



C'E' MA NON SI VEDE...

CoderDOJO su Javascript - Trento



Augusto Ciuffoletti

26 maggio 2015

Il topo scompare!

- Esiste una **proprietà** dello stile che dice se un certo **elemento** è visibile oppure no
- La proprietà si chiama `display` e la posso impostare così:

```
document.getElementById("ilTopo").style.display="none";
```

Nascondere il topo

- Oppure così:

```
document.getElementById("ilTopo").style.display="inline";
```

Nascondere il topo

- Se la proprietà `display` è `null` l'elemento non sente il passaggio del mouse e i click!

La porta che si apre e si chiude

- Hai provato a far apparire e scomparire il topo ogni volta che premi?
- Ora prova ad aprire e chiudere una porta!
 - Come fare? Metti insieme un'immagine della porta aperta ed una della porta chiusa, e ne mostri solo una
 - Quando la porta è aperta, quella chiusa è nascosta
 - Quando la porta è chiusa nascondi quella aperta

```
  

```

Le due porte (quella chiusa si vede)

Aprire e chiudere la porta

- Per aprire e chiudere la porta controllo chi ha **ricevuto** il click
 - se lo ha ricevuto la porta chiusa, mostro quella aperta e nascondo quella chiusa
 - ricorda, un elemento con `display a none` non sente il mouse!

```
function porta(event){  
  if ( event.target.getAttribute("id")== "chiusa" ) {  
    document.getElementById("aperta").style.display="inline";  
    document.getElementById("chiusa").style.display="none";  
  } else {  
    document.getElementById("aperta").style.display="none";  
    document.getElementById("chiusa").style.display="inline";  
  }  
}
```

Se è chiusa aprila

A cosa serve quell'!(event)?

- Cambiamo gioco: ora abbiamo una scacchiera con nove pietre preziose
- Creala facendo una tabella 3x3 e inserendo in ogni casella l'immagine di una pietra preziosa
 - ... ricorda, la tabella è tra tag `table`, ogni riga tra `tr`, ogni elemento tra `td`
- Come faccio a scrivere un programma per far scomparire ogni pietra quando la clicco?
- Devo scrivere un programma per ciascuna casella?

AIUTO: NOVE PROGRAMMI UGUALI!

Dimmi quello che sai

- Quando si esegue la **function** collegata ad un evento, è possibile dire al programma di che evento si tratta
- Dentro quell'`event` sono nascoste un mucchio di informazioni, ma a noi ne interessa solo una: `event.target`
 - corrisponde all'**Elemento** che ha ricevuto l'evento
- Per fare che ogni pietra sparisca quando viene cliccata devo scrivere **solo una** funzione

```
function scompare(event){  
    event.target.style.display = "none";  
}
```

Una funzione che nasconde qualsiasi elemento

- La funzione trova nel parametro `event` l'elemento che l'ha chiamata (`target`), e lo nasconde!

CHALLENGE: cerca la regina

- Tra tutte le pietre una soltanto ha l'identificatore regina
 - è la regina di tutte le pietre preziose!
- Quando clicchi sulla regina scompare tutta la scacchiera
- **CHALLENGE**: scrivi questo gioco con **una sola funzione**

Porta automatica (si chiude da sè)

- Facciamo richiudere la porta da sola (dopo tre secondi)
- Dopo aver aperto la porta dobbiamo mettere un timer, per richiuderla
- Vediamo come si fa:

```
function automatica(){
  document.getElementById("aperta").style.display="inline";
  document.getElementById("chiusa").style.display="none";
  var time=setTimeout(function(){
    document.getElementById("aperta").style.display="none";
    document.getElementById("chiusa").style.display="inline"},
  3000);
}
```

La porta automatica

- Il numero 3000 sta per 3 secondi (3000 millisecondi)

Aprire molte porte

- Ora costruiamo una tabella con tre porte

```
<table onclick="apre(event)"><tr>
<td> 
      
</td>
<td> 
      
</td>
<td> 
      
</td></tr></table>
```

Tre porte da aprire

- E' facile, sono tutte uguali!
- Ma come faccio a aprirle (e chiuderle) una per una?

Questa è difficile!

- Il click che apre può arrivare sull' con la porta aperta (se la porta è aperta) o sul quello con la porta chiusa (se è chiusa)
- Comunque vada, io devo sempre rendere visibile quella aperta, e invisibile quella chiusa

Come faccio?

Questa è difficile!

- Il click che apre può arrivare sull' con la porta aperta (se la porta è aperta) o sul quello con la porta chiusa (se è chiusa)
- Comunque vada, io devo sempre rendere visibile quella aperta, e invisibile quella chiusa

Come faccio?

