

# Tutorial 3

## CS 110

# While Loops

Pseudocode

```
while some condition is True:  
    # execute code block
```

# While Loops

1. Used to repeat blocks of code over and over again
2. The block repeats until the termination condition is met
  - a. If the termination condition is never False, you have an infinite loop.
  - b. If the termination condition is never True, your loop will never run.

# Example 1: Infinite Loop

```
while True:  
    print('Hello! How are you doing today?')  
    time.sleep(1)
```

In this case, the loop will never terminate because the expression (i.e. True) will always be true.

Test it: [warmup/a\\_while\\_always\\_true.py](#)

## Example 2: Loop that terminates after 5 iterations

```
counter = 0
while counter < 5:
    print('Hello! How are you doing today?')
    time.sleep(0.5)
    counter += 1
```

In this case, the loop terminates after 5 iterations

Test it: `warmup/b_while_termination_condition.py`

## Example 2: Algorithm

1. Initialize a counter outside of the loop. Example:

```
counter = 0
```

2. Make sure the termination condition relies on the counter. Example:

```
while counter < 5:
```

3. Make sure the counter gets incremented inside of the loop. Example:

```
    counter += 1
```

# Python keywords that are important for loops

- **break**  
Statement that tells the Python interpreter to immediately break out of the loop
- **continue**  
Statement that tells the Python interpreter to jump back to the top of the loop (and not execute any of the statements below it).

**Your Turn!**



# Activity 1: Numbers Game

Finish writing the **01\_number\_game.py** program so that it works as follows:

1. If the user's guess is too low, tell the user the number is too low and to guess again
2. If the user's guess is too high, it tells the user the number is too high and that they should guess again
3. If they guess the number correctly:
  - a. Tell them that they guess correctly and the number of guesses it took to guess correctly
  - b. Exit the program (i.e. break out of the loop)
  - c. If they type the letter 'Q', exit the program (i.e. break out of the loop)

## Activity 2: Drawing a Vertical Line of Circles

1. Open **02\_vertical\_circles.py**
2. See if you can use a while loop to recreate this functionality, where there is only **one** *make\_circle* function call that is repeated within a while loop.

### Hints:

- you will need to initialize a counter
- You will need to make use of the counter to position the y-coordinate of the circle

# Optional: When you're done, try to make these drawings...

