

## Oefeningen Web Scripting Week 3

*Maak de onderstaande programma's. Voer de programma's uit in de console (Node.js). Er hoeft niet met modules gewerkt te worden. Elke oefening mag in één bestand opgelost worden.*

(1) Maak een object-literal 'persoon' aan met daarin je naam, voornaam en de geboortedatum. Voor de geboortedatum gebruik je de voorgedefinieerde klasse Date.

[https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global\\_Objects/Date](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Date)

Maak in de object-literal de functie berekenleeftijd waarmee je je leeftijd kan berekenen. Maak ook de functie print die naam, voornaam en leeftijd afdruckt. Probeer alle functies uit.

(2) Herwerk oefening1. Maak gebruik van een class en voorzie setters en getters voor alle eigenschappen.

(3) Een bankrekeningnummer bestaat uit een landcode (2 posities), het banknummer (2 posities) en vervolgens uit 12 cijfers waarvan de eerste 10 een uniek nummer zijn en de laatste 2 het controlegetal zijn. Deze 2 laatste cijfers zijn het resultaat van een restberekening van de eerste 10 posities met het getal 97. Voorbeeld **BE 76 0012 5879 0995**

Schrijf een **klasse Account** waarin je apart de landcode, banknummer en de eerste 10 posities als argument van de constructor meegeeft. Schrijf vervolgens de functie print waarbij je de ganse banknummer afdruckt in de console, inclusief het controlegetal.

(4) Bereid de klasse Account uit met eigenschap saldo en voorzie methoden om geld te storten of te verwijderen van het account.

(5) Maak de klasse Vec die een vector in twee-dimensies voorstelt. Vec heeft eigenschappen x en y. In de constructor wordt een waarde voor x en y gespecificeerd. Voorzie de methode plus en minus. Beide methodes hebben een Vec object als parameter. De som of het verschil tussen de beide vectoren wordt als Vec object teruggeven:

$$(x1, y1) + (x2, y2) = (x1+x2, y1+y2)$$
$$(x1, y1) - (x2, y2) = (x1-x2, y1-y2)$$

Voorzie een getter voor length: als waarde voor length wordt de afstand tussen de vector en de oorsprong gegeven, hint

```
let x1=2;  
let x2=3;  
let lengte = Math.sqrt(x1*x1+y1*y1);
```

Voorzie de methode toString. Deze methode geeft een string terug. De terugkeerwaarde is van de vorm "(1,2)" voor een Vec met waarde voor x = 1 en waarde voor y = 2.

Maak de klasse InputException

```
class InputError extends Error {}
```

Zorg dat er een InputError opgeworpen wordt als de waarde voor x of y in de constructor niet een number is.

Zorg dat er een InputError opgeworpen wordt als de waarde voor het argument van plus en minus geen Vec-object is. Maak gebruik van try-catch block's om de exception handling van de constructor, plus en minus uit te proberen.

(6) Maak de klasse PolyLine. Deze klasse heeft 'vecs' een array van Vec-objecten als eigenschap. In de constructor wordt vecs als lege array aangemaakt.

Voorzie de methode addVec die een Vec-object aan vecs toevoegt. Er wordt een InputError opgeworpen als het argument geen Vec-object bevat.

Voorzie de methode toString die een voorstelling van alle Vec-objecten teruggeeft. De terugkeerwaarde is van de vorm "(1,2) (3,32) (2,2)" voor een polyline met drie Vec-objecten (1,2) (3,32), (2,2).

(7) Maak de klasse Date. Date heeft eigenschappen day, month, year.

Voorzie een constructor om een datum aan te maken, er worden waarden voor day, month en year meegegeven.

De datum moet in het formaat 1/1/2008 afgedrukt kunnen worden via de methode print(). Voorzie een extra methode printMonth() die de datum als volgt afdruckt: 1/jan/2008. Maak hiervoor gebruik van de static array MONTHS.

Voorzie setters en getters voor day, month en year.

Maak een static methode make om een datum aan te maken.

```
let date1=Date.make(1,2,2001);
```

## Extra oefeningen

(E.1) Schrijf een methode die kwadratische vergelijkingen oplost. Normaal gesproken heeft een dergelijke vergelijking 2 oplossingen. De methodeaanroep moet van de vorm  $LosOp(a,b,c, x1, x2)$  zijn

$$x1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$$

$$x2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$$

De oplossingen worden in de console getoond.

In het geval dat de discriminant  $\sqrt{b^2 - 4ac}$  negatief is, moet je een Error opgooien met een toepasselijke foutmelding.

(E.2) Maak de klasse Point met waarde voor x en y. Er worden functions setX, getX, setY, getY in deze klasse voorzien.

Maak de klasse WereldObject. Deze klasse heeft eigenschap location van datatype Point. Voorzie getX, setX, getY, setY in de klasse WereldObject.

Maak de klasse Hero. Hero is afgeleid van de klasse WereldObject. Maak de methode moveDown, moveUp, moveLeft, moveRight. Via deze methoden wordt de waarde van y verlaagd met 1, verhoogd met 1 of de waarde van x verlaagd met 1, verhoogd met 1.

(E.3) Als je de lengte van drie zijden a, b, c van een driehoek weet, dan kan de oppervlakte berekend worden door :  $Opp = \sqrt{s(s-a)(s-b)(s-c)}$

en waarbij  $s = (a+b+c) / 2$

Schrijf een methode om de oppervlakte te berekenen en te retourneren. Zorg dat Doe dit aan de hand van een object literal, waarbij je de methode implementeert in het object. Zorg voor de toepasselijke exceptions (en foutmeldingen) wanneer de drie lengten geen driehoek kunnen vormen.