



Programmeren1 Huiswerk

Arrays

Week 3

Arrays... toffe dingen niet? Handig ook, zo'n verzameling van getallen, Strings of andere objecten, maar kunnen we er ook iets écht tofs mee maken? Ja maar natuurlijk! Deze week ga je thuis aan de slag met het maken van een leuke animatie.

Bijspijkeren?

Zorg dat je bij blijft! Als je met deze oefening aan de gang gaat en het is al week 4 of later, dan moet je twee dingen doen:

- Beter gaan plannen. Hoe komt het dat je achter loopt? Hoe kan je dat voorkomen?
- Zorg dat je alle stof van week 1 tot en met 3 begrijpt. Stap daar niet te makkelijk overheen.



Wuppy Wave!



"Java arrays"

OK. Deze keer gaan we een verzameling wuppies laten bewegen. Hiervoor open je in Eclipse de code voor deze week. Die vind je in "HW_Arrays.zip".

Bestudeer eerst de gegeven code. Als je de code van de *klasse (class)* `WuppyWaver` doorkijkt, zie je een aantal regels waar je nog niets van snapt. Dat is geen probleem, dat komt later wel. Richt je nu even op de twee stukken tussen `YOUR CODE - START` en `YOUR CODE - END`, daar ga jij jouw code toevoegen. Er zijn twee van deze stukken:

Instance variables gedeelte:

Het eerste stuk kan je gebruiken voor *instantievariabelen* (instance variables). Zeker voor opdracht 4 en 5 zal je die als het goed is nodig hebben.

Gedeelte met `onInitialize()` en `performStep()`:

In dit tweede stuk kan je straks de twee genoemde methodes gebruiken om:

- `onInitialize()` De te gebruiken array's te *initialiseren*.
- `onInitialize()` Initiële waarden te zetten.
- `performStep()` De waarden elk stapje iets aan te passen.

Lees het commentaar bij deze twee methodes goed, zodat je begrijpt waar ze toe dienen.

Je ziet dat deze klasse (class) onder andere twee instantievariabelen heeft waar je aan mag komen*:

```
private int[] m_XPositions
```

```
private int[] m_YPositions
```

Deze twee arrays ga jij gebruiken om de X en Y coördinaten van alle wuppies in op te slaan:

Het eerste element van `m_XPositions` bevat de X coördinaat van de eerste wuppie, etc.

Het eerste element van `m_YPositions` bevat de Y coördinaat van de eerste wuppie, etc.

* Zoals je gewend bent zijn alle methodes / variabelen die beginnen met "aii" niet voor jou bedoeld.

[1] → Breid het programma uit, zodat je een rij met 10 wuppies te zien krijgt. Zorg er voor dat jouw uitvoer zo goed mogelijk op plaatje hier rechts lijkt.

Hiervoor moet je dus in elk geval het volgende doen:

- De twee arrays vullen (in een loop?).

De `performStep()` methode hoef je nu dus nog niet te gebruiken.

Tip: In de console zie je waar elk wuppie voor het eerst wordt getekend. Als je dus niets op het scherm ziet terwijl je het wel verwacht, kijk dan daar wat de X en Y waardes zijn.



Nu ga je ook de `performStep()` methode gebruiken. Deze methode wordt automatisch 60 keer per seconde aangeroepen. Als je hier dus elke keer de Y positie van een wuppie iets veranderd, zal het er bij uitvoer op lijken alsof de wuppie beweegt!

[2] → Zorg er voor dat de rij met wuppies na start automatisch omhoog(!) schuift. De wuppies gaan helemaal uit beeld aan de bovenkant (en komen nooit meer terug... snif).

Zoals je weet werken we bij **arrays** met indexes. Het eerste element heeft index 0, het tweede heeft index 1, etc. Voor het gemak noemen we de eerste (linker) wuppie nu "wuppie 0", de tweede noemen we "wuppie 1", etc.

[3] → Zorg er nu voor, dat hoe hoger de index van een wuppie is, hoe sneller hij naar boven schiet na de start van het programma. Na ongeveer 1 seconde moet jouw scherm eruit zien als de screenshot hier naast:

Zoals je ziet, is na het verstrijken van 1 seconde de rechter (laatste) wuppie al heel ver en de eerste nog niet. Elke volgende wuppie is steeds een stukje sneller (zoals gevraagd in opdracht 3).

[4] → Alle wuppies gaan weer even snel omhoog (zoals in opdracht 2), maar deze keer stuiten ze terug als ze de bovenkant raken. Als hun Y positie verraadt dat ze buiten beeld zijn, zorg jij er voor dat ze weer naar beneden gaan. Als ze de onderkant raken gaan ze weer omhoog. (Hoor ik hier een **if**-statement?)



De wuppies blijven nu dus altijd in beeld en zijn continu in beweging!

Tip: Het scherm is 600 bij 600 pixels.

[5] → Ten slotte voegen we alles samen: De wuppies bewegen zoals in opdracht 3 (hoe hoger hun index, hoe sneller ze gaan), maar stuiteren terug op de rand van het scherm.

Als je na ongeveer 10 seconden een screenshot maakt, zou die er zo uit kunnen zien:

Volgende week wordt de uitwerking behandeld in het practicum en / of hoorcollege. Zorg dat je dan al je vragen klaar hebt liggen!

En dan nog dit

Natuurlijk sta jij te popelen om ook het boek nog helemaal door te werken (als je dat nog niet gedaan hebt). Daar krijg je dan nu de kans voor!

[1] → Lees pagina 16 t/m 26 door en maak alle opgaven die je tegenkomt (en nog niet gemaakt had).

[2] → Zorg dat je de practicumopgaven van deze week helemaal hebt doorgewerkt.



Java™