

Web 2 – Test 2 van kans 1

We rekenen op ieders eerlijkheid en spreken het volgende af :

- **Je werkt de opdracht individueel uit, zonder hulp van anderen!**
 - communicatie met medestudenten of externen over de opdracht/oplossing is verboden.
 - zorg dat je GSM op stil staat en in je rugzak of jas blijft zitten.
 - je kijkt niet op elkaars scherm, stuurt elkaar geen berichten, wisselt geen bestanden uit, geeft elkaar geen hints, etc.
 - elk vermoeden van fraude zal bij de fraudecommissie worden gemeld (zie OER)
- Je mag tijdens de test informatie opzoeken in het cursusmateriaal en oplossingen vanop het leerpad (als je ze op je laptop hebt staan tenminste)
 - echter geen oplossingen van andere tests die je via-via kreeg 😊
- Je laptop staat gedurende de ganse test **in vliegtuigmodus**
 - je mag niks opzoeken op het internet
 - als je wil indienen, roep je eerst de docent en vraagt of je terug op internet mag
 - heb je geen toelating gevraagd en je gaat toch op internet, dan wordt dit als fraude beschouwd.
- Je hebt 2u20 tijd (IOEM : 2u55min) om deze opdracht af te werken en in te dienen
 - Je uploadt je oplossing op Chamilo **vóór** de tijd op is

Unzip de file 'beginsituatie opdracht recept.zip' in een nieuw Webstorm project of folder en noem dit

"familienaam voornaam test 2"

Je mag enkel het Javascript bestand veranderen, de HTML en CSS files laat je ongewijzigd.

Je oplossing laat de gebruiker toe om een recept aan te passen en bij te houden wat hij/zij al in huis heeft. Het recept bestaat uit een lijst van ingrediënten waarbij elk een benodigde hoeveelheid vermeldt.

Als je klaar bent ziet het er zo uit :

The screenshot shows a web form titled 'Recept voor tartiflette'. At the top, there is an 'Ingredient invoer (hoeveelheid per persoon!)' section with a text input field containing '2', a label 'Beschrijving van het ingredient', and a 'Voeg ingredient toe' button. Below this, the text 'Recept voor tartiflette' is followed by 'Ingredienten voor 2 personen'. A list of ingredients follows, each with a checkbox and a text description: '125 gram gerookte spek' (unchecked), '2 x ui' (checked), '5 cl witte wijn' (unchecked), '8 cl room' (unchecked), '0.62 kilo aardappelen' (unchecked), '0.50 x reblochon kaas' (unchecked), 'zout' (checked), and 'peper' (unchecked).

Figuur 1 : net na het opstarten

The screenshot shows the same web form after modifications. The 'Ingredient invoer' section now has a text input field containing '1', a label 'x glas whiskey voor de kok', and a 'Voeg ingredient toe' button. The text 'Recept voor tartiflette' is followed by 'Ingredienten voor 3 personen'. The list of ingredients is updated: '187.50 gram gerookte spek' (unchecked), '3 x ui' (checked), '7.50 cl witte wijn' (unchecked), '12 cl room' (unchecked), '0.93 kilo aardappelen' (unchecked), '0.75 x reblochon kaas' (checked), 'zout' (unchecked), 'peper' (unchecked), and '3 x glas whiskey voor de kok' (checked).

Figuur 2 : na enkele wijzigingen

De functionaliteit van deze applicatie bestaat uit drie delen :

- **Aan- en afvinken** : door op een ingredient te klikken kan men aanduiden dat men het wel/niet in huis heeft. Al naargelang verschijnt er een vinkje en wordt de tekst doorstreept.

Bv. in de overgang van figuur 1 naar 2, werd (o.a.) op de ingredienten 'zout' en 'kaas' geklikt.

- **Toevoegen** : men kan een nieuw ingredient toevoegen door bovenaan in de tekstvelden de hoeveelheid en beschrijving in te vullen en vervolgens op de knop te klikken. Het ingredient verschijnt dan onderaan de lijst en gedraagt zich net zoals alle andere ingredienten.

Bv. in de overgang van figuur 1 naar 2, werd (o.a.) een nieuw ingredient 'whiskey' toegevoegd (je ziet bovenaan nog hoe de invoervelden werden ingevuld). Het nieuwe ingredient werd daarna aangevinkt door erop te klikken. Merk op dat de benodigde hoeveelheid van dit nieuwe ingredient afgestemd is op 3 personen.

