TECNOLOGIE WEB

Esame del 21/02/2018 – Turno 1 A  
Tempo a disposizione: 2 ore

**REGOLE:**

Tramite browser, potete accedere solo a:

<http://www.w3schools.com/> (manuali e tutorial)

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-click sul link oppure usando il browser (si consiglia l’utilizzo di Google Chrome).

Potete usare Atom per scrivere il codice.

**IMPORTANTE**:

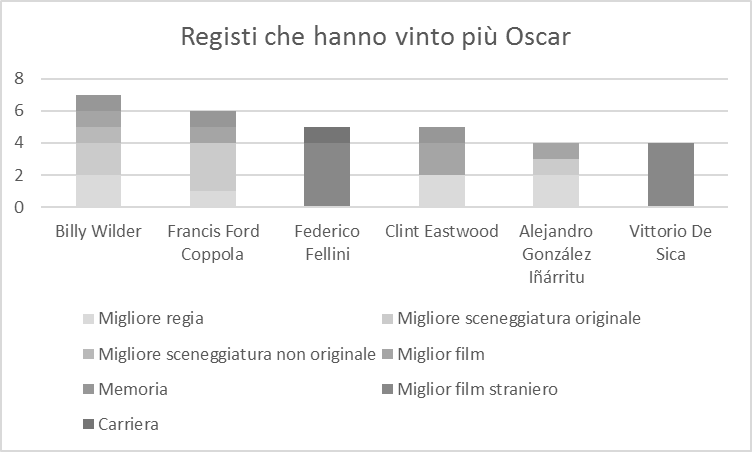
Lavorate all’interno della cartella della consegna, e consegnate TUTTI i file delle soluzioni (non solo il file della soluzione).

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

**HTML**

Scrivere il codice HTML5 accessibile e semanticamente corretto per realizzare un documento che contenga una sezione “Premio Oscar”, che contenga tre sottosezioni:

* una, dal titolo “Caratteristiche”, senza alcun contenuto.
* una, dal titolo “Categorie” che a sua volta sia divisa in due sottosezioni, intitolate rispettivamente “Premi Speciali” e “Altri Premi”; le due sottosezioni non hanno contenuto.
* una, intitolata “Statistiche”, che contenga la seguente immagine, con caption “Registi che hanno vinto più Oscar”.

