



DOM

Document Object Model

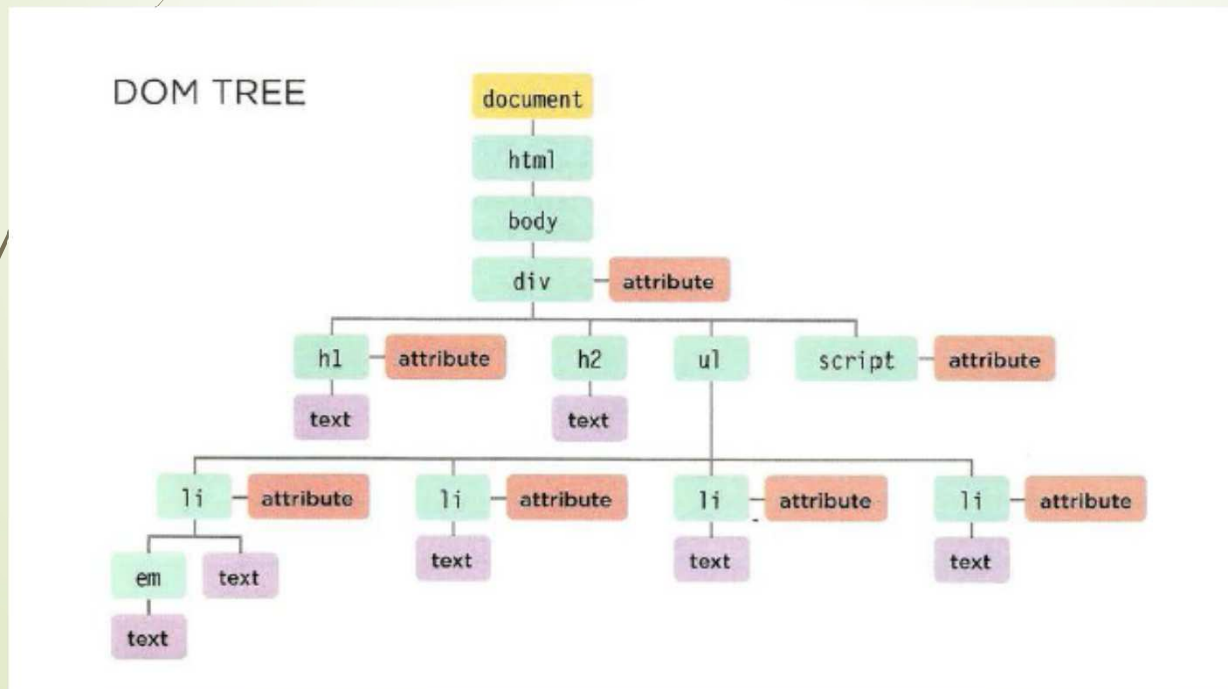


DOM

- Kako da browser kreira model HTML stranice
- Kako Javascript da pristupi i promeni sadržaj stranice
- DOM nije ni deo HTML-a, ni deo Javascript-a
- Zaseban set pravila
- Sastoji se od objekata
- Svaki objekat predstavlja deo učitane stranice
- DOM definiše i metode i osobine (*property*) koje se koriste za ažuriranje objekata

DOM tree

- Na vrhu je document objekat koji predstavlja stranicu, a njegova deca predstavljaju elemente na stranici





Document objekat

► **Property**

document.title

document.lastModified

document.URL

document.domain

► **Method**

document.write()

document.getElementById()

document.querySelectorAll()

document.createElement()

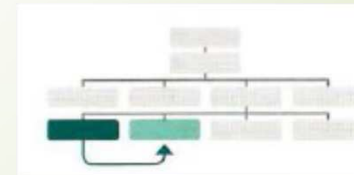
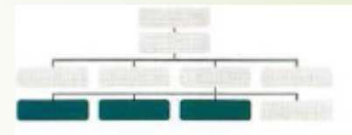
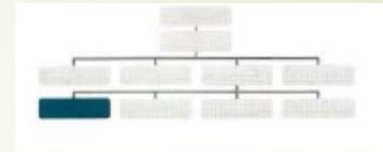
document.createTextNode()

Pristup elementima - metode

- getElementById()
- querySelector ()

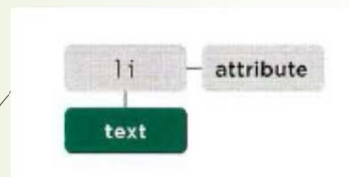
- getElementsByClassName()
- querySelectorAll()
- getElementsByTagName()

- parentNode
- previousSibling/nextSibling
- firstChild/lastChild

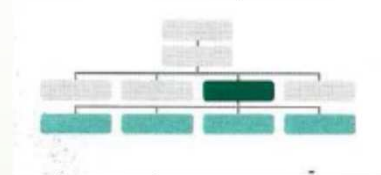


ažuriranje

- nodeValue – property – omogućava pristup sadržaju čvora u stablu sa tekstualnim sadržajem



