

ISTITUTO TECNICO INDUSTRIALE STATALE "G. VALLAURI"

Via S. Michele, 68 – 12045 Fossano

Indirizzo **INFORMATICA**

Classe **III INF B**

14 marzo 2024

Verifica di Tecnologie

Esercizio 1 (2,5 punti)

Si vuole simulare un test online che all'avvio visualizza all'utente un elenco di domande relative a javascript.

Quiz JavaScript

- Qual è il linguaggio di scripting lato client più comune?
☐ a) Java
☐ b) Python
☐ c) Ptp
☐ d) JavaScript
- Cosa significa HTML?
☐ a) HyperTransfer Markup Language
☐ b) HyperText Markup Language
☐ c) HyperText Modelling Language
- Quale tra le seguenti dichiarazioni di variabili è corretta in JavaScript?
☐ a) let x = 10;
☐ b) int x = 10;
☐ c) variable x = 10;
- Che cosa contiene in JavaScript una variabile non inizializzata?
☐ a) null;
☐ b) ""
☐ c) undefined;
☐ d) NaN;
- Come si scrive una funzione in javascript ?
☐ a) function somma (a, b){};
☐ b) function somma (int a, int b){};
☐ c) function somma (let a, let b){};

Verifica Risposte

Le risposte relative ad ogni domanda sono mutuamente esclusive e nel vettore **risposteCorrette** è memorizzato il value della risposta corretta.

```
const risposteCorrette = ['d', 'b', 'a', 'c', 'a'];
```

Poiché il codice javascript contiene una unica procedura **calcolaPunteggio()**, non è necessario l'utilizzo dell'evento **onload**, ed i vari puntatori possono essere creati ed inizializzati direttamente all'interno di questa procedura.

Dopo che l'utente ha risposto a tutte le domande, in corrispondenza del click sul pulsante **VerificaRisposte** l'applicazione deve:

- Controllare che l'utente abbia risposto a tutte le 5 domande. Se manca qualche risposta visualizzare un semplice messaggio del tipo "Devi rispondere a tutte le domande".
- Se l'utente ha risposto a tutte le domande l'applicazione deve calcolare il voto sommando 2 punti per ogni risposta esatta
- In corrispondenza di eventuali risposte sbagliate occorre deselezionare il radio button corrispondente
- Al termine visualizzare il voto in neretto nel tag DIV finale (inizialmente nascosto tramite apposita impostazione CSS) e disabilitare il pulsante **VerificaRisposte**

Quiz JavaScript

- Qual è il linguaggio di scripting lato client più comune?
☐ a) Java
☐ b) Python
☐ c) Ptp
☒ d) JavaScript
- Cosa significa HTML?
☐ a) HyperTransfer Markup Language
☐ b) HyperText Markup Language
☐ c) HyperText Modelling Language
- Quale tra le seguenti dichiarazioni di variabili è corretta in JavaScript?
☐ a) let x = 10;
☐ b) int x = 10;
☐ c) variable x = 10;
- Che cosa contiene in JavaScript una variabile non inizializzata?
☐ a) null;
☐ b) ""
☒ c) undefined;
☐ d) NaN;
- Come si scrive una funzione in javascript ?
☒ a) function somma (a, b){};
☐ b) function somma (int a, int b){};
☐ c) function somma (let a, let b){};

Verifica Risposte

il tuo voto è 6

Esercizio 2 (2,5 punti)

Ordina le lettere in ordine alfabetico

N	C	K	Q	S	O	P	
<input type="button" value="sposta"/>	<input type="button" value="sposta"/>	<input type="button" value="sposta"/>	<input type="button" value="sposta"/>	<input type="button" value="sposta"/>	<input type="button" value="sposta"/>	<input type="button" value="sposta"/>	<input type="button" value="sposta"/>

All'avvio l'applicazione deve generare **sette** numeri **diversi tra loro** relativi ai codici ASCII delle lettere maiuscole (A=65, Z=90). Per ogni numero generato copiare il carattere ascii corrispondente all'interno dei relativi Text Box. L'ultimo Text Box rimane vuoto ed il relativo pulsante viene disabilitato.

Il metodo statico **String.fromCharCode(int codice)** restituisce il carattere avente il codice ASCII indicato.

In corrispondenza del click su uno dei pulsanti **sposta** (ad es quello evidenziato in rosso) il sistema deve :

- spostare la lettera corrispondente all'interno della **cella libera** (inizialmente l'ultima)
- abilitare il pulsante della ex cella libera e disabilitare il pulsante appena cliccato.
- In questo modo la cella cliccata diventa la nuova cella libera

Se le prime 7 celle sono tutte occupate il pulsante Verifica viene abilitato, altrimenti viene disabilitato.

