato sopra, e senza usare jQuery, fare in modo che, facendo click sul pulsante “Clicca qui”, compaia una finestra di alert con scritto “Hello World”.

Esercizio 2

Utilizzando il documento HTML specificato sopra, e senza usare jQuery, fare in modo che, facendo click sul pulsante “Clicca qui”, si chieda all’utente di specificare una stringa (si guardi la funzione `prompt()`), e si cambi il contenuto del paragrafo “para1” con la stringa composta da “Tu hai scritto che ” e quanto fornito dall’utente. Si guardino al proposito le proprietà `getElementsByID` dell’oggetto `document` e `innerHTML` dell’oggetto `Node`.

Esercizio 3

Partendo dal documento HTML specificato sopra si aggiungano due campi numerici e in Javascript, e senza usare jQuery, si faccia in modo che, facendo click sul pulsante “Clicca qui”, si raccolgano i valori dei due campi (proprietà `value` dell’oggetto `Node`), li si sommi e si metta il risultato nel paragrafo “para1”. A tal proposito può essere utile la funzione `parseInt`.

Esercizio 4

È stato predisposto un documento JSON all'indirizzo:

`http://site1825.tw.cs.unibo.it/auto.json`

Utilizzando il documento HTML specificato sopra, e usando jQuery, si carichi via Ajax il file JSON e se ne visualizzi il contenuto sotto forma tabellare nel div di un documento HTML (si può usare Bootstrap).

Esercizio 5

Creare un costruttore Automobile() che ha come parametri un modello (una stringa), una fotografia (un URL) e un prezzo (un numero). Automobile restituisce un oggetto di tipo Automobile con questi come membri interni e in più una proprietà leggibile e scrivibile, inizialmente vuota: il codice (una stringa).

L'oggetto deve anche mettere a disposizione un metodo:

- `mostra()`, che restituisce un frammento HTML che espone le proprietà dell'automobile.

Creare con Bootstrap¹ una semplice struttura HTML a due colonne, che comprenda nella colonna a sinistra i seguenti buttoni:

- “Crea automobile”, che mostra un modale Ok/Cancella che contiene un form con campi per modello, url della fotografia e prezzo giornaliero, e in seguito all’Ok crei un’istanza di Automobile e venga aggiunto ad un array globale `automobili`.
- “Mostra Stato”, che vuota un apposito `div` della colonna di destra e vi pone lo stato attuale dell’array `automobili` chiamando il metodo `mostra()` di ciascun elemento e mettendo dentro al `div` suddetto i frammenti di DOM corrispondenti.

¹ Si aggiunga il seguente codice a <head>:

```
<link rel="stylesheet" href="//maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css">
<script src="//code.jquery.com/jquery-3.2.1.min.js"></script>
<script src="//maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min.js"></script>
```

Esercizio 6

Sulla base dell'esercizio precedente creare altri due costruttori:

- Cliente: ha come parametro un nome (una stringa). Ha una proprietà `inaccessibile`, `possesso`, che può contenere al massimo una `Automobile` ed è inizialmente vuota. Ha i seguenti metodi:
 - `mostra()`, che restituisce un frammento DOM che visualizza le proprietà del Cliente; il DOM inoltre contiene un bottone associato al metodo `noleggia()` e uno associato al metodo `restituisci()`.
 - `noleggia(codice)`, che verifica che non ci siano automobili in possesso, e se è così chiama `noleggia()` dell'Autosalone, e in caso di successo aggiunge l'automobile nella proprietà `possesso`.
 - `restituisci()`, che verifica che ci sia un'automobile in possesso e se così chiama `restituisci()` dell'Autosalone e toglie l'automobile dalla proprietà `possesso`.
- Autonoleggio: è un singoletto (singleton), ovvero un oggetto garantito in copia unica. L'autonoleggio ha due proprietà `inaccessibili`:
 - il parco macchine, che è una lista di istanze di `Automobili` disponibili, inizialmente vuota
 - il parco clienti, che è una lista di istanze di `Clienti`, inizialmente vuotae cinque metodi:
 - `mostraAuto()`, che inserisce in un apposito div un frammento di DOM contenente il parco macchine, ottenuto chiamando iterativamente il metodo `mostra()` di tutte le automobili. L'elenco delle macchine mostra solo le macchine disponibili. Le voci dell'elenco sono inattive.
 - `mostraClienti()`, che inserisce in un apposito div un frammento di DOM contenente il parco clienti, ottenuto chiamando iterativamente il metodo `mostra()` di tutti i clienti del parco clienti.

L'elenco dei clienti mostra tutti i clienti, diversificando se al momento hanno o meno noleggiato un'automobile.

- `acquista(Automobile)`, che inserisce nel parco macchine una Automobile assegnandole un codice (si usi e modifichi il pulsante “Crea Automobile” dell'esercizio precedente);
- `accogli(Cliente)`, che inserisce nel parco clienti un Cliente (si prenda a prestito il codice del pulsante “Crea Automobile” dell'esercizio precedente);
- `noleggia(codice)`, che controlla se l'Automobile con il codice specificato è nel parco macchine, restituendo un errore se ci sono problemi, altrimenti restituisce l'Automobile togliendola dal parco macchine.
- `restituisci(Automobile)`, che re-inserisce nel parco macchine l'Automobile passata come parametro.

Si modifichi l'interfaccia come segue: l'utente rappresenta l'autonoleggiatore e ha a disposizione l'elenco delle macchine disponibili, l'elenco dei clienti, e alcuni pulsanti:

- Cliccando sul pulsante “Acquista Auto” viene chiamato il metodo `acquista()` e al completamento la macchina è inserita nel parco macchine.
- Cliccando sul pulsante “Accogli cliente” viene chiamato il metodo `accogli()` e al completamento viene inserito un nuovo cliente nel parco clienti.
- Le voci dell'elenco delle automobili non sono attive.
- Le voci dell'elenco dei clienti sono attive:
 - Se il cliente ha una macchina in suo possesso, cliccando sulla voce viene chiesta conferma della restituzione. Se è confermato, si chiama `restituisci()`, la macchina viene tolta dal possesso del cliente e reinserita nell'elenco del parco macchine (si aggiorni ovviamente la lista visibile).
 - Se il cliente NON ha una macchina in suo possesso, cliccando sulla voce compare un nuovo modale con un semplice menu di selezione di tutte le macchine

disponibili nel parco macchine. Selezionando una macchina e confermando l'ok al modale, si effettua il noleggio.