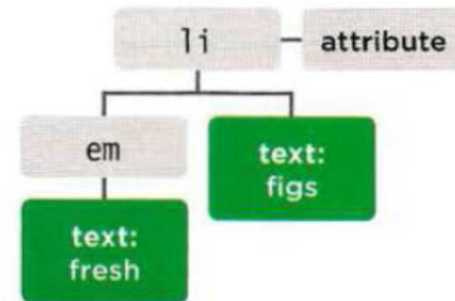
- innerHTML – property – omogućava pristup deci i tekstualnom sadržaju



- textContent – ažuriranje i skupljanje teksta čvora i njegove dece

ažuriranje

➤ `<li id="one">freshfigs`



➤ figs

```
document.getElementById('one').firstChild.nextSibling.nodeValue;
```

①-----②-----③-----④

➤ fresh figs

```
document.getElementById('one').textContent;
```

➤ `freshfigs`

➤ `document.getElementById('one').innerHTML`



DOM manipulation

- `createElement()`
 - kreira element kao promenljivu
- `createTextNode()`
 - kreira tekstualni čvor kao promenljivu
- `appendChild()`
 - Specidicira kom elementu da se doda kreirani čvor
 - kreirani čvor dodaje u DOM stablo



DOM manipulation

```
var btn = document.createElement("BUTTON");  
var t = document.createTextNode("CLICK ME");  
btn.appendChild(t);  
document.body.appendChild(btn);
```



DOM manipulation

- `hasAttribute()`
- `getAttribute()`
- `setAttribute()`
- `removeAttribute()`



Document.write()

- document.write()
- Jednostavan način da se doda na stranici sadržaj koji nije bio u originalnom kodu stranice
- Nije preporučljivo za korišćenje
- Jednostavno kao primer početnicima kako mogu da dodaju sadržaj na stranicu
- Mana je da radi samo kada se stranica u potpunosti učitava
- Može da se dogodi da ne doda sadržaj ili da ga prepíše preko cele stranice
- Retko se koristi



Events

UI EVENTS Occur when a user interacts with the browser's user interface (UI) rather than the web page

EVENT	DESCRIPTION
load	Web page has finished loading
unload	Web page is unloading (usually because a new page was requested)
error	Browser encounters a JavaScript error or an asset doesn't exist
resize	Browser window has been resized
scroll	User has scrolled up or down the page

KEYBOARD EVENTS Occur when a user interacts with the keyboard (see also input event)

EVENT	DESCRIPTION
keydown	User first presses a key (repeats while key is depressed)
keyup	User releases a key
keypress	Character is being inserted (repeats while key is depressed)

MOUSE EVENTS Occur when a user interacts with a mouse, trackpad, or touchscreen

EVENT	DESCRIPTION
click	User presses and releases a button over the same element
dblclick	User presses and releases a button twice over the same element
mousedown	User presses a mouse button while over an element
mouseup	User releases a mouse button while over an element
mousemove	User moves the mouse (not on a touchscreen)
mouseover	User moves the mouse over an element (not on a touchscreen)
mouseout	User moves the mouse off an element (not on a touchscreen)



Events

- ▶ Kako pokrenuti javascript kod iz događaja
 1. Selektujemo element (čvor) za koji želimo da se pokrene odgovarajući kod
 2. Odredimo na koji događaj želimo reakciju
 3. Dodamo kod koji opisuje šta želimo da odradimo nakon događaja

Events

- Kako pokrenuti javascript kod iz događaja
- Event handlers – koja funkcija se poziva kada je događaj pokrenut

```
element.onevent = functionName;
```

ELEMENT EVENT CODE

A reference to the DOM element node is often stored in a variable.

```
function checkUsername() {  
    // code to check the length of username  
}  
var e1 = document.getElementById('username');  
e1.onblur = checkUsername;
```

The event name is preceded by the word "on."

The code starts by defining the named function.

The function is called by the event handler on the last line, but the parentheses are omitted.

Events

Event listeners

- Noviji način obrade događaja,
- mogu da obave više od jedne funkcije,
- Starije verzije browser-a ih ne podržavaju (\geq IE8)

```
element.addEventListener('event', functionName [, Boolean]);
```

ELEMENT EVENT CODE EVENT FLOW

A reference to the DOM element node is often stored in a variable.

