

## Opdracht 5b-1) Voeg een keuze toe

Deze les wordt afgetekend als:

- Je een nieuw programma hebt gemaakt ('les-5b')
- Je in dat programma minstens één keuze verwerkt.

### Opdracht.

Open een nieuw programma.

Zorg er nu voor de de lezer van je verhaal minstens één keer kan kiezen. Bijvoorbeeld kiezen uit twee dieren.

```
if input('Wil je een verhaal over een hond of een kat?') == 'hond':  
    print('Dit verhaal gaat over een kat.')  
else:  
    print('Dit verhaal gaat over een hond.')
```

Tips voor de if-else

- Denk aan de dubbele punten
- Denk aan de dubbele =
- Denk aan de spaties

## Opdracht 5b-2) if-else commandos en gewone prints

Deze les wordt afgetekend als:

- Je een verhaal maakt waarin if-else in zit
- er in je verhaal ook minstens één print staat die **niet** in een if-else zit

### Opdracht.

Zorg er nu voor dat je verhaal na de if-else verder gaat met een paar gewone print() regels.

Let goed op dat die regels nu niet met twee spaties beginnen.

## Opdracht 5b-3) if-else commandos en een variabele

Deze les wordt afgetekend als:

- Je een verhaal maakt waarin **twee** keer if-else in zit
- er in je verhaal ook minstens één print staat die **niet** in een if-else zit
- je ook een variabele gebruikt, in een if-else

### Opdracht.

Voeg een nieuwe if-else toe. Zet nu in de takken van de if-else een variabele. Bijvoorbeeld:

```
#hierboven staat dus al wat code van eerdere opdrachten!!!
```

```
#een print met een variabele
```

```
if input('hond of kat?') == 'hond':
    dier = 'hond'
else:
    dier = 'kat'
```

## Opdracht 5b-4) if-else commandos en een lijst

Deze les wordt afgetekend als:

- Je een verhaal maakt waarin **drie** keer if-else in zit
- je ook een lijst gebruikt, in een if-else
- je ook iets uit de lijst aanwijst, buiten een if-else

### Opdracht.

Voeg een nieuwe if-else toe. Zet nu in de takken van de if-else nu een lijst. Bijvoorbeeld:

```
#hierboven staat dus al wat code van eerdere opdrachten!!!
```

```
#een print met een lijst
```

```
if input('repielen of zoogdieren?') == 'reptielen':
    vrienden = ['hagedis', 'schildpad', 'kameleon']
else:
    vrienden = ['kat', 'hond', 'koe']
```

```
print('De', 'vriend', 'van', dier, 'is', vrienden[1], '.')
```

## Opdracht 5b-5) Sla de ingevoerde tekst ook op (extra)

Deze les wordt afgetekend als:

- Je de invoer van een gebruiker opslaat in een variabele
- Je die variabele gebruikt in je verhaal

Let op! In deze opdracht komt nieuwe code voor die nog niet is uitgelegd!

Je kunt de invoer van een gebruiker ook opslaan in een variable. Je zet de variabele dan voor de input(), zo:

```
dier = input('Over welk dier gaat dit verhaal?')  
print('Dit', 'verhaal', 'gaat', 'over', dier)
```

Zo kan de gebruiker ieder dier invoeren. Je kunt de input()s zelf aan elkaar koppelen, zo:

```
dier = input('Over welk dier gaat dit verhaal?')  
print('Dit', 'verhaal', 'gaat', 'over', dier)
```

```
kleur = input('Welke kleur heeft' dier, '?')  
print('Het', dier, 'is', kleur)
```

## Opdracht 5b-6) Voeg een shuffle toe (extra)

Deze les wordt afgetekend als:

- Je bovenaan je code hebt geschreven `import random`
- Je een zin in je code hebt waarin steeds een ander dier verschijnt
- Je in je schrift schrijft hoe deze code werkt

### Opdracht.

Lees deze opdracht goed! Er komt nieuwe code in voor die nog niet is uitgelegd in de klas.

Je hebt een lijst met vijf dieren erin gemaakt. Maar je krijgt nu steeds dezelfde te zien in je verhaal. We gaan dat nu veranderen, zodat je steeds een ander dieren in je verhaal krijgt.

Random is Engels voor willekeurig. Met random-opdrachten kun je Python steeds iets anders laten doen. Shuffle betekent schudden. De shuffle()-opdracht schudt de dieren door elkaar.

Zet deze code onder `vrienden =`:

```
random.shuffle(vrienden)
```

Voer je code een paar keer uit. Wat gebeurt er nu?

Schrijf duidelijk in je schrift hoe je denkt dat deze code werkt.

Einde werkblad! Hebben we alles afgestempeld? Dan mag je meehelpen met stempelen.