

### Doelstellingen

Tijdens dit labo maken we nog een oefening op DOM, zonder jQuery. De topics van vandaag zijn dus:

- DOM
- Events

Bij wijze van uitbreiding zullen we een typisch Javascript app patroon

### Werkwijze, timing en upload

Dit labo loopt over één sessie en moet op het einde van de labosessie geupload worden via Toledo. Upload één enkele zip met daarin je oplossingen. De docent kan vragen de opdrachten thuis verder af te werken – je kunt dan wat extra punten verdienen. Neem die dan gewoon het volgende labo mee; je hoeft niet opnieuw te uploaden.

Het huishoudelijk reglement is van toepassing.

### Opgaves

#### 1. Sudoku

*Opmerking over de opgave: de HTML maakt gebruik van de HTML5 `contenteditable` attribuut; dit werkt niet in Internet Explorer voor tabelcellen.*

Open de opgave voor de Sudoku. We zullen dit in verschillende stappen scripten tot een volwaardig Sudoku spel.

##### 1.1 Verbinden van het script aan de HTML

Maak een script aan en verbind het aan je HTML; gebruik het script encapsulation patroon in strict mode.

*Tip: inspireer je op [http://ikdoeict.be/leercentrum/slides/javascript/02\\_dom.html#/script-encapsulation](http://ikdoeict.be/leercentrum/slides/javascript/02_dom.html#/script-encapsulation)*

##### 1.2 Variabelen

Definieer en initialiseer drie variabelen met de bijhorende DOM node(s):

- de tabel zelf
- de rijen van de tabel

- alle cellen van de hele tabel

Vind hiervoor de juiste DOM selectoren; controleer met `console.log()`.

### 1.3 Window load event

Telkens in een cel in editmode op een toets gedrukt wordt, zullen we

- de celwaarde instellen (methode `setValue`)
- de cel verlaten ('blur')
- het gehele bord controleren (methode `checkBoard`)

Werk dit uit maar laat de methodes `setValue` en `checkBoard` nog leeg; denk al wel na over de eventuele parameter(s).

### 1.4 Uitwerking `setValue()`

Dit is een zeer korte methode. Zorg ervoor dat enkel een getal tussen 1 en 9 geaccepteerd wordt; gebruik de `textContent` property van de cell om de gecontroleerde waarde effectief in de cel te plaatsen.

### 1.5 Uitwerking `checkBoard()`

Controleer tenslotte het spelbord na elke toets. We beginnen met enkel de horizontale rijen te controleren op dubbele waarden. Elke cel waar een dubbele waarde staat, zullen we de CSS-class 'invalid' toekennen.

- wis eerst van alle cellen de invalid state
- overloop nu alle rijen één voor één; markeer alle cellen waar er dubbels voorkomen als invalid
- tel het aantal lege cellen, en pas de 'nog in te vullen' tekst aan
- tel het aantal foutieve cellen, en pas de 'fouten: ' tekst aan
- het spel is afgelopen als het aantal lege cellen en het aantal fouten 0 is; bedenk een passende visuele feedback

*Tip: de stijl geassocieerd met de class "invalid" is reeds voorzien in het CSS bestand*

### 1.6 Bijkomende opdrachten (optioneel)

Naar keuze:

- controleer ook de kolommen, of
- zet het om naar het App of Game patroon (zie theorie)