

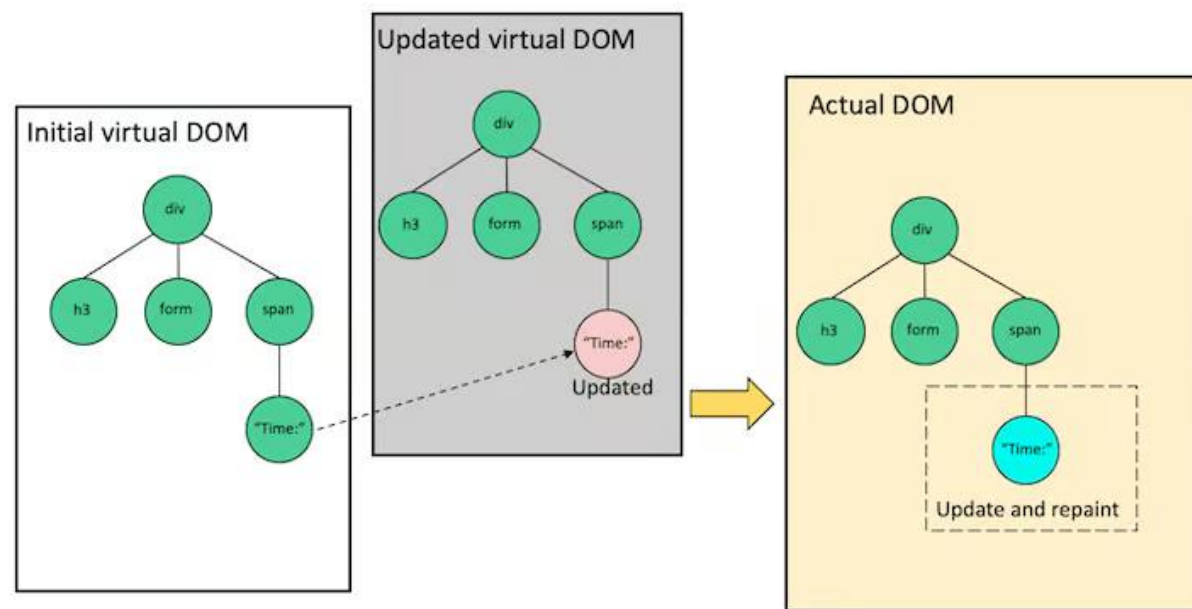
Spletne tehnologije

VAJA 1

Niko Lukač

VAJA 1

V okviru dane vaje izdelajte JavaScript (JS) ogrodje za vodenje navideznega oz. virtualnega DOM (Virtual Document Object Model, VDOM).



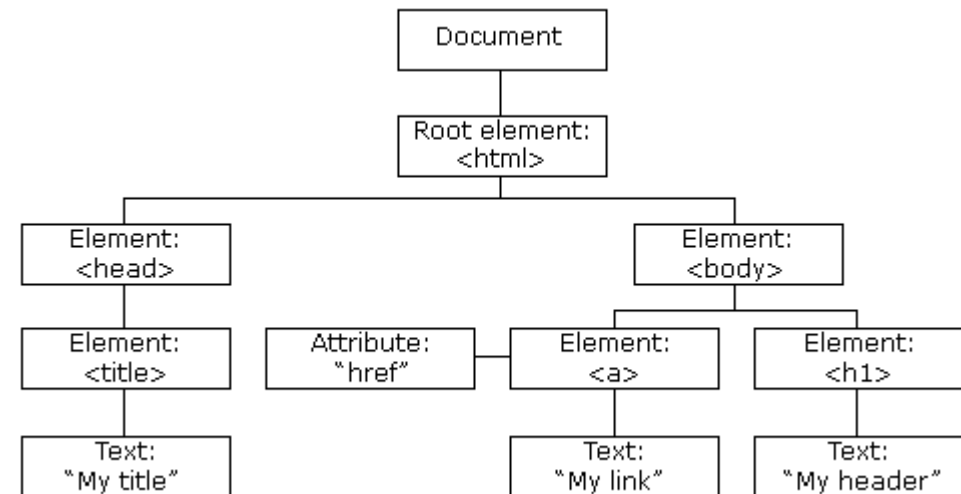
Vir slike: <https://blog.logrocket.com/virtual-dom-react/>

Zahteve

1. Ustvarjanje novih virtualnih elementov HTML, kjer so možni tudi atributi in vsebina elementov. Pri tem si hranite virtualni element kot JS objekt. Ustvarjanje virtualnih pod-elementov HTML v virtualnih elementih HTML. Pri tem uporabite vsebovanje JS objektov. (2%)

- Uporabite JS objekte za

- Elemente
- Tekst
- Attribute (po želji)



- Možno ustvarjanje v literalni obliki ali s konstruktorjem. DOM API je v tej fazi nepomemben.