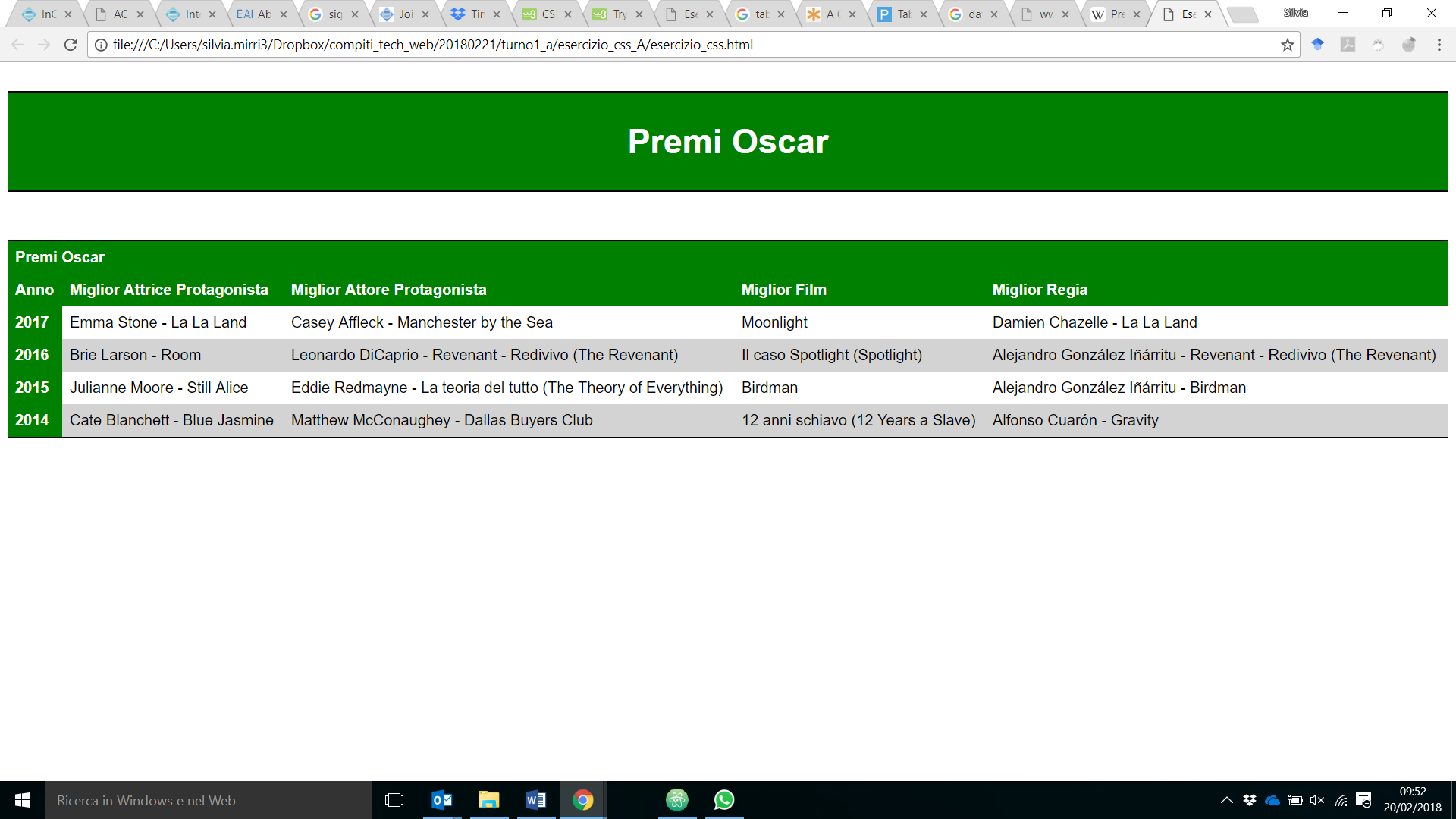


Il documento deve essere HTML5 valido e deve essere accessibile secondo le WCAG2.0 a livello A (la validazione con tool automatici dell’accessibilità non è di per sé sufficiente).

L’immagine presente nella prima riga della tabella è disponibile al file **registi.png**.

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

**CSS**



Dato il file esercizio\_css.html, 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%.
* Lo sfondo di <header> è di colore green (con bordi superiore e inferiore black, di 2 px).
* Il testo in <header> è di colore bianco.
* La larghezza di <header> e della tabella deve essere pari al 100%.
* I colori dei bordi e degli sfondi degli elementi della tabella (intestazione, corpo e footer della tabella) devono essere uguali a quelli riportati nello screenshot, usando i colori: green, black e lightgrey. In particolare, lo sfondo delle righe pari deve essere lightgrey, mentre quello delle righe dispari deve essere bianco.
* Il bordo superiore e inferiore della tabella deve avere uno spessore pari a 2px. Le righe e le celle della tabella non devono avere bordo.
* Il testo nelle celle di intestazione deve essere bianco su sfondo verde.

