

# Opdracht Klassen-Objecten-2

## Voetbalscores

---

### Inleiding

Het doel van deze opdracht is om een applicatie te schrijven die de resultaten van een voetbalteam inleest en op het scherm afdrukt. De resultaten bestaan uit een aantal wedstrijden. Dit zijn wedstrijden die door hetzelfde team zijn gespeeld tegen verschillende tegenstanders, samen met de uitslag en het totaal aantal gescoorde punten. Bij het vak Programming zou je voor alle gegevens van een wedstrijd apart een array aanmaken. Bij Object Oriented Programming maak je van een wedstrijd een class en verwerk je gegevens van een wedstrijd in een object.

model::Wedstrijd
+naamTegenstander: String +voorDoelpunten: int +tegenDoelpunten: int
+Wedstrijd(naamTegenstander: String, voorDoelpunten: int, tegenDoelpunten: int) +Wedstrijd(naamTegenstander: String) +Wedstrijd() +bepaalWedstrijdPunten(): int +printUitslag() +printWedstrijdPunten()

### Stappenplan

- Maak een project aan genaamd 'Voetbalscores'.
  - Maak twee packages aan genaamd 'controller' en 'model'.
  - Maak in de package 'controller' een klasse aan genaamd 'VoetbalScoresLauncher' en zorg dat deze class main methode bevat.
  - Maak in de package 'model' een klasse aan genaamd 'Wedstrijd'.
- De klasse Wedstrijd moet voldoen aan bovenstaand UML diagram.
  - Maak een default constructor Wedstrijd(), die de naam op 'Onbekend' zet en de doelpunten voor en tegen beide op 0.
  - Maak een constructor Wedstrijd(String naamTegenstander) die de meegegeven naam kopieert in het bijbehorende attribuut, en de voor- en tegenDoelpunten beide op 0 zet.
  - Maak een all-args constructor Wedstrijd(String naamTegenstander, int voorDoelpunten, int tegenDoelpunten) die de meegegeven naam en getallen kopieert in de bijbehorende attributen.
  - Maak een methode bepaalWedstrijdPunten die op basis van de doelpunten teruggeeft of er 3, 1 of 0 punten is verdiend (bij resp. winst, gelijkspel of verlies).
  - Maak een methode printUitslag() die op basis van de waarde van de attributen de uitslag print: Uitslag van de wedstrijd tegen PEC: 5-0
  - Maak een methode printWedstrijdPunten() die gebruikt maakt van de methode bepaalWedstrijdPunten() en uitprint: Dit levert 3 wedstrijdpunten op.

3. Test de werking van je klasse in de main methode van de klasse VoetbalScoresLauncher.
  - a. Maak een object aan van het type Wedstrijd door de constructor aan te roepen:  
`Wedstrijd eersteWedstrijd = new Wedstrijd("PEC", 5, 0);`
  - b. Roep de `printUitslag()` en de `printWedstrijdPunten()` methodes aan op dit object.
4. Test de werking ook door informatie van de gebruiker te verwerken:
  - a. Vraag de gebruiker om de naam van een tegenstander.
  - b. Maak een tweede object aan van het type Wedstrijd waarbij je in de constructor alleen de naam van de tegenstander meegeeft.
  - c. Vraag de gebruiker achtereenvolgens om de doelpunten voor en de doelpunten tegen en zorg dat de opgegeven waardes toegekend worden aan de bijbehorende attributen van het tweede object. Bedenk zelf hoe dit moet. Zorg voor een gelijkspel.
  - d. Roep de `printUitslag()` en de `printWedstrijdPunten()` methodes aan voor dit tweede object.

## Output voorbeeld

Voor de duidelijkheid is input die de gebruiker invoert *schuin en onderstreept*.

```
Uitslag van de wedstrijd tegen PEC: 5-0
Dit levert 3 wedstrijdpunten op

Wat is de tegenstander? PSV
Doelpunten gescoord: 2
Doelpunten tegenstander: 2
Uitslag van de wedstrijd tegen PSV: 2-2
Dit levert 1 wedstrijdpunt op
```

## Richtlijnen bij coderen (zie ook HBO-ICT code conventions [ICC])

- Zorg dat je naam en het doel van het programma bovenin staan (ICC #1).
- Gebruik de juiste inspringing (indentation) bij de lay-out (ICC #2).
- Let op juist gebruik hoofdletters en kleine letters (ICC #3).
- Gebruik goede namen (ICC #4).
- Voeg waar nodig commentaar toe die inzicht geven in je code (ICC#7).