

## Lab 8

Vandaag is: Mon Oct 28 2013  
factuurdatum is: Fri Oct 11 2013  
betaaldatum is: Sun Nov 10 2013  
U hebt nog: 12 dagen om te betalen

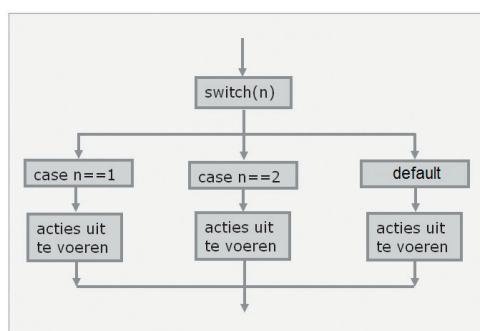
## 2.9 Switch

In deze paragraaf maken we kennis met de switch-opdracht:

Planning	Inleveren
	Opgave 58
	Vaardigheid-lab 09

De opdracht switch is ook een beslissingsstructuur. Deze kan efficiënter zijn dan de opdracht if. We gebruiken de opdracht switch om te beslissen welk blok uit veel blokken code moet worden uitgevoerd. De syntaxis van de opdracht switch is als volgt:

```
switch (n) {  
  case 1:  
    acties uit te voeren als n == 1  
    break;  
  case 2:  
    acties uit te voeren als n == 2  
    break;  
  default:  
    acties uit te voeren als n anders is dan 1 en 2  
}
```



Figuur 2.19 Stroomdiagram van de opdracht switch

- **Opgave 58**

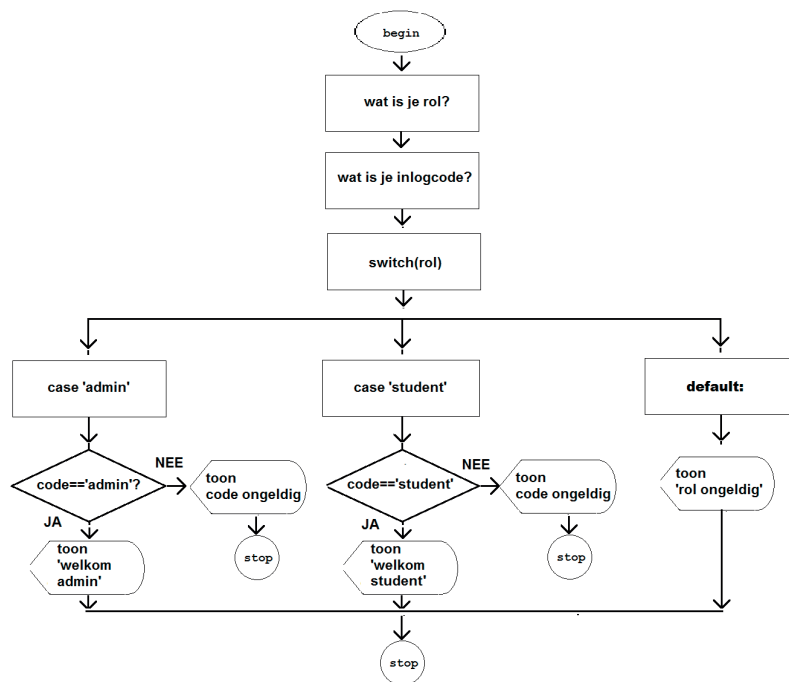
Open een nieuw bestand en sla het op als **switch-opdracht.html**. Voeg de volgende code eraan toe:

```
<!DOCTYPE html>
<html lang="nl">
<head>
  <meta http-equiv="Content-Type" content="text/html;
    charset=UTF-8">
  <title>switch-opdrachten</title>
</head>
<body>
  <script>
    var datum = new Date();
    var vandaag = datum.getDay();
    var uur = datum.getHours();
    console.log("Het is vandaag:" + datum);
    switch (vandaag) {
      case 1:
        console.log("maandag is mijn eerste stagedag");
        break;
      case 2:
        console.log("dinsdag is mijn tweede stagedag");
        break;
      case 3:
        console.log("woensdag is mijn JavaScriptdag");
        break;
      case 4:
        console.log("donderdag is mijn werkdag");
        break;
      case 5:
        console.log("vrijdag is mijn roostervrije dag");
        break;
    }
  </script>
</body>
</html>
```

- **Vaardigheid-lab 09**

Als je veel beslissingen of opties moet coderen met alleen de if-opdracht, eindig je met if-opdrachten binnen if-opdrachten binnen nog meer if-opdrachten. Dan wordt je code vrijwel onbegrijpelijk. Met de opdracht switch kunnen we op een meer gestructureerde manier beslissingen of opties coderen.

In deze lab-opdracht codeer je een script volgens onderstaand stroomdiagram. Dit script sla je op als **lab09.html**.



Figuur 2.20 Stroomdiagram bij Vaardigheid-lab 09

## 2.10 Eigen functies

In deze paragraaf maken we kennis met de volgende JavaScript-opdrachten en concepten:

- functies
- externe functies
- functieparameters
- scope van variabelen

Planning	Inleveren
	Opgaven 59 t/m 68
	Vaardigheid-lab 10

We hebben kennism gemaakt met een aantal methodes, bijvoorbeeld de methodes voor het Array-object en het Date-object. Maar we kunnen ook onze eigen functies coderen en hergebruiken. In deze paragraaf van hoofdstuk 2 leren we hoe we onze eigen functies coderen.