# OOP Eindopdracht

In de OOP Challenge moet je een formulier maken en aan de hand van de waarden die je invult moet je dan een berekening maken. Deze berekening laat dan zien welk vervoermiddel sneller is; de fiets of de scooter. Op deze pagina wordt in een stappenplan uitgelegd hoe je dit kunt aanpakken.

Een voorbeeld van hoe de applicatie er zou kunnen uitzien zie je: hier.

image-1676903058536.png

# Stap 1

Maak in HTML een web form waarin alle invoer velden worden weergegeven. Dit zijn de velden zoals je ze in de voorbeeldapplicatie ziet, dus lengte van de weg, vertraging door file, soort fiets, bandenspanning fiets, soort scooter en bandenspanning scooter.

## Bronnen om te raadplegen

Form (algemeen): https://www.w3schools.com/html/html_forms.asp

Radio buttons (type fiets en scooter): https://www.w3schools.com/tags/att_input_type_radio.asp

In het form geeft je ook aan hoe je het formulier wilt posten, via GET of POST. Weet je niet wat te kiezen, neem dan GET.

# Stap 2

Zorg ervoor dat je het formulier post en dat je op een nieuwe pagina (de pagina waarna je het formulier stuurt), alle variabelen met hun waarden afdrukt.

Je output ziet er dan bijvoorbeeld zo uit:

image-1676903949706.png

# Stap 3:

Druk nu ook de snelheid van het gekozen voertuig af:

image-1676904185300.png

# Stap 4:

Je ziet nu in het bovenstaande voorbeeld dat de scooter 25 km/uur rijdt. Hoe lang doet hij dan over 50 km?

Elk uur rijdt de scooter 25 km, dus hij doet 2 uur over 50 km.

Dit kan je ook in een formule zetten.

```
reistijd = weglengte / standaardSnelheidScooter
```

Als de bandenspanning 50 (%) bedraagt dan doet de scooter er 2x zo lang over, dus in het voorbeeld hierboven doet hij er geen 2 uur maar 4 uur over. We zetten dit ook in de berekening.

```
reistijd = weglengte / standaardSnelheidScooter
reistijd = reistijd * bandenspannignScooter/100
```

Op regel 2 wordt de (nieuwe) reistijd dus de (oude) reistijd x bandenspanning/100.

Bereken de reistijd voor de scooter én voor de fiets. Gebruik de bovenstaande formules. Druk de waardes af en controleer of alles klopt.

# Stap 5

Je hebt nu de reistijd in uren. Elk uur heeft 60 minuten. Vermenigvuldig de reistijd met 60 zodat je de reistijd in minuten krijgt.

Druk de reistijden van de scooter en fiets nu af in minuten.

Alleen de scooter heeft last van vertraging. Tel het aantal minuten vertraging nu op bij de reistijd van de scooter.

# Stap 6

Je hebt nu twee reistijden. Bepaal welk voertuig het snelste is en geef een reisadvies zoals dat ook in de voorbeeldapplicatie is gedaan.

Je bent nu eigenlijk klaar maar we gaan onze code nu in OOP (objecten) zetten.

# Stap 7, OOP

- Voeg aan het **eind** van je code, code toe die alles in OOP zet.

- Maak de classes en maak (in eerste instantie) public properties in de classes. Elke variabele in je 'oude' code moet ergens in een class als property worden geplaatst.
- Instantieer de objecten en geef alle public properties de juiste waarden. Voor het zetten van de juiste waarden gebruik je de $_GET['key'], values, dat zijn de waarden die uit het form komen.
- Voeg een public property *reistijd* toe in he class.
- Zet een method/functie in de class *fiets* en *scooter* waarin je de reistijd **berekent**. Gebruik hiervoor dezelfde formules die je ook in je 'oude 'code  (uit stap 1 t/m6) hebt gebruikt.
- Roep de methods/functies aan waarmee de berekening wordt uitgevoerd. De property reistijd van scooter en van fiets worden nu door het gebruik van de method/functie in de class op de juiste waarde gezet.

  Druk deze properties af en vergelijk of de waarden kloppen aan de hand van je 'oude' code van stap 1 t/m 6
- Deze reistijden (properties) kun je nu gebruiken om een reisadvies te geven. Werkt alles dan haal je al je code die je in stap 1 tot en met stap 6 hebt gemaakt weg.

# Stap 8 (optioneel)

Je kunt nu de properties van je classes private maken en voor het zetten en uitlezen van de waarden *setters* en *getters* gebruiken.

# Stap 9, vormgeving, testen en commentaar

Maak alles af. Zorge dat alles er netjes uitziet. Let op de **vormgeving** (CSS) en **test** alles goed. Plaats nog wat **commentaar** in je code zodat anderen (en jij later) ook begrijpen wat je doet.

--

roc.ovh/link/655

Revision #3
Created 2023-02-20 14:20:17 UTC by Max
Updated 2023-02-20 15:12:09 UTC by Admin