

Uitleg JavaScript JS + CSS Clock

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <title>JS + CSS Clock</title>
  <link rel="icon" href="https://fav.farm/✓">
  <link rel="stylesheet" href="style.css">
</head>

<body>

  <div class="clock">
    <div class="clock-face">
      <div class="hand hour-hand"></div>
      <div class="hand min-hand"></div>
      <div class="hand second-hand"></div>
    </div>
  </div>

  <div class="stopwatch">
    <p id="stopwatch-display">00:00:00</p>
    <div class="stopwatchButtons">
      <button id="startStopwatch">Start</button>
      <button id="resetStopwatch">Reset</button>
    </div>
  </div>

  <script>

    // Code made for clock with Tutorial

    const secondHand = document.querySelector('.second-hand');
    const minsHand = document.querySelector('.min-hand');
    const hourHand = document.querySelector('.hour-hand');

    function setDate() {
      const now = new Date();

      const seconds = now.getSeconds();
      const secondsDegrees = ((seconds / 60) * 360) + 90;
      secondHand.style.transform = `rotate(${secondsDegrees}deg)`;

      const mins = now.getMinutes();
```

```

    const minsDegrees = ((mins / 60) * 360) + ((seconds / 60) * 6) +
90;
    minsHand.style.transform = `rotate(${minsDegrees}deg)`;

    const hour = now.getHours();
    const hourDegrees = ((hour / 12) * 360) + ((mins / 60) * 30) +
90;
    hourHand.style.transform = `rotate(${hourDegrees}deg)`;
  }

  setInterval(setDate, 1000);

  setDate();
  // ---

  const stopwatchDisplay = document.getElementById('stopwatch-
display');
  const startStopwatchBtn =
document.getElementById('startStopwatch');
  const resetStopwatchBtn =
document.getElementById('resetStopwatch');

  let stopwatchInterval;
  let stopwatchSeconds = 0;

  function updateStopwatch() {
    stopwatchSeconds++;
    const hours = Math.floor(stopwatchSeconds / 3600);
    const minutes = Math.floor((stopwatchSeconds % 3600) / 60);
    const seconds = stopwatchSeconds % 60;

    stopwatchDisplay.textContent = `${formatTime(hours)}:$
{formatTime(minutes)}:${formatTime(seconds)}`;
  }

  function startStopwatch() {
    startStopwatchBtn.textContent = 'Stop';
    startStopwatchBtn.removeEventListener('click', startStopwatch);
    startStopwatchBtn.addEventListener('click', stopStopwatch);
    stopwatchInterval = setInterval(updateStopwatch, 1000);
  }

  function stopStopwatch() {
    startStopwatchBtn.textContent = 'Start';
    startStopwatchBtn.removeEventListener('click', stopStopwatch);
    startStopwatchBtn.addEventListener('click', startStopwatch);
    clearInterval(stopwatchInterval);
  }

```

```

function resetStopwatch() {
    stopStopwatch();
    stopwatchSeconds = 0;
    stopwatchDisplay.textContent = '00:00:00';
}

function formatTime(time) {
    return time < 10 ? `0${time}` : time;
}

startStopwatchBtn.addEventListener('click', startStopwatch);
resetStopwatchBtn.addEventListener('click', resetStopwatch);
</script>
</body>

</html>

```

Hieronder staat kort uitgelegd hoe de JS+CSS Clock is gebouwd:

In de setDate() functie wordt de code gebruikt die ik met behulp van een tutorial heb geschreven om de klokwijzers bij te werken op basis van de huidige tijd. Hier is een overzicht van wat er gebeurt in de functie:

Allereerst worden de elementen van de klokwijzers (seconden, minuten en uren) geselecteerd met behulp van de document.querySelector() methode. Deze elementen worden opgeslagen in de variabelen secondHand, minsHand en hourHand.

Daarna wordt de new Date() constructor gebruikt om de huidige tijd en datum te verkrijgen. De tijdwaarden (seconden, minuten en uren) worden geëxtraheerd uit het verkregen tijdstempel met behulp van de methoden getSeconds(), getMinutes() en getHours() van het Date object.

Op basis van deze tijdwaarden worden de rotatiehoeken berekend voor elke wijzer. Bijvoorbeeld, voor de secondenwijzer (secondHand), wordt de rotatiehoek berekend door de verhouding van de huidige seconden tot 60 (het aantal seconden in een minuut) te nemen en te vermenigvuldigen met 360 graden. Er wordt ook een verschuiving van 90 graden toegevoegd om de wijzers correct uit te lijnen met het 12-uurs markeerpunt.

De transform eigenschap van de klokwijzers wordt ingesteld met behulp van de berekende rotatiehoeken, waardoor de wijzers visueel worden bijgewerkt op de webpagina.

Om ervoor te zorgen dat de klokwijzers continu worden bijgewerkt, wordt de setDate() functie elke seconde opnieuw aangeroepen met behulp van de setInterval() functie. Dit zorgt voor een vloeiende beweging van de wijzers en houdt de tijd actueel.

Als extra heb ik een stopwatch-functionaliteit toegevoegd aan de webpagina. Onder de klok wordt een stopwatch weergegeven, inclusief een weergavegebied voor de verstreken tijd, een "Start" knop om de

stopwatch te starten of te pauzeren, en een "Reset" knop om de stopwatch terug te zetten naar nul.