

While Loops

Chelsea Parlett Pelleriti

Review

In if (and if/else), boolean condition is checked once, and then never again.

What if we want to check multiple times?

E.g. is it time to leave yet?

WHILE:

CONDITION



FALSE



EXIT



TRUE

Syntax

while <boolean expression>:

 <do stuff>

 <increment stuff>

What happens if the while is NEVER true?

What happens if the while is ALWAYS true?

Parts of a While Loop

- Initialization: Make your boolean statement true or false
- Control: When should your loop run?
- Modification: Change your boolean!!!

```
x = 0
```

```
while x < 10:
```

```
    print("x is small")
```

```
    x += 1
```

Sentinel Loop

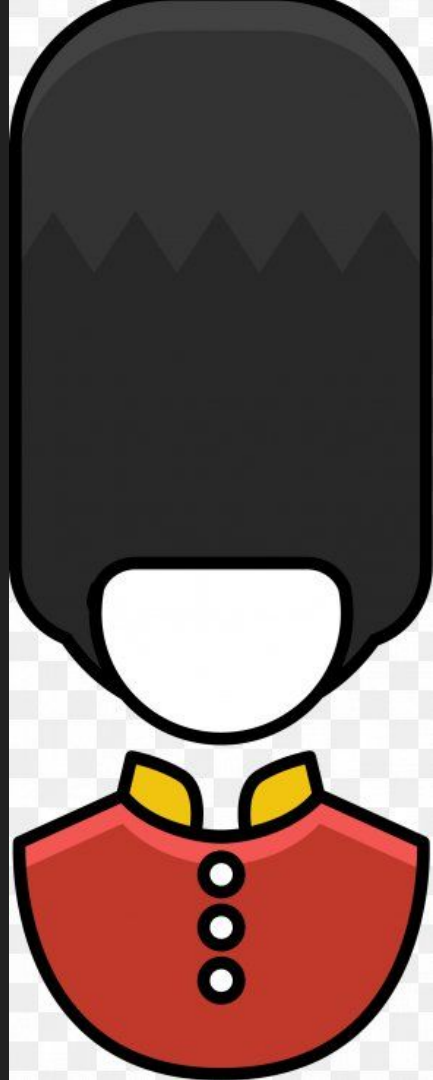
condition_UNsatisfied = True # sentinel

while condition_UNsatisfied:

<do things>

if <condition is actually satisfied>:

condition_UNsatisfied = False



Break and Continue

Break

Break immediately exits the current Loop and does not execute any

other part (include skipping the else part of the loop!)

*use when answer has been found or you no longer need to calculate anything



Continue

Continue is like a less extreme version of break.

It immediately exits from one iteration of the loop.