




JavaScript

Events & the DOM



The Travelling Hobbit: a challenge

```
function showMordor() {
  document.querySelector("#message").innerText = "I am in Mordor!";
}
function showRohan() {
  document.querySelector("#message").innerText = "I am in Rohan!";
}
function hideMessage() {
  document.querySelector("#message").innerText = "";
}
var mordor = document.querySelector("#mordor");
var rohan = document.querySelector("#rohan");

mordor.addEventListener("mouseover", showMordor, false);
mordor.addEventListener("mouseout", hideMessage, false);

rohan .addEventListener("mouseover", showRohan, false);
rohan .addEventListener("mouseout", hideMessage , false);
```

The challenge:

- Do we really need to find every single location and add the 2 event handlers to each one, one step at a time?
- What if we have 10 locations? 100?

A better solution

- Use `document.querySelectorAll` to get all elements that need to listen to events
- Use a loop to iterate over all selected elements and add the same event handlers: `showMessage` and `hideMessage` (*let's try it now!*)
- ***Will this cause a problem?***
- Yes: our `showMessage` function should know what message to show!

```
function showMordor() {
    document.querySelector("#message").innerText = I am in Mordor!;
}
function showRohan() {
    document.querySelector("#message").innerText = I am in Rohan!;
}
```

A better solution: solved?

- When our event handler function is called, an event object is automagically (implicitly) passed to that function:
 - ```
function handleEvent(e) (
 //e is the event object!
)
```
- The event object is different for different events:
  - For example, a mouse event will know the location of the mouse coordinates
  - All event objects know what triggered the event - i.e., the target object
  - `e.target` (the target property of the event object) gives us access to that object!
- If we clicked a div and that div had an event handler attached, then we can access that div inside its event handler function!

# A better solution: solved, almost...

```
function showMessage(e) {
 var div = e.target;
 console.log("This element triggered the event: " +div);
 document.querySelector("#message").innerText = Still, where am I????;
}
function hideMessage() {
 document.querySelector("#message").innerText = "";
}
var mordor = document.querySelector("#mordor");
mordor.addEventListener("mouseover", showMessage, false);
mordor.addEventListener("mouseout", hideMessage, false);
```

# A better solution: add message text to the HTML

```
<div id="places">
 <div id="greyhavens">
 <p>Welcome to Grey Havens, the gateway to the West!
 </div>
 <div id="rivendell">
 <p>Welcome to Rivendell, the House of Lord Elrond!
 </div>
 <div id="rohan">
 <p>Welcome to Rohan, the Land of the horse lords!
 </div>
 <div id="mordor">
 <p>Abandon all hope, traveller - you are in Mordor!
 </div>
</div>
```

# How do we access that text?

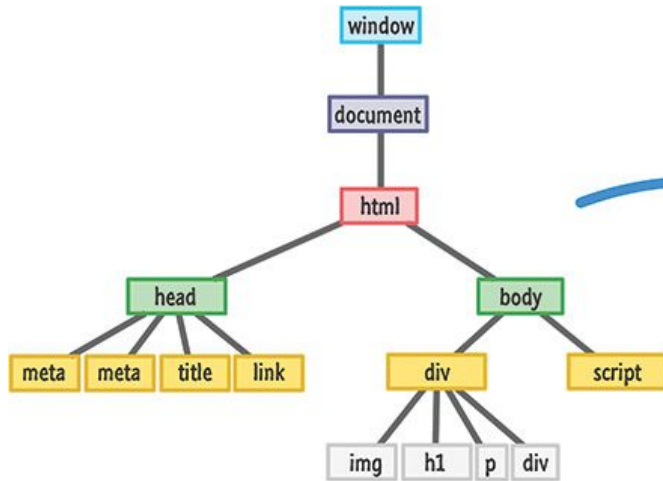
We need to get to the paragraph INSIDE the division that triggered the event!

```
<div id="mordor">
 <p>Abandon all hope, traveller - you are in Mordor!
</div>
```

How do we do that?

Recall chapter 25: Traversing the DOM!