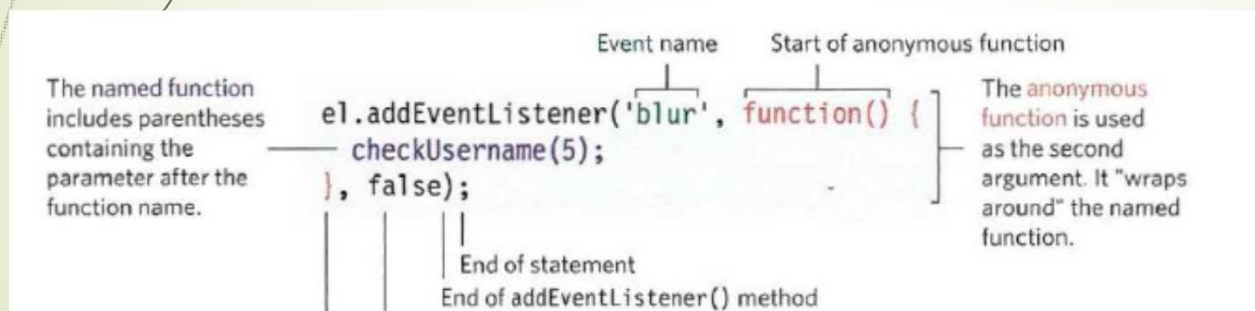
```
function checkUsername() {  
    // code to check the length of username  
}  
var e1 = document.getElementById('username');  
e1.addEventListener('blur', checkUsername, false);
```

The event name is enclosed in quotation marks.

The code starts by defining the named function. The function is called by the event listener on the last line, but the parentheses are omitted.

Events

- Šta ako je potrebno da se proslede argumenti ?
- Annonymus function



Events

- Šta ako je IE verzija 8 i manje
- Umesto `addEventListener()` koristi se `attachEvent()`

If the browser supports
`addEventListener()`:

Run the code inside
these curly braces

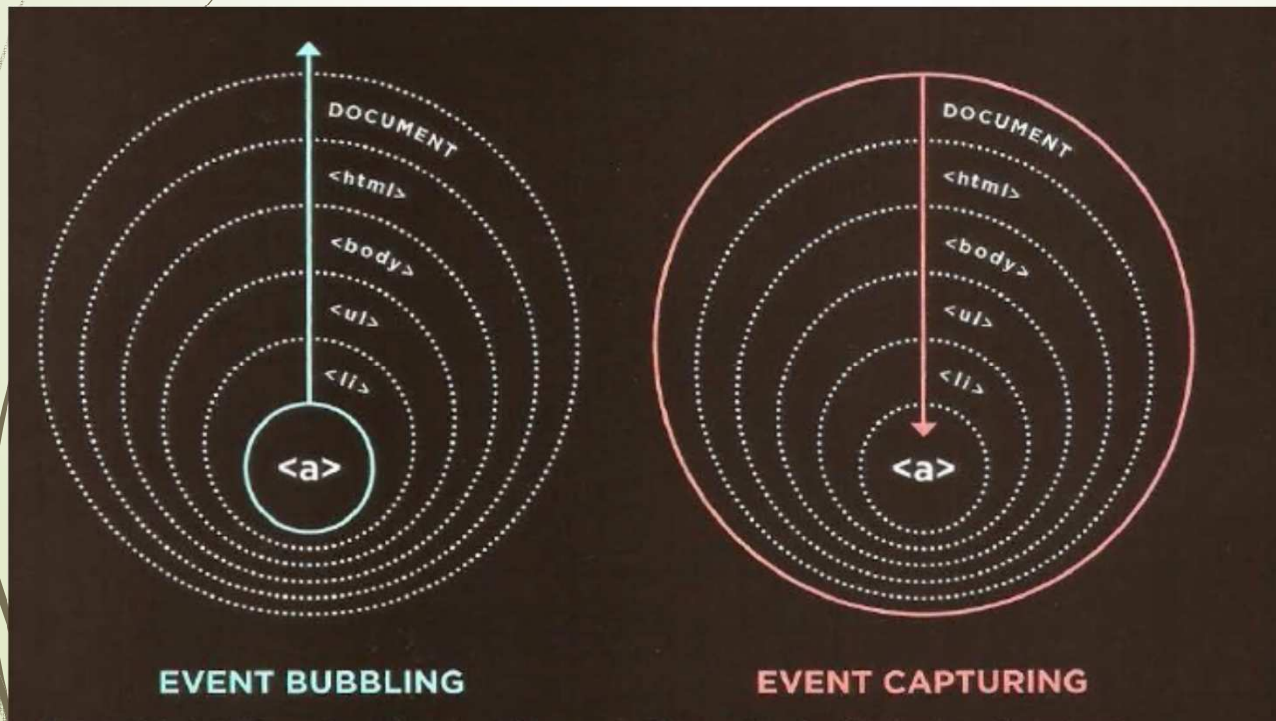
If it doesn't, do
something else:

Run the code inside
these curly braces

```
if (e1.addEventListener) {  
    e1.addEventListener('blur', function() {  
        checkUsername(5);  
    }, false );  
} else {  
    e1.attachEvent('onblur', function() {  
        checkUsername(5);  
    });  
}
```

Events

- Event Flow (False – bubbling, true – capturing)
- default je false





Events

- Kreiranje listener-a za veliki broj događaja može dosta da uspori stranicu i iskoristi veliki deo resursa
- Vrš se delegacija event listener-a na roditelja