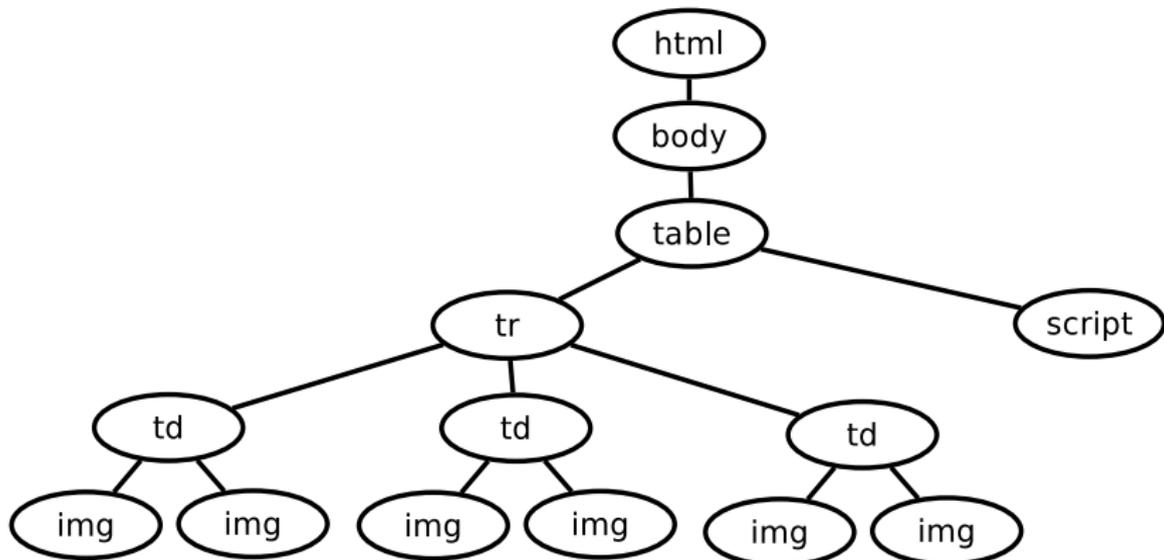
NAVIGANDO!

- Se dall'elemento che ha ricevuto l'evento (il click) mi **sposto** su quello che lo contiene (il <td>)...
- ...tutto diventa più facile: mostro la prima immagine e nascondo la seconda
- Prima di vedere come si fa, capiamo meglio.

Padri e figli: una grande famiglia!

- Una pagina web può essere una genealogia, o tante scatole una dentro l'altra
 - In cima `<html>` è la scatola che contiene tutte le altre
 - Dentro, un'unica scatola `<body>`, l'unico **figlio**
 - Poi noi abbiamo due figli del `<body>`: una `<table>` e lo `<script>`
 - La table ha un unico figlio: un tag `<tr>` (ci siamo quasi)
 - Il tag `<tr>` ha tre figli, tutti `<td>`
 - e ciascuno di questi ha due figlie ``
- Facciamo un disegno del nostro albero genealogico: il DOM

Il nostro DOM



- Il click *colpisce* una ``, non so se la destra o la sinistra
- Salgo di uno, mostro la prima, nascondo la seconda, e la porta si apre o resta aperta

Come fare?

- Il metodo per salire di uno di chiama `parentNode`
- Per ottenere tutti gli figli con un certo *tag*
`getElementsByTagName`
- Quindi se `event.target` è una ``, con

```
var figli = event.target.parentNode.getElementsByTagName("img");
```

Salgo di uno e seleziono i figli ``

- Ottengo tutti i figli e poi `figlio[0]` è il primo, e `figlio[1]` è il secondo

CHALLENGE

- Ti bastano questi elementi per scrivere la tua soluzione?
- Con un click singolo apri ciascuna delle porte
- Con un click doppio la chiudi
- Poi puoi provare a fare che quando il mouse entra la porta si apre, e quando esce la porta si chiude!
 - ...e magari non subito, ma dopo un po'...

```
function apre(event){
    var figli = event.target.parentNode.getElementsByTagName("img");
    figli[0].style.display="inline";
    figli[1].style.display="none";
}
function chiude(event){
    var figli = event.target.parentNode.getElementsByTagName("img");
    figli[0].style.display="none";
    figli[1].style.display="inline";
}
```

La porta automatica

Challenge Finale: Memory

- Quasi un gioco di memoria
- Bisogna trovare lo smeraldo, ma la casella si nasconde dopo tre secondi che l'hai scoperta
- Quando trovi lo smeraldo, la tabella scompare
- Per risolvere il challenge devi mettere insieme tutto quello che sai!
 - quando clicchi la cella con l'id regina devi nascondere la tabella
 - quando clicchi le altre caselle:
 - prima nascondi il bottone e mostri la pietra
 - dopo tre secondi nascondi la pietra e mostri il bottone

PROVACI

```
function scopre(event){
  var casella = event.target.parentNode;
  if ( casella.id == "regina" ) {
    document.getElementById("tabella").style.display="none";
  } else {
    var figli = casella.getElementsByTagName("img");
    figli[0].style.display="inline";
    figli[1].style.display="none";
    var time=setTimeout(
      function(){
        figli[0].style.display="none";
        figli[1].style.display="inline"
      }, 3000);
  }
}
```

- Ricorda che una casella deve avere id=regina