- **Aanpassen** : als men het aantal personen aanpast, wordt automatisch de benodigde hoeveelheid van elk ingredient herberekend

Bv. in de overgang van figuur 1 naar 2, werd het aantal personen naar '3' veranderd. Merk op dat de hoeveelheden van de ingredienten nooit '.00' op het einde tonen (vergelijk bv. ingredient 'spek' in de recepten voor 2 en 3 personen, idem voor 'witte wijn').

Bekijk nu eerst eens het HTML-document, meer bepaald het `` element met id 'lstIngredients'.

Elk ingrediënt is een `` element van de volgende vorm (bv. voor 2 personen) :

- Voor een *ingredient zonder hoeveelheid*
`peper`
- Voor een *ingredient met hoeveelheid*
`5 cl witte wijn`

Je ziet dat er bij een *ingredient met hoeveelheid*, telkens een aparte `` is voorzien waarin de benodigde hoeveelheid staat. In het voorbeeld hierboven staat er '5'.

Dit `` element bevat een custom data-attribuut 'data-amount-per-person', dat aangeeft hoeveel er per persoon wordt voorzien. Wanneer het aantal personen wijzigt, wordt deze waarde vermenigvuldigd met het nieuwe aantal personen en het resultaat in de `` gezet.

Vandaar dat er hierboven '5' staat in de , er wordt immers 2.5 per persoon voorzien en we kozen in dit voorbeeld voor 2 personen (2x2.5 = 5).

In de gegeven HTML-code staat er trouwens '???' in elke ``, het is echter de bedoeling dat dit bij de start van de applicatie vervangen wordt door een correct getal zodat de gebruiker de '???' nooit te zien krijgt.

Bij een *ingredient met hoeveelheid*, heeft het `` element heeft een class 'amount'. Deze class is aanwezig zodat je dergelijke ingrediënten makkelijk uit de DOM-tree kunt opvissen. Dit kon ook zonder class op basis van een CSS attribuut selector, maar een CSS class selector is eenvoudiger 😊

Je mag de inhoud van de HTML en CSS bestanden niet wijzigen! Je zult enkel het Javascript bestand moeten aanpassen, om de volgende functies te implementeren :

- setup
- toggleCheckmark
- addIngredient
- updateHoeveelheden

Wat deze precies moeten doen wordt in de volgende secties uitgelegd.

Tip : Bouw je oplossing stapsgewijs op. Hieronder staan 4 genummerde stappen, probeer die in de gegeven volgorde uit te voeren. Tussendoor kun je altijd wat console output produceren om te zien of je de correcte waarden hebt en of de juiste functies worden opgeroepen.

Verwijder achteraf wel alle console output want die kost punten als hij blijft staan.

Stap 1 : initialisatie

Alle initialisatie gebeurt door de **'setup'** functie, zorg ervoor dat deze op de correcte manier wordt opgeroepen bij het inladen van de pagina.

Je zult hier natuurlijk (ook) allerlei event listeners moeten koppelen. Welke event listeners er nodig zijn, zal (pas) in de volgende stappen duidelijk worden. Je zult de setup functie op dit moment dus nog niet kunnen afwerken.

Onderstaande initialisatie wordt eigenlijk pas relevant als je aan stap 4 begint.

(je kan dus gerust stappen 2 en 3 oplossen terwijl je oplossing nog '???' toont bij de benodigde hoeveelheden van sommige ingrediënten)


In de HTML-code zie je dat de benodigde hoeveelheid voor elk ingrediënt (i.e. de tekst in een `` met class 'amount') letterlijk '???' is. Zoals gezegd moet dit bij het starten van de applicatie vervangen worden door het correcte getal, zodat de gebruiker de '???' nooit ziet.

Deze benodigde hoeveelheid moet ook bij het opstarten berekend worden op basis van wat er in de HTML-code staat, je mag de getallen niet simpelweg in je Javascript hardcoderen.

Anders gezegd, ook bij het initialiseren zul je de waarde uit het invoerveld moeten opvragen en combineren met de waarden van de custom data-attributen om zo de benodigde hoeveelheden in te vullen. Indien je in de HTML-code bv. de waarde van het invoerveld op '3' zou zetten en/of enkele waarden van die custom data-attributen zou aanpassen, dan moet bij het herladen alles nog steeds kloppen.

