|  |
| --- |
| **ù** |
| **Esercizio prova pratica 4/11/2020**   * **Regole**:   + Nominare il file html come COGNOMEMATRICOLA.HTML. Le parti CSS, Javascript e HTML vanno unite in un unico file .html.   + Possiamo consultare il materiale fornito a lezione.   + Possiamo consultare manuali   + Abbiamo novanta minuti di tempo. * Il testo della prova è disponibile poco più avanti. * Ogni prova presenta delle foto di esempio di quanto va realizzato. Chiaramente non verremo penalizzati per una tonalità di colore diversa o per una differenza di px nelle dimensioni di un elemento. * Contrariamente agli esami di Fondamenti di programmazione non dobbiamo capire una “struttura dati” particolare. Prendiamo ad esempio questa prova pratica: gli elementi della slot possono essere realizzati con input, con semplici div, con celle di una tabella. L’importante è realizzare un codice che esegue quanto detto nella traccia. * Il codice fornito da Tesconi non funziona correttamente, quindi mi limiterò a farvi vedere la mia soluzione:   <!DOCTYPE html>  <html>  <head>  <meta charset="utf-8">  <title>Slot machine</title>  <style>  #box {  padding: 4px;  background-color: lightgrey;  text-align:center;  }    h1 {  margin:0;  margin-bottom:10px;  }    .cella {  background-color:white;  border:1px solid black;  padding:15px;  font-size:26px;  }    .colonna {  display: inline-block;  margin:10px;  }  </style>  </head>  <body **onLoad="genera\_slot()"**>  <div id="box">  <h1>Slot Machine</h1>  Attenzione agli eventi  <div id="area\_slot">    </div>  <b>Puntata</b>: <input type="text" autofocus value="0" id="puntata"> € <button id="avvio" **onclick="avvio()"**>Avvia</button> <button disabled id="stop" **onclick="stop()"**>Stop</button>  </div>  <script type="text/javascript">  Col primo array mi salvo i vari riferimenti alle celle della slot machine (i riferimenti ottenuti utilizzando la getElementById): questa cosa ci permetterà di modificare i colori in modo comodo e veloce.  var array\_elementi = new Array();  var array\_operazioni = new Array();  var posizione\_slot = new Array();    function genera\_slot() {  var area\_slot = document.getElementById('area\_slot');  for(i = 0; i < 3; i++) {  html = document.createElement('div');  html.setAttribute('class', 'colonna');  *Inserisco colonne e celle della slot machine nell’HTML.*  *Riempo l’array\_elementi nel corso della creazione.*  html.setAttribute('id', 'c' + i);  array\_elementi[i] = new Array();  for(l = 0; l <= 9; l++) {  array\_elementi[i][l] = document.createElement('div');  array\_elementi[i][l].setAttribute('class', 'cella');  array\_elementi[i][l].setAttribute('id', 'c' + i + '\_r' + l);  array\_elementi[i][l].append(l);    html.appendChild(array\_elementi[i][l]);  }  area\_slot.appendChild(html);  }  }    function avvio() {  var input = document.getElementById('puntata');  if(input.value < 0 || input.value > 100)  return false;  *Inverto lo status dei bottoni AVVIO e STOP oltre a rendere non modificabile l’input con la puntata*    input.disabled = true;  var bottone\_avvio = document.getElementById('avvio');  var bottone\_stop = document.getElementById('stop');  bottone\_avvio.disabled = true;  bottone\_stop.disabled = false;    for(i = 0; i < 3; i++) {  posizione\_slot[i] = Math.floor(Math.random()\*10);  velocita = Math.floor(Math.random()\*5+1)\*10 + 50;  array\_operazioni[i] = setInterval("cambiaSlot("+i+")", velocita);  }  Per ogni colonna imposto il cambiamento periodico della cella colorata di rosso. Per ottenere la velocità calcolo un valore random compreso tra 0 e 5 (5 incluso, includo il 5 sommando 1 e arrotondando per difetto), moltiplico il valore random per 10 e sommo 50 (senza la somma otterrei un numero compreso tra 0 e 50). Ricordarsi che Math.random() non restituirà mai 1.  }    function cambiaSlot(colonna) {  array\_elementi[colonna][posizione\_slot[colonna]].style.backgroundColor = 'white';  posizione\_slot[colonna] = (posizione\_slot[colonna] + 1)%10;  array\_elementi[colonna][posizione\_slot[colonna]].style.backgroundColor = 'red';  }  Utilizzo l’array posizione\_slot per ricordarmi la posizione dell’elemento colorato di rosso. Ogni volta che eseguo la funzione coloro questo elemento di bianco, passo all’elemento successivo (della colonna) e coloro questo di rosso.    function finegioco() {  var input = document.getElementById('puntata');  if(posizione\_slot[0] == posizione\_slot[1] && posizione\_slot[1] == posizione\_slot[2])  messaggio = 'Hai vinto ' + puntata\*10 + ' €!';  else  messaggio = 'Hai perso!';    var finestra = window.open("", "Finestra", "width=300,height=200");  finestra.document.write("<!DOCTYPE HTML><html><head><title>Risultato</title><style>body { text-align: center; }</style></head><body><h1>" + messaggio + "</h1></body></html>");  setTimeout(function() { finestra.close(); }, 5000);    reset();  }  Determino il messaggio da stampare (di vittoria o di sconfitta, apro la finestra e ne imposto la chiusura dopo 5000 millisecondi (5 secondi). Infine chiamo la funzione reset() per riportare la slot macchine allo stato iniziale.  *Inverto lo status dei bottoni AVVIO e STOP oltre a rendere non modificabile l’input con la puntata*  function stop() {  var bottone\_avvio = document.getElementById('avvio');  var bottone\_stop = document.getElementById('stop');  bottone\_avvio. disabled = false;  bottone\_stop.disabled = true;    clearTimeout(array\_operazioni[0]);  setTimeout("clearTimeout(" + array\_operazioni[1] + ")", 1000);  setTimeout("clearTimeout(" + array\_operazioni[2] + ")", 2000);  setTimeout("finegioco()", 3000);  }    Blocco subito l’esecuzione periodica della funzione cambiaSlot() nella prima colonna, la blocco nella seconda dopo 1 secondo e nella terza dopo 2 secondi. Dopo 3 secondi eseguo la funzione finegioco().  function reset() {  var input = document.getElementById('puntata');  input.disabled = false;  input.value = 0;  array\_elementi[0][posizione\_slot[0]].style.backgroundColor = 'white';  array\_elementi[1][posizione\_slot[1]].style.backgroundColor = 'white';  array\_elementi[2][posizione\_slot[2]].style.backgroundColor = 'white';  }  Rendo la puntata nuovamente modificabile e imposto il valore dell’input a 0. Concludo rendendo bianche le celle rimaste rosse nelle tre colonne.    </script>  </body>  </html> |