Ordina le lettere in ordine alfabetico

N		K	Q	S	O	P	C
<input type="button" value="sposta"/>	<input type="button" value="sposta"/>	<input type="button" value="sposta"/>	<input type="button" value="sposta"/>	<input type="button" value="sposta"/>	<input type="button" value="sposta"/>	<input type="button" value="sposta"/>	<input type="button" value="sposta"/>

Il pulsante **Verifica** controlla se le prime 7 lettere sono correttamente posizionate **IN ORDINE ALFABETICO CRESCENTE**, nel qual caso visualizza un messaggio del tipo "Bravo, hai indovinato !" e disabilita il pulsante stesso, altrimenti visualizza un messaggio del tipo "Ordinamento non corretto, Riprova !".

Esercizio 3(4 punti)

Nella cartella **IMG** sono disponibili 9 immagini di strade, le **prime 4** contengono un semaforo, le **successive 5** senza semaforo.

All'avvio l'applicazione deve visualizzare le 9 immagini consecutivamente come immagine di sfondo dei 9 tag DIV presenti nel file html.

- All'interno dei tag DIV in cui viene inserita una immagine contenente un semaforo (cioè i primi 4), aggiungere un attributo html nascosto **semaforo=true**
- All'interno dei rimanenti tag DIV aggiungere lo stesso attributo ma con valore false: **semaforo=false**



Tramite apposita impostazione CSS i 9 tag DIV hanno tutti un bordo di 3px dello stesso colore dello sfondo (#AFA) in modo che non risulti visibile all'utente

In corrispondenza del **click** su un tag DIV:

- Se il tag DIV non ha il bordo, assegnare (o ricolorare) il bordo di rosso
- Se invece il tag DIV ha il bordo rosso, assegnare (o ricolorare) il bordo #AFA

Suggerimento: assegnare a ciascun tag DIV un attributo nascosto **checked** con il seguente significato:

- **false** significa che l'immagine non ha il bordo (cioè ha il bordo verde)
- **true** significa che l'immagine ha il bordo rosso



In corrispondenza del **click** sul pulsante **VERIFICA** verificare che l'utente abbia effettivamente selezionato TUTTE e SOLO le 4 immagini contenenti il semaforo.

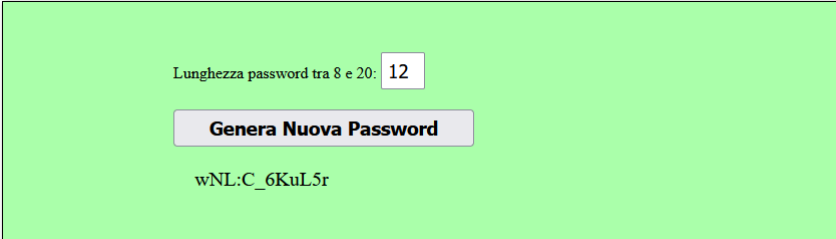
- In caso di errore visualizzare semplicemente un messaggio "Riprova"
- In caso di ok nascondere l'intero **container1** e visualizzare l'intero **container2** (inizialmente nascosto tramite apposita impostazione CSS)

container2

Il **textbox** di inserimento lunghezza password deve accettare soltanto caratteri numerici (ascii 48-57) ed il backspace (ascii 8). Tutti gli altri caratteri NON devono essere lasciati passare. (eventi **onkeypress** oppure **onkeydown**)

Se il contenuto del textbox NON è **vuoto** e contiene un numero compreso tra 8 e 20, il pulsante **Genera Nuova Password** deve essere abilitato, altrimenti deve essere disabilitato (evento **oninput**)

In corrispondenza del **click** sul pulsante **GENERA NUOVA PASSWORD** generare e visualizzare all'interno dell'apposito tag **span** una stringa casuale avente la lunghezza impostata e contenente qualunque combinazione di caratteri ascii compresi tra **47** e **122**, esclusi i caratteri 60 e 62 (cioè < >)



Lunghezza password tra 8 e 20:

Genera Nuova Password

wNL:C_6KuL5r

Facoltativo (da fare a casa)

Creare una nuova cartella "**Esercizio 4**" replicando l'esercizio precedente e fare in modo che, all'avvio, l'applicazione posizioni le 9 immagini in posizione casuale.

Per il resto il funzionamento deve rimanere lo stesso identico dell'esercizio precedente.