Stap 2 : aan- en afvinken

De gebruiker kan een ingrediënt aan- of afvinken door erop te klikken.

Gans het gebied van het `` element is klikbaar en vertoont hetzelfde klikgedrag, dus zowel de tekst van het `` element, de `` alsook de checkbox vooraan kunnen geklikt worden. En omdat deze `` elementen, block elementen zijn kun je zelfs in het lege stuk rechts van de tekst klikken. Je kunt dit klikbare gebied visueel herkennen aan de mouse pointer  als die over een ingrediënt hoovert.

Maak je over de checkbox geen zorgen, deze wordt m.b.v. wat clevere CSS toegevoegd. Als je nieuwsgierig bent, kun je na de test eens uitpluizen hoe dit werkt.

Het aan- en afvinken van een ingrediënt is gewoon een kwestie van het al dan niet aanwezig zijn van een **‘checked’ class** op het `` element. Je kan dit makkelijk zien in de HTML-code :

- Aangevinkt
`<li class="checked">zout`
- Afgevinkt
`peper`

Voor *ingredienten met hoeveelheid* werkt dit precies op dezelfde manier, zoals gedemonstreerd in de HTML-code (bv. *ingredient met hoeveelheid* ‘ui’ is ook aangevinkt).

De code die het aan- of afvinken bewerkstelligt, zit volledig in functie **‘toggleCheckmark’**.

Normaliter kun je het met die ene functie voor elkaar krijgen, maar naargelang hoe je de click events afhandelt (bv. al dan niet steunen op event bubbling) kan het zijn dat je nog een extra functie moet toevoegen. Om dezelfde reden mag je zelf kiezen welke parameter(s) jouw functie ‘toggleCheckmark’ heeft.

Terzijde, niet relevant voor de test : deze oplossing voor het aan- en afvinken van een checklist staat volledig los van het recept verhaal en zou dus perfect kunnen dienen voor andere lijstjes. Dat is trouwens ook de reden waarom het `` element aparte classes ‘checkboxlist’ en ‘IstIngredients’ heeft. De CSS-code werd hiervoor in een aparte file geplaatst en mits een kleine inspanning zou je dit ook kunnen doen met de relevante code uit je javascript bestand.

Stap 3 : toevoegen

De gebruiker kan bovenaan de pagina een nieuw ingrediënt ingeven dat, na een klik op de knop, onderaan het recept verschijnt.

Er zijn twee tekstvelden voorzien en een knop :

Ingrediënt invoer (hoeveelheid per persoon!)		
<input type="text"/>	Beschrijving van het ingrediënt	Voeg ingrediënt toe

De gebruiker vult links de hoeveelheid in per persoon en rechts een beschrijving. Let op : het al dan niet leeg zijn van het linker veld, bepaalt het soort ingrediënt :

- Voor een *ingrediënt zonder hoeveelheid*

Ingrediënt invoer (hoeveelheid per persoon!)		
<input type="text"/>	basilicum naar smaak	Voeg ingrediënt toe

- Voor een *ingrediënt met hoeveelheid*

Ingrediënt invoer (hoeveelheid per persoon!)		
1.5	x glas gin-tonic voor de kok	Voeg ingrediënt toe

Door op de knop te klikken, wordt het ingrediënt onderaan het recept toegevoegd. In een recept voor 2 personen krijgen we dan voor bovenstaande ingrediënten :

- ☐ basilicum naar smaak
- ☐ 3 x glas gin-tonic voor de kok

De invoervelden worden niet automatisch leeggemaakt na het toevoegen van een ingrediënt.

Denk eraan dat een nieuw ingrediënt zich net zo moet gedragen als een ingrediënt dat reeds in de HTML-code aanwezig was. Je kan het dus ook aan- en afvinken en de benodigde hoeveelheid moet zich aanpassen aan het aantal personen (zie stap 4).

De functie '**addIngrediënt**' is de event listener voor de knop en doet het nodige om een nieuw ingrediënt in de DOM-tree te krijgen (dat tevens het juiste gedrag vertoont, zie stappen 2 en 4).

