

```

20 <div class="slideshow-container">
21 <div class="mySlides1">
22 | 
23 </div>
24
25 <div class="mySlides1">
26 | 
27 </div>
28
29 <div class="mySlides1">
30 | 
31 </div>
32
33 <a class="prev" onclick="plusSlides(-1, 0)"&#10094;</a> <!-- Zorgt ervoor dat de functie plusSlides word uitgevoerd en dat er 1 vanaf gaat, dus hij gaat terug -->
34 <a class="next" onclick="plusSlides(1, 0)"&#10095;</a> <!-- Zorgt ervoor dat de functie plusSlides word uitgevoerd en dat er 1 bij komt, dus hij gaat verder -->
35 </div>
36
37 <p><br></p>
38 <div class="slideshow-container">
39 <div class="mySlides2">
40 | 
41 </div>
42
43 <div class="mySlides2">
44 | 
45 </div>
46
47 <div class="mySlides2">
48 | 
49 </div>
50
51 <a class="prev" onclick="plusSlides(-1, 1)"&#10094;</a> <!-- Het tweede getal is hier een 1 ipv 0 omdat dit een andere slider is, anders werkt het niet -->
52 <a class="next" onclick="plusSlides(1, 1)"&#10095;</a>
53 </div>
54
55 </body>
56
57 <script src="script.js" type="text/javascript"></script>
58
59
60
61
62
63

```

Dit is de Html code voor de Slideshow Poc. Je zit dat ik een div heb gemaakt met daarin de afbeeldingen. Iedere afbeelding staat nog in een aparte div met de classnaam MySlides1 of MySlides2. Deze zien we later terugkomen in de Javascript. Het is belangrijk dat deze een aparte naam hebben omdat het twee aparte slideshows zijn. We willen niet dat wanneer we op een pijltje klikken er 2 slideshows gaan veranderen.

Verder zie je ook "prev" onclick="plusSlides(-1,0)" en "next" onclick="plusSlides(1,0)". Dit betekent dat de functie genaamd plusSlides word uitgevoerd. Deze zien we in de Javascript dalijk terug. Wanneer je op de next button klikt met (1,0) komt er dus 1 bij door de functie plusSlides. Daliik in de Javascript zien we waar deze bij komt en hoe hiermee de juiste afbeelding vertoond word.

Voor de tweede slideshow gebruiken we (-1, 1) en (1, 1). Ik beide gevallen zal het eerste getal steeds veranderen om een andere afbeelding aan te geven en het tweede getal hetzelfde blijven.

Scrol verder voor het Javascript gedeelte.

```

1  var slideIndex = [1,1];
2  var slideId = ["mySlides1", "mySlides2"]; /*Hier maak ik variabele voor de slide id's */
3  showSlides(1, 0);
4  showSlides(1, 1); /*Dit zijn functies showslides die uitgevoerd worden, de waardes hierachter worden meegegeven*/
5      // (1,0) is voor de eerste slider en (1,1) is voor de tweede slider, in de html staat <a class="prev" onclick="plusSlides(-1, 0)">&#10094;</a> bij de terugknop
6      // en <a class="next" onclick="plusSlides(1, 0)">&#10095;</a> bij de next knop, deze geven dus de +1 en de -1 om verder te gaan en terug.
7
8      // er staat onclick="plusSlides" dit betekent dus dat de functie hieronder word opgeroepen
9
10 function plusSlides(n, no) { /*Met deze functie kun je heen en terug in de slideshow door n te veranderen, ga je verder dan komt er een bij, ga je terug dan gaat er een vanaf
11     showSlides(slideIndex[no] += n, no);
12 }
13
14 function showSlides(n, no) {
15     var i;
16     var x = document.getElementsByClassName(slideId[no]);
17     if (n > x.length) {slideIndex[no] = 1} //als n groter is dan de lengte van de array word deze weer terug gezet waardoor n nooit groter word dan het aantal foto's
18     if (n < 1) {slideIndex[no] = x.length} //dit doet hetzelfde als de regel hierboven maar dan
19     for (i = 0; i < x.length; i++) { // voor wanneer n kleiner word zodat je ook achteruit kan gaan in de array en dus de foto's
20         x[i].style.display = "none"; //Dit zorgt ervoor dat er met een loop alle foto's steeds worden verborgen
21     }
22     console.log(x[no]-1)
23     x[slideIndex[no]-1].style.display = "block"; // hier word gecheckt welke waarde n heeft. Als n = 0 word de eerste afbeelding weergegeven, is n = 1 dan de tweede etc.
24 }
25
26

```

Dit is het Javascript gedeelte van mijn code. Je zit dat ik in het begin variabele aanmaak voor SlideIndex en SlideId. Hierna maak ik functies die uitgevoerd worden en geef ik de waardes aan mee die ik net uitlegde bij de Html Code. Deze functies worden later in de code weer gebruikt.

Hierna komt de functie plusSlides die we ook in de Html code zagen. Hierdoor verandert het getal n in (n, no). Dit komt omdat in de Html code ook de (1,0) of (-1,0) meegegeven werd aan de onclick="plusSlides".

n verandert dus als je de next of previous button aanklikt in de slideshow. De volgende code zorgt ervoor dat de juiste afbeelding word weergegeven.

De functie showSlides checkt eerst of n groter is dan de array door if (n > x.length) {slideIndex[no] = 1} en zet deze weer terug als dat zo is. Zo krijg je een herhalende slideshow die zich steeds opnieuw af kan spelen.

De if statement daaronder zorgt voor hetzelfde maar dan de andere richting op door if (n < 1) {slideIndex[no] = x.length}.

De for loop daaronder zorgt steeds dat de waardes van de slides die momenteel niet actief zijn een style.display none krijgen. Je verbergt dus met een loop steeds alle foto's.

Op het einde word de krijgt de actieve slide een display block waardoor deze weergegeven word. Als n verandert gaat het getal bijvoorbeeld van (1,0) naar (2,0) waardoor een andere afbeelding de display block krijgt en word weergegeven.