TECNOLOGIE WEB

Esame del 05/07/2019 – Compito A  
Tempo a disposizione: 2 ore

**REGOLE.**

Tramite browser, potete accedere solo a:

- localhost

- <http://achecker.ca/checker/index.php> (per validare l’accessibilità del codice HTML)

- <https://validator.w3.org/> (per validare il documento HTML)

I link sopra sono accedibili facendo Ctrl-C Ctrl-V sul link ed aprendolo con Google Chrome.

Nella cartella del compito trovate una copia del W3Schools a cui potete accedere offline usando il browser.  
Potete usare Atom per scrivere il codice.

**IMPORTANTE**:

**lavorate all’interno della cartella della consegna, e consegnate TUTTI i file delle soluzioni (non solo il doc del compito).**

|  |
| --- |
| ESERCIZIO N. 1 **(7 punti)** |

Scrivere un documento HTML valido con codice HTML5 accessibile e semanticamente corretto per realizzare una form che invia i seguenti dati con metodo GET (per Prenotazione Ombrellone):

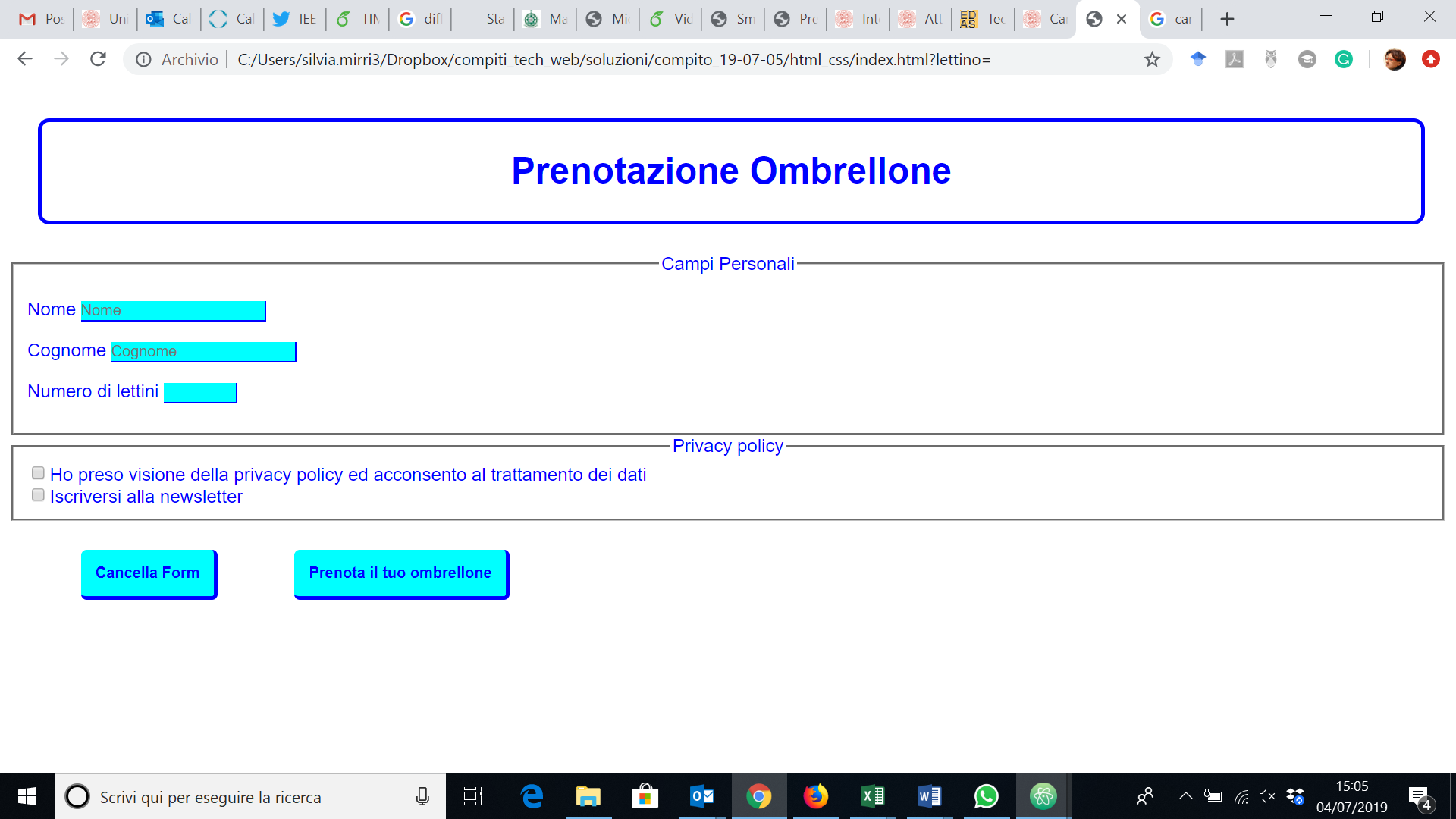
* Gruppo di campi descrittivi personali, tra cui: Nome, Cognome, numero di lettino (con valore da 1 a 100) e corredati dal titolo “Campi Personali”.
* Gruppo di campi inerenti la Privacy Policy: un checkbox accompagnato dalla stringa “Ho preso visione della privacy policy ed acconsento al trattamento dei dati” e un checkbox con la stringa “Inscriversi alla newsletter”. Questo gruppo di campi sarà corredato dal titolo “Privacy policy”.