Primer VDOM elementa v obliki JS objekta:

V literalni obliki:

```
{  
  type: 'ol',  
  props: { className: 'seznam' },  
  children: [  
    { type: 'li', props: {}, children: ['1'] },  
    { type: 'li', props: {}, children: ['2'] },  
    { type: 'li', props: {}, children: ['3'] }  
  ]  
}
```

- Bodite pozorni na rezervirane besede v JS !


Zahteve

2. Izris (angl. rendering) virtualnih elementov. Vhod v izris naj bo korenski prazni element (npr. `<div id="root"></div>`). Pri tem si pomagajte z DOM API. (2%)

- Izrisujete rekurzivno (DFS ali BFS), DOM API:
 - `document.createElement(...)`
 - `document.createTextNode(...)`
 - `element.setAttribute(...)`
 - `element.innerHTML = ...` ali `element.textContent = ...`
 - `element.appendChild(...)`
- Lahko si hranimo reference na JS DOM elemente (ti ki so izrisani z DOM API)
 - Npr. `this._el=document.createElement(...)`
- DOM API: https://developer.mozilla.org/enUS/docs/Web/API/Document_Object_Model

Preslikava (izris) JS objekta v HTML

```
{  
  type: 'ol',  
  props: { className: 'seznam' },  
  children: [  
    { type: 'li', props: {}, children: ['1'] },  
    { type: 'li', props: {}, children: ['2'] },  
    { type: 'li', props: {}, children: ['3'] }  
  ]  
}
```



```
<ol class="seznam">  
  <li>1</li>  
  
  <li>2</li>  
  
  <li>3</li>  
</ol>
```

Zahteve

3. Primerjava dveh VDOM-ov (angl. diffing), pri čemer posodobimo prvi VDOM glede na spremembe v drugem VDOM. Pri tem omogočimo tudi izris za samo spremenjenje elemente in attribute. (3%)

- Uporabite rekurzijo, preverjate:
 - attribute elementa
 - vsebina elementa
 - spremembe v pod-elementih (brisanje, dodajanje)
- Uporabite lahko Node DOM API:
 - `element.cloneNode()`
 - `element.removeChild()`
 - `element.replaceChild()`
- Node DOM API: <https://developer.mozilla.org/en-US/docs/Web/API/Node>

Zahteve

4. Ustvarite naključnih 10000 komentarjev, ki pripadajo prvem VDOM-u ter ga izrišite.

- Naredite kopijo danega VDOM-a, da dobite drugi VDOM ter v njem spremenite (lahko v JS) vsebino in barvo teksta naključnih X komentarjev.
- Nato posodobite prvi VDOM glede na spremembe drugega z operacijo diff.

Zahteve

- Izmerite čas ponovnega izrisa celotnega spremenjenega prvega VDOM-a, ali izrisa le spremenjenih komentarjev (**+ upoštevanje časa operacije diff**). Eksperiment ponovite pri $X=\{10, 20, 50, 100, 300, 500\}$.
- Izrišite črtni grafikon, kjer x os definira parameter X , y os pa izmerjen čas za oba možna izrisa. (5%)

Zahteve

- Ni dovoljena uporaba zunanjih knjižnic ali ogrodij.
- **Oddajte naslednje v arhivu ZIP:**
 - ogrodje (npr. ogrodje.js) = npr. vsebuje funkcije kot kreiranje VDOM elementov, izrisovanje, diff
 - demo aplikacija (npr. demo.html + demo.js) = za izris in spremembo 10000 komentarjev
 - grafikon_hitrosti.png = merjenje časa demo aplikacije
- **Vrednost naloge:** 12% od celotne ocene pri predmetu