# Oefeningen Onderdeel 2

# Even opwarmen …

## Oefening 1 – 6 Variaties op hello world

Bijna elk boek of elke cursus start met een hello world applicatie. Wij kunnen uiteraard niet achterblijven.

Toon de boodschap “Hello world” op volgende manieren:

1. In een html document, zonder css en zonder javascript. Gewoon statische tekst dus.
2. In een html document, met inline css, waarbij je de boodschap in het blauw toont op een witte achtergrond. Nog steeds zonder javascript.
3. In een html document, met een console.log die je toont met inline javascript.
4. In een html document, met een alert die je toont met inline javascript.
5. In een html docment, met een alert die je toont vanuit een **apart javascript bestand**.
6. In een html docment, in een input box nadat je op een knop drukt (waarbij je de code ook weer in een apart javascript bestand zet). Je mag deze variatie naar hartelust stylen.

## Oefening 2 – Controleer op even getallen

Maak een pagina waar een gebruiker een getal kan invullen. Als het getal even is toon je de boodschap ‘Het getal is even’ in een groene kleur. Als het getal oneven is toon je de boodschap ‘Het getal is oneven’ in een rode kleur.

Gebruik geen alerts. Gebruik geen inline JavaScript.

# De pijlers – Deel 1

## Oefening 3 – URl’s in .NET

Maak een .NET Console App in Visual Studio:

1. Die controleert of een bepaalde string een geldige http of https URL is.
2. Indien een geldige URL
   1. Alle onderdelen van de URL toont
   2. Scheme, host, port, query en fragment
3. Tip
   1. Gebruik MSDN en stackoverflow bij problemen of vragen.
   2. Meer bepaald deze links: [Uri (MSDN)](https://docs.microsoft.com/en-us/dotnet/api/system.uri?view=netframework-4.7.2) en [Is there a URL validator on .Net (SO)](https://stackoverflow.com/questions/927847/is-there-a-url-validator-on-net).

## Oefening 4 – URL’s in Javascript

Idem als vorige oefening maar maak nu een HTML pagina met een JavaScript bestand:

1. Die controleert of een bepaalde string een geldige http of https URL is.
2. Indien een geldige URL
   1. Alle onderdelen van de URL toont
   2. Scheme, host, port, query en fragment
3. Tip:
   1. Gebruik MSDN en stackoverflow bij problemen of vragen.
   2. Meer bepaalde deze link: [Check if a JavaScript string is a URL](https://stackoverflow.com/questions/5717093/check-if-a-javascript-string-is-a-url).

## Oefening 5 – Web server

Nog twee variaties op “Hello world”:

1. Je laatste uitwerking in de oefening 1 mag je nu laten ‘serven’ door een NodeJS server.
2. Doe nu hetzelfde door middel van een ASP.NET Core server.

In beide gevallen zal je er dus moeten voor zorgen dat de server ‘static’ files kan serven.

## Oefening 6 – Toevoegen van header

Pas de vorige oefening aan zodat je een ‘**Cache-Control**’ header toevoegt aan elke request. Deze header moet aangeven dat de browser de bestanden maximaal 15 seconden mag cachen.

Tips:

* Tip: [headers toevoegen aan static files in NodeJS-express](https://regbrain.com/article/cache-headers-express-js).
* Tip: [headers toevoegen aan static files in ASP.NET Core](https://docs.microsoft.com/en-us/aspnet/core/fundamentals/static-files?view=aspnetcore-2.1&tabs=aspnetcore2x).
*  Opgelet: als je test in Chrome zal je merken dat Chrome sowieso steeds een nieuwe request uitstuurt: [Is Chrome ignoring Cache-Control (SO)](https://stackoverflow.com/questions/11245767/is-chrome-ignoring-cache-control-max-age)? Dat komt omdat Chrome ervan uitgaat dat je een refresh doet omdat de pagina niet correct is (= dus Chrome implementeert hier de web standaard niet, om gebruiksvriendelijker te zijn). Om te testen open je dus best gewoon een andere tab, dan zal Chrome die ‘hack’ niet doen.



## Oefening 7 – Tracker cookie

Pas de vorige oefening aan met een ‘tracker cookie’:

* Indien geen cookie op een binnenkomende request: voeg een cookie toe waarin een UUID/GUID zit.
* Indien wel een cookie op een binnenkomende request: log de cookie op de console.

Controleer of de cookie goed toekomt bij de browser en of deze steeds wordt meegestuurd bij refreshes.

Opgelet: [How the GDPR affects cookie policies](https://www.itgovernance.eu/blog/en/how-the-gdpr-affects-cookie-policies)



## Oefening 8 – Analyseer een bestaande webSITE of WEBapplicatie

Analyseer een website of webapplicatie die je regelmatig bezoekt. Hiervoor maak je gebruik van de developer tools: wat valt je op? Zijn er errors? Hoeveel files worden er gebruikt?

## Oefening 9 – wedstrijd webapplicatie voor een product

Maak een website voor een wedstrijd rond een bepaald product (vb. een nieuwe spelconsole).

De eerste pagina is een ontvangstpagina met een knop ‘Deelnemen’. Deze ‘Deelnemen’ knop leidt de gebruiker naar een formulier (°) waar twee vragen worden gesteld: (1) een vraag over het product en een (2) een schiftingsvraag. Op deze pagina staat een ‘Indienen’ knop: deze leidt de gebruiker naar een formulier waar contactgegevens (°) moeten ingevuld worden. Na ingave van alle gegevens wordt een ‘Bedankt voor uw deelname’ pagina getoond.

(°) Benodigde contactgegevens: voornaam, familienaam, stad/gemeente, email adres en telefoonnummer.