



Vier op een Rij

FHICT

1.1 Voorkomende leerdoelen

file handling, exception, private/public, casting, list en foreach, UI separation.

Zie Chapter ?? voor meer informatie over deze leerdoelen.

1.2 Vereiste voorkennis

OIS11.

1.3 Vier op een rij

Maak het spel 4 op een rij waarbij je tegen de computer kunt spelen, maar houdt wel rekening met onderstaande eisen: 1. Programmeer klasse Zet met twee integer property's Rij en Kolom (het property Rij is read-only, deze wordt uitgerekend omdat de stenen naar beneden vallen). De constructor accepteert een referentie naar klasse Spel (zie hieronder) en een integer-kolom.

2. Een enum Veld met mogelijke waarden Rood en Geel.

```
[ enum Veld {Rood,Geel}
```

3. Klasse Spel met intern een private array van 6 bij 7 (tweedimensionaal array)

```
[ Veld[] bord;
  bord = new Veld[6,7];
```

4. Het is een console-applicatie (dus geen Forms gebruiken). Het speelbord hoeft niet te worden afgedrukt. Mag wel, maar dan met `Console.WriteLine` (loop dan door het array heen en druk het stap voor stap af)

5. De klasse `Spel` heeft methoden als `BedenkEenZet()` om de beste zet voor de computer te bedenken en `AccepteerEenZet(Zet z)` (om een zet van de gebruiker te accepteren).

■ Note

De methode `BedenkEenZet()` kan in eerste instantie op zoek gaan naar de eerste de beste vrije kolom. Indien er geen vrije kolom meer is dan kan de methode `GelijkSpel()` worden aangeroepen die het spel stopt. Maak private hulpmethoden zoals `bool HeeftSpelerGewonnen()` en `bool HeeftComputerGewonnen()` die je zelf aanroept.

```
[ private Random Randje;

  public Zet BedenkEenZet()
  {
    return new Zet(Randje.Next(7));
  }
```