

Object Oriented Programming

(OOP / Objectgeoriënteerd programmeren)

Doelen

- Je weet wat Object Oriented Programming (OOP) is
- Je weet wat classes en objecten zijn
- Je kunt zelf een simpele class schrijven

Tot nu toe

- Variabelen
- Datatypen (String, Integer, Boolean, Array, ...)
- Loops (for, foreach, while)
- Functions

Procedural Programming

Procedural Programming

- Simpel
- Code wordt van boven naar beneden uitgevoerd
- Je kunt er alles en elke applicatie mee maken

Waarom dan OOP?


```
<?php
$car_1 = 'Ford Mustang GT';
$pos_car_1 = 0;
$finish = 25;

while ($pos_car_1 != $finish) {
    $pos_car_1++;
}

echo $car_1 . " made it!";
```



```
<?php
$car_1 = 'Ford Mustang GT';
$pos_car_1 = 0;
$finish = 25;

while ($pos_car_1 != $finish) {
    $pos_car_1++;
}

echo $car_1 . " made it!";

$car_2 = 'Aston Martin DB11';
$pos_car_2 = 0;
$finish = 25;

while ($pos_car_2 != $finish) {
    $pos_car_2++;
}

echo $car_2 . " made it!";
```



```

<?php
$car_1 = 'Ford Mustang GT';
$pos_car_1 = 0;
$finish = 25;

while ($pos_car_1 != $finish) {
    $pos_car_1++;
}

echo $car_1 . " made it!";

$car_2 = 'Aston Martin DB11';
$pos_car_2 = 0;
$finish = 25;

while ($pos_car_2 != $finish) {
    $pos_car_2++;
}

echo $car_2 . " made it!";

```

Of toch liever:

```

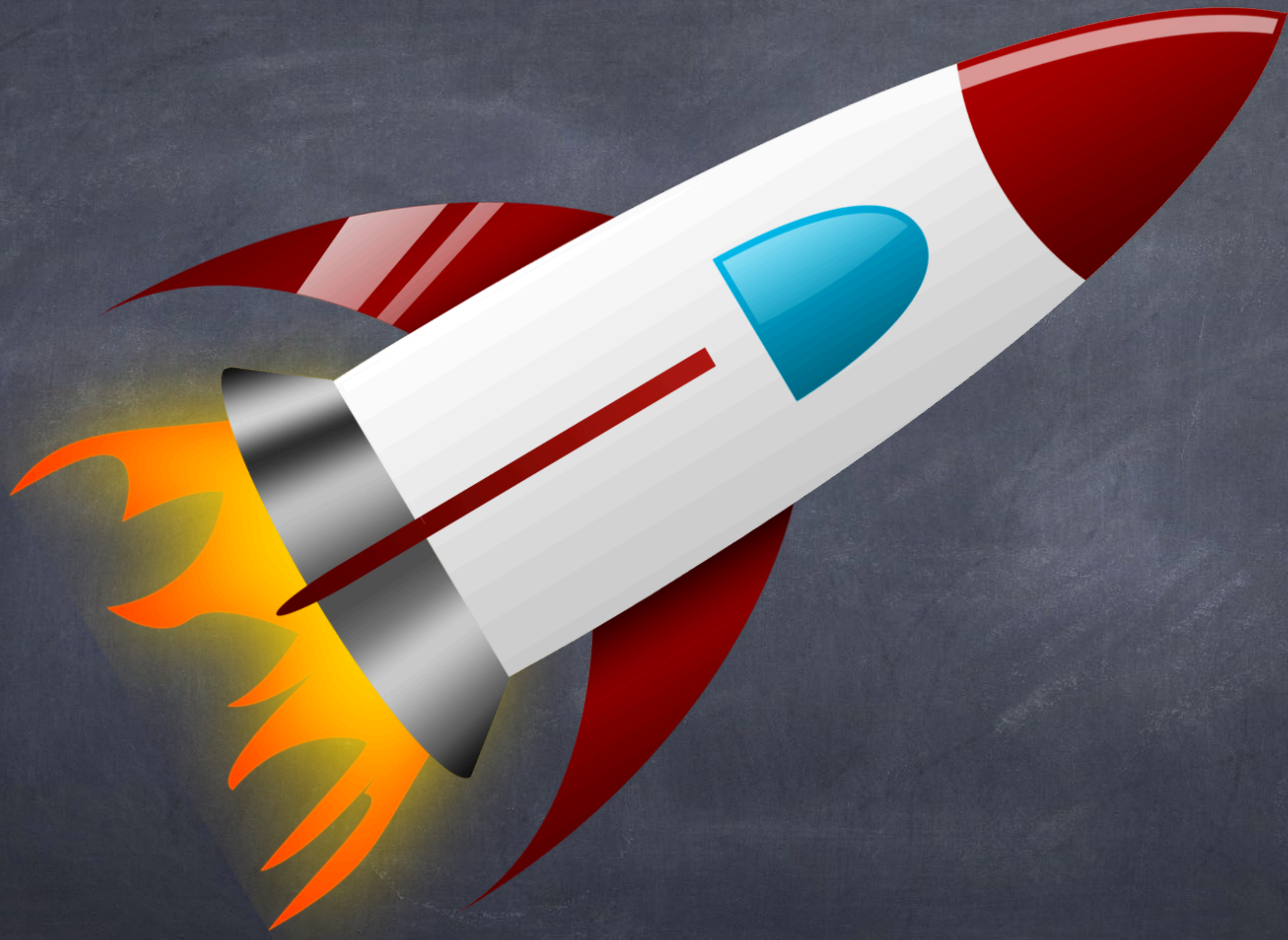
<?php
require_once( 'Car.php' );

$start = 0;
$finish = 25;

$car1 = new Car();
$car1->setName( 'Ford Mustang GT' );
$car1->driveTo($finish);

$car2 = new Car();
$car2->setName( 'Aston Martin DB11' );
$car2->driveTo($finish);

```

https://www.youtube.com/watch?v=sB_nEtZxPog&t=114s

Opdracht

<https://git.newdeveloper.nl/jkossen/app17-2017-2018>

Leesvoer: leesvoer/oop_les1.adoc (hier staat de opdracht ook in)