

### **Algemene instructies examen**

- Alle bestanden zijn reeds voor je aangemaakt. Deze bestanden mag je niet van naam en niet van plaats veranderen.
- Je mag enkel wijzigingen aanbrengen in de head van je HTML-bestand en in de javascript bestanden. Aan de data mag je NIETS veranderen. Doe je dat wel krijg je automatisch 0 op de vraag.
- De examens zullen verbeterd worden met behulp van Google Chrome. Controleer je werk dus ook in deze browser.

*OPGELET! Het niet volgen van bovenstaande instructies resulteert, per inbreuk, in puntenverlies! Mogelijks tot 100% puntenverlies op één of meerdere deelvragen.*

**VEEL SUCCES!**

## Opdracht 1

(11 punten)

Schrijf een stukje code dat een select element maakt en vult met de 5 orders uit de orders array. Elke optie in het select element moet de tekst “order {orderId} – {date}” bevatten, waarbij de datum in de volgende indeling moet zijn: “June 11, 2023”.

order 1 - June 11, 2023 ▼

order 1 - June 11, 2023

order 2 - July 7, 2023

order 3 - October 23, 2023

order 4 - December 16, 2023

order 5 - January 7, 2024

Wanneer een order is geselecteerd, moeten de producten in de orderLines array die overeenkomen met de geselecteerde order worden weergegeven op de pagina. Ook als de pagina net voor het eerst geladen is moeten de producten uit het eerst geselecteerde order zichtbaar zijn op de pagina. Deze producten toon je in een ongeordende lijst.

De producten moeten echter in een specifieke volgorde op de pagina worden getoond. Het goedkoopste product (per stuk) moet als eerste komen en het duurste product moet als laatste komen. Als de prijs van producten identiek is moeten de producten op basis van de lengte van de description gerangschikt worden. De langste description eerst.

order 1 - June 11, 2023 ▼

- Headset - Logitech G533 - 1 x 99 EUR = 99 EUR
- Graphics Card - Nvidia RTX 3080 - 2 x 699 EUR = 1398 EUR

order 5 - January 7, 2024 ▼

- Mouse - Logitech G305 - 1 x 39 EUR = 39 EUR
- Keyboard - Logitech G413 (new edition) - 2 x 99 EUR = 198 EUR
- Keyboard - Logitech G413 - 1 x 99 EUR = 99 EUR
- Headset - SteelSeries Arctis 7 - 1 x 129 EUR = 129 EUR
- Graphics Card - Nvidia RTX 3070 - 3 x 499 EUR = 1497 EUR

Bereken vervolgens d.m.v. een functie het gemiddelde bedrag van alle orders in de orders array. Als het totale bedrag van de order hoger is dan het gemiddelde bedrag, moet het totale bedrag van het geselecteerde order vetgedrukt, groen en onderstreept worden weergegeven onder de producten. Als het totale bedrag van de order lager is dan het gemiddelde bedrag, moet het totale bedrag cursief, rood en met lettergrootte 20 worden weergegeven.

order 1 - June 11, 2023 ▼

- Headset - Logitech G533 - 1 x 99 EUR = 99 EUR
- Graphics Card - Nvidia RTX 3080 - 2 x 699 EUR = 1398 EUR

Total cost: 1497 EUR

order 2 - July 7, 2023 ▼

- Mouse - Logitech G502 - 2 x 49 EUR = 98 EUR
- Keyboard - Corsair K70 - 3 x 129 EUR = 387 EUR

Total cost: 485 EUR

Laat als laatste nu nog de bank zien die bij het rekeningnummer van het order hoort. Dit kun je als volgt berekenen:

Ligt het nummer van drie cijfers na de eerste ‘-’ tussen de 001 en de 049, dan moet het logo van BNP Paribas getoond worden. Ligt het cijfer tussen 050 en 099, dan moet Belfius getoond worden. Is het cijfer 185, dan moet ING getoond worden.

De alternatieve tekst van de afbeelding moet de naam van de bank zijn.  
De afbeelding moet 200px breed zijn.