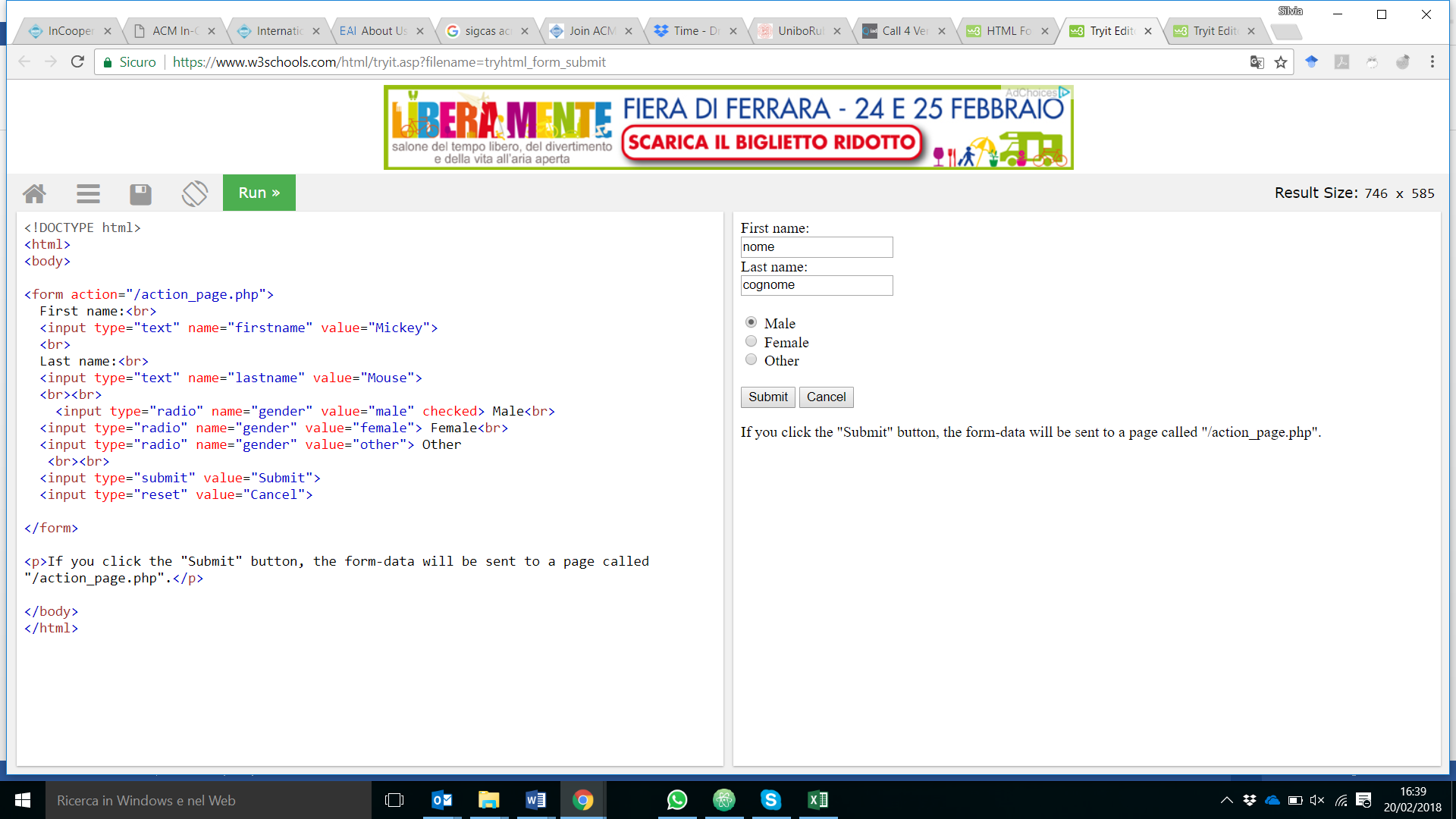
Il posizionamento degli elementi di blocco può essere basato su una soluzione float oppure flexbox. L’uso delle media-query non è richiesto.

Non sono ammesse modifiche al file .html.

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

**Domanda**

Data la form presente nello screenshot qui di seguito, presentare i principali problemi di accessibilità che potrebbero essere presenti e le strategie per creare un codice accessibile.



Scrivere qui la risposta:

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

**JQuery**

Dato il file html **'esercizio\_javascript.html'** in allegato, creare il codice JavaScript o jQuery in modo tale che:

* Al caricamento della pagina siano nascosti tutti gli span.
* Venga controllata la soluzione del sudoku proposta. Se è valida deve essere visualizzato lo span con classe win, altrimenti quello con id lose. Per controllare la validità è necessario controllare che:
  + Ogni riga contenga tutti i numeri da 1 a 9.
  + Ogni colonna contenga tutti i numeri da 1 a 9.
  + Ogni regione contenga tutti i numeri da 1 a 9.

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

**PHP**

Scrivere il codice PHP valido (ovvero che esegua correttamente su server web Apache) per completare correttamente una partita di sudoku. Lo script dovrà quindi:

* Il database è popolato da righe contenenti numeri da 0 a 9 intervallati da virgole, il numero 0 significa che cella è vuota.
* Creare una funzione inserisciCella() che all’atto del submit, nel caso il radio button sia impostato su “Inserisci”:
  + Inserisca il numero preso dall’input type con value=”numero” nella cella corrispondente al valore preso dall’input type con value=”cella” controllando che la cella sia effettivamente vuota.
* Creare una funzione mossePossibili() che all’atto del submit, nel caso il radio button sia impostato su “mosse possibili”:
  + Controlli che la cella corrispondente al valore preso dall’input type con value=”cella” sia vuota.
  + Stampi in output tutti i valori ammissibili per quella cella in accordo con le regole del gioco (Scopo del gioco è quello di riempire le caselle bianche con numeri da 1 a 9, in modo tale che in ogni riga, in ogni colonna e in ogni regione quadrata, siano presenti tutte le cifre da 1 a 9 senza ripetizioni).
* Dovete supporre che il db esista (nome database: sudoku; nome tabella: riga; username: “root”, pw: “”) e che la tabella "riga" sia strutturata e riempita secondo le istruzioni che trovate nel file "README\_DB.txt".