

for en if en break

Aan het einde van dit werkblad kun jij:

- voorspellen wat Python code met een `for` en een `if` en een `break` erin doet
- goede en foute for lussen van elkaar onderscheiden

Spelregels:

- Denk niet te diep na over de codes, het gaat er om dat je het snel gaat kunnen
- Schrijf de codes over als daarom gevraagd wordt, juist door opschrijven oefen je goed
- Spieken/overschrijven is zinloos, het gaat juist om het oefenen. Snap je iets niet, vraag het dan aan ons

for en if uitleg

Je weet al dat je met een `for` code kunt herhalen voor ieder element uit een lijst. Maar je kunt de `for` ook combineren met een `if`. Dat is handig als je data wilt *filteren*, maw als je alleen bepaalde elementen uit een lijst wilt selecteren:

```
bingo_getallen = [2,4,19,25,26,47]
```

```
for getal in bingo_getallen:  
    if is_even(getal):  
        print(getal)
```

Nu worden alleen de even getallen geprint:

```
2  
4  
26
```

Let op deze dingen:

- je springt nu 2 keer in, een keer voor de `for` en een keer voor de `if`
- alle regels die met 2 spaties beginnen horen bij de lus
- alle regels die met 4 spaties beginnen horen bij de if

Opdrachten

1. Je krijgt een aantal codes. Wat wordt er geprint?

Voorbeeld:

```
snacks = ['chips', 'taco', 'drop', 'toffee']
for snack in snacks:
    if snack[1] == 't': #met [] kun je ook 1 letter uit tekst aanwijzen
        print(snack)
```

Voor iedere snack `snack` in de lijst `snacks` kijkt de code of de eerste letter de t is. Dus de code print:

```
taco
toffee
```

Nu jij! Let op, de codes kunnen ook fout zijn. Schrijf dan 'fout'.

```
fruit = ['appel', 'druif', 'banaan']
for f in fruit:
    if 'a' in f:
        print('Ik hou van', f)
```

```
even_getallen = [2,4,6,8,10,12]
for getal in even_getallen:
    if getal > 10:
        print(getal * getal) #een sterretje is keer in Python
```

```
even_getallen = [2,4,6,8,10,12]
for getal in even_getallen:
    if getal == 12:
        print(getal + 5)
```

```
oneven_getallen = [1,3,5,7,9,11,13,15]
for getal in oneven_getallen:
    if getal % 5 == 0: ## betekent 'is deelbaar door'
```

```
print(getal)
```

```
even_getallen = [2,4,6,8,10,12,14,16,18,20,22,24,26,28,30]
for getal in even_getallen:
    if getal > 10 and getal % 4:
        print(getal)
```

Je hebt eerder een accumulator gezien. Die kan natuurlijk ook samen met een `if`:

```
som_boven_10 = 0
metingen = [2,4,14,25,26,42]
for huidige_meting in metingen:
    if huidige_meting > som_boven_10:
        som_boven_10 = som_boven_10 + huidige_meting
print(som_boven_10)
```

break uitleg

Je kunt een `for` ook stopzetten. Dat is bijv. handig als je naar een bepaalde waarde aan het zoeken bent. Heb je de waarde gevonden, dan stop je de `for` met een `break`, zo:

```
even_getallen = [2,4,6,8,10,12,14,16,18,20,22,24,26,28,30]
for getal in even_getallen:
    if getal > 10 and getal % 4:
        print(getal)
        break
```

Nu wordt de lus gestopt zodra een getal is gevonden dat

Extraatje! Deze code kun je ook anders schrijven. Schrijf code op die precies hetzelfde doet met minder regels:

```
bingo_getallen = [2,4,14,25,26,42]
mijn_lot = 17
for getal in bingo_getallen:
    if getal == mijn_lot:
        print('En de winnaar is...' + mijn_lot)
```