'BE41-063-0123456-10'

order 1 - June 11, 2023

order 2 - July 7, 2023

• Headset - Logitech G533 - 1 x 99 EUR = 99 EUR

• Graphics Card - Nvidia RTX 3080 - 2 x 699 EUR = 1398 EUR

Total cost: 1497 EUR

• Mouse - Logitech G502 - 2 x 49 EUR = 98 EUR

• Keyboard - Corsair K70 - 3 x 129 EUR = 387 EUR

Total cost: 485 EUR



## Opdracht 2

(9 punten)

Voor deze opdracht ga je een aantal functies moeten schrijven die een bepaalde output moeten geven. Elk van je functies moet blijven werken als er in de toekomst eender waar meer producten zouden worden toegevoegd aan de array. Is dit niet het geval dan zal je code voor de delen die dan niet meer zouden werken geen punten opleveren.

Schrijf voor elke functie `console.log()`'s die hetzelfde uit zien als de screenshots onder 'output' bij elke oefening: bijvoorbeeld:

```
console.log('Functie 1:', filterFoodsByFairTradeBrands(products));
```

Correcte output:

```
▼ Functie 1:
  ▼ [{...}] ⓘ
    ▶ 0: {food: 'Milk', calories: 42, deelbaarDoorTwee: true}
      length: 1
    ▶ [[Prototype]]: Array(0)
```

Foute output:

```
Functie 1: [object Object] vraag2.js:120
```

Per ontbrekende console log of log van een onjuist formaat zal er een punt worden afgetrokken.

Opgaves:

Functie 1:

Schrijf een functie die een array van alle producten waarvan alle brands fair-trade zijn, teruggeeft. Echter niet het hele object moet teruggegeven worden maar enkel de naam van het voedsel beginnende met een hoofdletter, het aantal calorieën, en of het aantal karakters in de naam deelbaar is door 2. Dit laatste moet natuurlijk berekend worden.

Output:

```
▼ Functie 1:
  ▼ [{...}] ⓘ
    ▶ 0: {food: 'Milk', calories: 42, deelbaarDoorTwee: true}
      length: 1
    ▶ [[Prototype]]: Array(0)
```

## Functie 2:

Schrijf een functie die alle producten van 50 calorieën zoekt.

Vervolgens moet voor elk merk van deze producten waarvan de naam van het merk een streepje bevat een object gecreëerd worden met de volgende properties:

- nameWithStars: De naam van het merk maar de '-' zijn vervangen met '\*\*\*'
- lastPart: een **functie** die het laatste deel van nameWithStars (dus de string na de laatste '\*\*\*') teruggeeft. Lees binnen deze functie de property nameWithStars uit om tot je resultaat te komen.

Deze objecten moeten in een array worden gestopt en de array moet teruggegeven worden door de functie.

Output: Opgelet! Het laatste item in de screenshot hieronder zou niet in het resultaat mogen zitten.  
De screenshot klopt niet.

```

Functie 2: vraag2.js:146
▼ (4) [{...}, {...}, {...}, {...}] ⓘ
  ▶ 0: {nameWithStars: 'Fair***Trade***USA', lastPart: f}
  ▶ 1: {nameWithStars: 'Equal***Exchange', lastPart: f}
  ▶ 2: {nameWithStars: 'Haagen***Dazs', lastPart: f}
  ▶ 3: {nameWithStars: 'Cal***Organic Farms', lastPart: f}
    length: 4
  ▶ [[Prototype]]: Array(0)
Output na uitvoeren functie eerste item in array: USA vraag2.js:147
```

## Functie 3:

Deze functie moet als resultaat één lange string teruggeven.

Voor elk product in de array berekent de functie de som van alle werknemers van de merken die eraan toebehoren.

Als de som van het aantal werknemers groter is dan 20000, wordt de naam van het voedsel omgekeerd aan de resultaatstring toegevoegd. Als het aantal werknemers niet groter is dan of gelijk is aan 20000, wordt de kortste merknaam van de merken van dat voedsel in hoofdletters aan de resultaatstring toegevoegd.

Elk van de waardes in de resultaatstring moet gescheiden worden met een komma.

Output:

```

Functie 3: elppa,ananab,yrrehc,GRIMMWAY vraag2.js:168
FARMS,eseehc,FAGE,ORGANIC VALLEY,hcanips
```