

# Oefeningen JavaScript, ECMAScript 2015 en TypeScript - CM Telecom

## 01 – JAVASCRIPT DESIGN PATTERNS

- a) Maak een eigen object (bijvoorbeeld `Person`, `Product`, `Car` of anders) en breidt dit uit op verschillende manieren
  - o Prototype pattern
  - o Revealing module pattern
  - o Revealing prototype pattern
  - o Maak een herdistribueerbare module van je object en bespreek/leg uit aan een collega hoe hij wordt gebruikt.
- b) Begin met een eigen DOM-library à la jQuery en zorg er voor dat er twee of drie publieke functies zijn die het volgende kunnen:
  - o Een element selecteren in het DOM (`.get()`).
  - o Een element verbergen (`.hide()`)
  - o Een element tonen (`.show()`)
  - o Maak een kleine demo van de mogelijkheden, eventueel met een knop en event handler (klikken = element verbergen, nogmaals klikken = element tonen).
  - o Gevorderd: voor meer informatie over eigen libraries construeren, zie bijvoorbeeld <http://net.tutsplus.com/tutorials/javascript-ajax/build-your-first-javascript-library/>. Let hierin voornamelijk op de manier om meerdere elementen in het DOM te selecteren/mappen.
- c) Lees meer over JavaScript design patterns bij Addy Osmani:  
<http://addyosmani.com/resources/essentialjsdesignpatterns/book/>.
  - o Zoek of je de in de cursus behandelde patronen kunt terugvinden en bestudeer de codevoorbeelden (prototype pattern, etc.).
  - o Pik er twee of drie patronen uit en maak hiermee een eigen voorbeeld.
- d) Maak een eigen, herdistribueerbaar object, bijvoorbeeld een Stopwatch-module, of verzin zelf een object.
  - o De stopwatch moet in een pagina meerdere keren te instantiëren zijn (bijvoorbeeld `var sw1 = new Stopwatch()`, `var sw2 = new Stopwatch()`, etc).
  - o Elke stopwatch-instantie moet methods hebben als `start()`, `stop()`, `reset()`, `currentTime()`, en wat je verder maar handig vindt voor een stopwatch.
  - o Maak meerdere knoppen in de pagina, waarmee de stopwatch-instanties gestart, gestopt, enzovoort kunnen worden. De huidige tijd van de stopwatch wordt in een result-div op de pagina getoond.
- e) Maak een Parent-Child relatie tussen classes, waarbij de child-klasse alle members en methods uit de parentclass overerft.
  - o Denk aan: voertuigen, gebouwen, personen, producten, etc.
  - o Instantieer verschillende instanties en check of je in de child-classes zowel members als methods uit de parent-class kunt gebruiken, als uit de child-class zelf.
  - o Override een methode uit de parent-class in een child-class en test hoe dit werkt.
  - o Optioneel: geef zowel de parent- als de child-class een member met dezelfde naam (aangeraden!). Test in de instanties hoe dit werkt. Welke waarde heeft de member gekregen?