Devono essere presenti due bottoni: uno con scritto “Cancella Form” e uno con scritto “Prenota il tuo Ombrellone”.

La form è preceduta da un titolo: Prenotazione Ombrellone.

|  |
| --- |
| ESERCIZIO N. 2 **(6 punti)** |

**CSS**



Dato il file html dell’esercizio 1, realizzare il file .css (esterno) così da ottenere il layout e lo stile riportati nello screenshot qui sopra (browser: Chrome), tenendo in considerazione quanto segue:

* Tutti i font devono avere lo stesso font-family, che deve essere Arial. La dimensione deve essere del 100%.
* Il testo del titolo è di colore blue, allineato al centro, e il suo box ha un bordo dello stesso colore, spessore 2px, con un radius di 15px; il occupa una larghezza pari al 90% della pagina. Margini e padding devono essere simili a quelli riportati nello screenshot.
* I testi dei legend devono essere in colore blu e allineati al centro.
* I testi delle label sono di colore blu, allineati a sinistra.
* Gli elementi di tipo input hanno solo il bordo inferiore e il bordo destro (colore blu, di tipo solid, spessore di 2px); il loro colore di sfondo è cyan.
* I bottoni hanno solo il bordo inferiore e il bordo destro (colore blu, di tipo solid, spessore di 2px, con un radius di 5px); il loro colore di sfondo è cyan. Il testo è in grassetto. Margini e padding dei bottini devono essere simili a quelli riportati nello screenshot.

L’uso delle media-query non è richiesto.

|  |
| --- |
| ESERCIZIO N. 3 **(5 punti)** |

Descrivere brevemente le principali novità introdotte da CSS3:

Scrivere qui la risposta:

|  |
| --- |
| ESERCIZIO N. 4 **(7 punti)** |

**JQuery**

Dato il file html 'index.html' in allegato, creare il file ‘soluzione.js’ JavaScript o jQuery in modo tale che:

* Al click sul bottone “Aggiungi” si dovrà:
  + Controllare che il contenuto dell’input nome non sia vuoto. Se vuoto dovrà essere visualizzato un messaggio di errore all’interno del paragrafo (che dovrà poi scomparire alla volta successiva se non vuoto).
  + Aggiungere il testo contenuto nell’input all’interno del tag ol. Il testo dovrà essere preceduto da un bottone “Up” e da un bottone “Down”.
  + Alla fine il campo input dovrà essere svuotato.
* Al click sul bottone “Up” si dovrà spostare l’elemento prima del precedente.
* Al click sul bottone “Down” si dovrà spostare l’elemento dopo il successivo.
* **NB: NON SONO AMMESSE MODIFICHE AL FILE HTML**

|  |
| --- |
| ESERCIZIO N. 5 **(7 punti)** |

**PHP**

Scrivere il codice PHP valido (ovvero che esegua correttamente su server web Apache) che legga i dati che gli sono stati inviati tramite POST nella variabile “soglia1” e “soglia2”.

In questa pagina, occorrerà:

Controllare che sia la variabile "soglia1" che la variabile “soglia 2” non siano nulle e che siano valide, ovvero che siano numeri positivi.

* Salvare su array tutti i numeri maggiori della variabile “soglia1” e minori della variabile “soglia2”.
* Ordinare i numeri in ordine crescente tramite l’algoritmo Bubble Sort (NB: non è possibile usare le funzioni predefinite per l’ordinamento).
* Restituire i numeri ordinati in un array in JSON.
* Dovete supporre che il db esista (nome database: luglio; nome tabella: numeri; username: root, pw: ) e che la tabella "numeri" sia strutturata e riempita secondo le istruzioni che trovate nel file "README\_DB.txt".
* **NB:** anche in caso di errori e input non valido deve essere restituito un JSON.
* **Wikipedia:** Il Bubble sort è un semplice algoritmo di ordinamento di una lista di dati. Ogni coppia di elementi adiacenti viene comparata e invertita di posizione se sono nell'ordine sbagliato. L'algoritmo continua nuovamente ad eseguire questi passaggi per tutta la lista finché non vengono più eseguiti scambi, situazione che indica che la lista è ordinata. E’ presente un’immagine animata esplicativa nella cartella.