Belangrijk : na het toevoegen van een nieuw ingrediënt, moet diens getoonde hoeveelheid kloppen voor het ingestelde aantal personen. Bijvoorbeeld, we vulden hierboven '1.5' in voor het aantal glazen gin-tonic maar omdat het aantal personen op '2' staat wordt in het recept de hoeveelheid '3' getoond ($2 \times 1.5 = 3$).



Tip van Flip : de functie 'updateHoeveelheden' (zie stap 4) herberekent de hoeveelheden van alle ingrediënten, dus inclusief het nieuwe ingrediënt - als je het slim aanpakt!

Stap 4 : aanpassen

De gebruiker kan het recept makkelijk aanpassen aan het aantal personen.

Hiervoor is een eenvoudig numeriek invoerveld voorzien :

Ingrediënten voor personen

Zodra de gebruiker deze waarde wijzigt, passen de hoeveelheden van de ingrediënten zich aan.

Het event dat je hiervoor kunt gebruiken bij het `<input>` veld, heet 'input'. We zagen dit al eens in de colorpicker opdracht. Telkens de waarde wijzigt, wordt de 'input' event listener van het `<input>` element opgeroepen.

Het programma interpreteert de ingevoerde tekst als een geheel getal, dus als we '2.65' zouden invoeren (of '2,65' naargelang je instellingen) dan neemt het programma gewoon '2'. Indien het invoerveld leeggemaakt wordt, gebeurt er niks. Zorg dat je programma in deze situatie geen fout produceert omdat het plots met een lege string te maken krijgt!

Ter herhaling: elk *ingredient met hoeveelheid* heeft een `` met een custom data-attribuut dat aangeeft wat de hoeveelheid is voor 1 persoon.

Deze ('hoeveelheid per persoon' maal 'aantal personen') geeft ons de **benodigde hoeveelheid** van dat ingredient in het recept en wordt getoond in het `` element.

Je mag ervan uitgaan dat de `` elementen in de HTML-pagina steeds correct zijn opgesteld, zoals beschreven bovenaan pagina 3 van deze opgave.

De functie '**updateHoeveelheden**' herberekent voor alle *ingredienten met hoeveelheid*, de benodigde hoeveelheid in het recept en toont dit in hun `` element. Deze functie is natuurlijk de 'input' event listener bij het invoerveld.

Hint : je zult deze functie natuurlijk moeten (laten) oproepen telkens het aantal personen wijzigt, maar bedenk dat je ze ook 'handmatig' kunt oproepen op andere plaatsen in de code. Meer bepaald in stappen 1 en 3 als de hoeveelheden (nog) niet kloppen!

De benodigde hoeveelheid afronden

Normaliter wordt de benodigde hoeveelheid afgerond getoond op 2 cijfers na de komma.

Bv. indien de hoeveelheid per persoon '1.125' is en we stellen het recept in op '3' personen, dan is de benodigde hoeveelheid '3.375' (want $1.125 \times 3 = 3.375$). Het programma toont echter niet '3.375' maar toont dit afgerond op 2 cijfers na de komma, dus '3.38'.

De benodigde hoeveelheid als geheel getal

Indien de benodigde hoeveelheid mooi uitkomt op een geheel getal, dan wordt deze als een geheel getal getoond.

Bv. indien de hoeveelheid per persoon '1.25' is en we stellen het recept in op '4' personen, dan is de benodigde hoeveelheid '5.00...' maar we tonen deze netjes als '5'.

In het gegeven recept kun je dit makkelijk uitproberen voor ingrediënten 'spek' en 'witte wijn'. Als je figuren 1 en 2 er nog eens bijneemt, zul je zien dat hun benodigde hoeveelheid de ene keer als kommagetal getoond wordt en de andere keer als geheel getal.

Hoe kun je nu weten of een Number value in Javascript al dan niet een integer (i.e. geheel getal) is?

Er bestaat een `Number.isInteger(..)` functie om dit te doen :

- `Number.isInteger(5.00)` retourneert true
- `Number.isInteger(3.75)` retourneert false

Mocht je je afvragen wat `isInteger(..)` retourneert in andere gevallen, dan kun je dit makkelijk op de console uitproberen in je browser.

Indienen

Stop je volledige folder in een zip file 'familienaam voornaam test 2.zip' en dien dit in op Chamilo. Als ik je file unzip dan moet alles gewoon werken, zonder hernoemen of kopiëren van files.

Je oplossing moet absoluut vóór het einduur op Chamilo staan!

Veel succes!