# Traversing the DOM



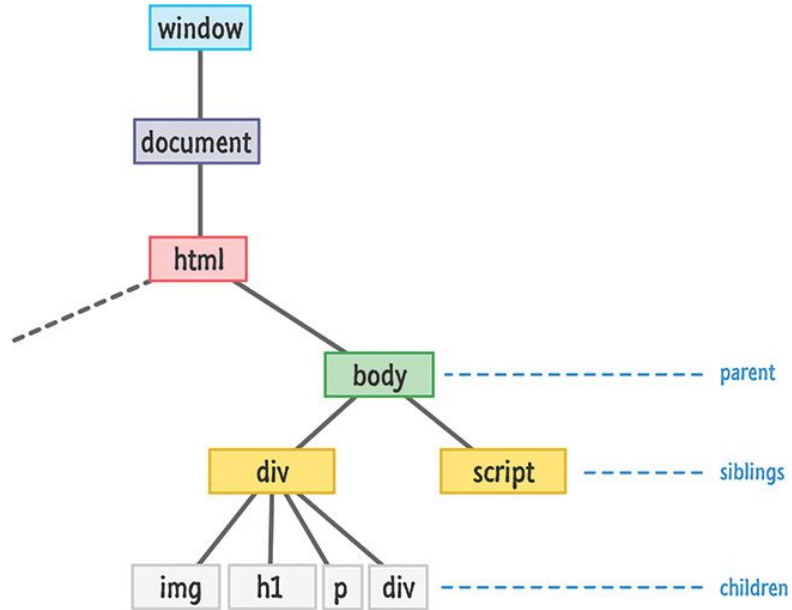
The DOM



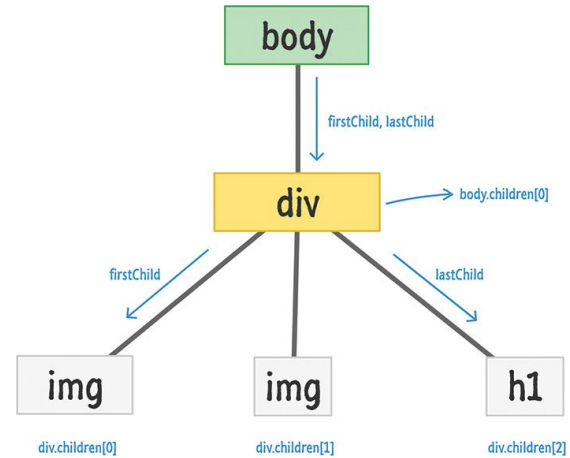
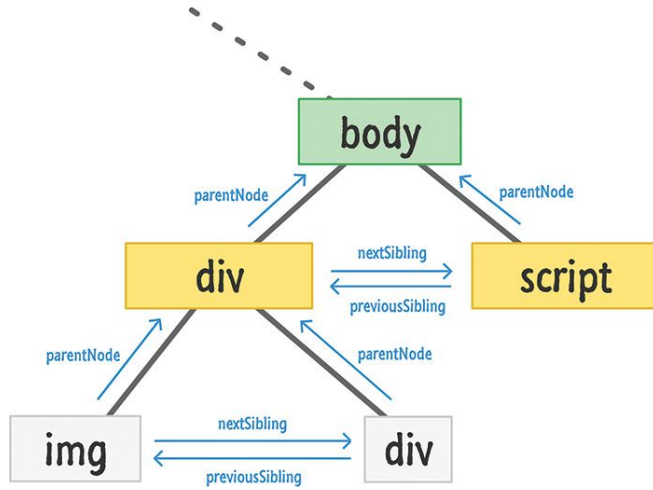
The Browser  
(aka what you see)



# Traversing the DOM



# Traversing the DOM



# Solution, finally!

## HTML:

```
<div id="mordor">
 <p>Abandon all hope, traveller - you are in Mordor!
</div>
```

## JavaScript:

```
function showMessage(e) {
 var div = e.target;
 var p = div.children[0]
 var text = p.innerText;
 document.querySelector("#message").innerText =text;
}
```