

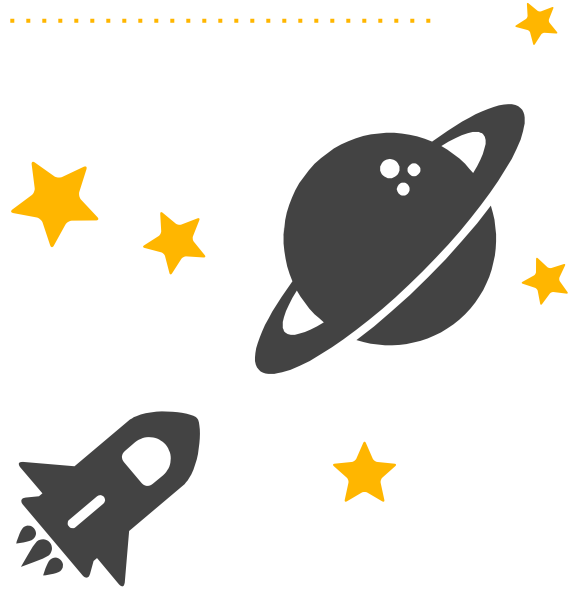


Chatbots with Personality

Loops

For doing things over, and over, and over...!

<http://interactivepython.org/runestone/static/thinkcspy/PythonTurtle/TheforLoop.html>



Loop practice

One big usage of loops is to avoid code duplication (i.e., avoid copy/paste)

```
# Bubble Tea Loop Examples
# Author: Angelica Lim
# Date: Jan. 22, 2018

# ---Here is the most basic way to print a menu. ---
print("***** Here is our menu using the old method.")
print("Mango milk")
print("~~~~~")
print("Taro milk")
print("~~~~~")
print("Matcha milk")
print("~~~~~")
print("Oreo milk")
print("~~~~~")
print("Papaya milk")
print("~~~~~")
print("Chocolate milk")
print("~~~~~")
```

We are duplicating `print(~~)` and `milk` here more than twice!! :(
If there were 20 items on the menu, we'd duplicate that 20x!

I like concise code!



Code Reviewer

Is the code **uplicated** more than twice?

Let's Code



More on Loops

```
# ---Here is a BETTER way that AVOIDS code duplication---
print("***** Here is our menu using a loop.")

# Make a list of flavours
flavours = ["Mango", "Taro", "Matcha", "Oreo", "Papaya", "Chocolate"]

# Go through each of the flavours and print it
for flavour in flavours:
    print(flavour + " milk")
    print("~~~~~")
```

Same output, shorter code.

On every loop **iteration**, the variable **flavour** takes the **value** of the items in **flavours**

```
***** Here is our menu using a loop.
Mango milk
-----
Taro milk
-----
Matcha milk
-----
Oreo milk
-----
Papaya milk
-----
Chocolate milk
-----
```

How to **repeat** code

Use a **loop**:

1. Write out one iteration or pass of your code
2. Think about *what would change* between repetitions
 - a. If *nothing* changes, you can use **for i in range(n)** and keep the loop body the same to repeat **n** times.
 - b. If *something would change* between iterations, factor out that element into a variable **i** / **elem** to use either:
 - **for i in range(n)** → integers
 - **for elem in list:** → any data type

Later, we'll also learn how to use functions later this term.

- # 1. Print a random topic
- # 2. Ask P1 for 3 words about topic
- # 3. Ask P2 for 1 word about topic
- # 4. If P2's word was a P1 word,
 - # P2 wins! Otherwise, they lose.
- # 5. Play 3 rounds.



Code Reviewer

Is the code
duplicated more
than twice?

Method 1:

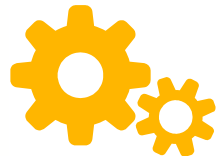
Looping over a list

for

in

This means: **for** every *word* **in** the *words* list, do the following (indented)

```
1 # Mind Reader Game
2 # Angelica Lim
3 # Jan. 24, 2021
4 # This is a 3 round 2-player game. The 1st player reads a word, and secretly
5 # enters 3 words they associate with it. The 2nd player must then try to guess
6 # at least one of the words. If it's a match, they win!
7 import random
8
9 # Introduce the game and create word list
10 print("Welcome to Mind Reader")
11 words = ["cat","snow","hot"]
12
13 # Do 3 rounds
14 for word in words:
15
16     # Ask the first player to enter 3 words associated with a given word
17     print ("Player 1, enter 3 words you think of when I say " + word)
18
19     # Get the 3 words from the user
20     first_word = input("First word: ")
21     second_word = input("Second word: ")
22     third_word = input("Third word: ")
23
24     # Clear the screen
25     print(100*" \n")
26
27     # Ask the 2nd player to guess
28     print("Player 2, what is one word you think Player 1 associates with " +
29         | word + "?")
30     guess = input()
31
32     # Check if they match and tell them if they win!
33     if guess in [first_word, second_word, third_word]:
34         | print("You got it!")
35
36     # Otherwise, if they got it wrong
37     else:
38         | print("No match! They said ",first_word,second_word,
39             | "and",third_word)
```





Range

Generates a list of numbers

<http://interactivepython.org/runestone/static/thinkcspy/PythonTurtle/TheforLoop.html>

```
› list(range(10))  
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]  
› list(range(3,6))  
[3, 4, 5]
```

```

1 # Mind Reader Game
2 # Angelica Lim
3 # Jan. 24, 2021
4 # This is a 3 round 2-player game. The 1st player reads a word, and
5 # secretly enters 3 words they associate with it.
6 # The 2nd player must then try to guess at least one of the words.
7 # If it's a match, they win!
8 import random
9
10 # Introduce the game
11 print("Welcome to Mind Reader")
12
13 words = ["cat", "snow", "hot", "coffee", "Canada", "sport"]
14
15 # Do 3 rounds
16 for i in range(3):
17
18     # Ask the first player to enter 3 words associated with a given word
19     selected_word = random.choice(words)
20     print ("Player 1, enter 3 words you think of when I say " + selected_word)
21
22     # Get the 3 words from the user
23     first_word = input("First word: ")
24     second_word = input("Second word: ")
25     third_word = input("Third word: ")
26
27     # Clear the screen
28     print(100*" \n")
29
30
31 # Ask the 2nd player to guess
32 print("Player 2, what is one word you think Player 1 associates with " +
33     | selected_word + "?")
34 guess = input()
35
36 # Check if they match and tell them if they win!
37 if guess in [first_word, second_word, third_word]:
38     | print("You got it!")
39
40 # Otherwise, if they got it wrong
41 else:
42     | print("No match! They said ", first_word, second_word,
43         | "and", third_word)

```

Remember to change the word variable **inside** the loop, or else it will stay the same for all 3 rounds!

We have found another way to select a word, since we are not looping over a list of strings.

Method 2: Looping with range



Conversion

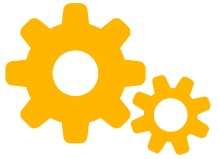
Data type conversion

Type Conversions

- "Hello" is a **string**.
- Convert between types:
 - String to **integer**
`int("42")`
 - String to **float**
`float("3.1415")`
 - Any number to **string**
`str(my_age)`

```
1 # Ask user for their age
2 input_age = input("Age? ")
3
4 # Compute age next year
5 age = int(input_age)
6 age_next_year = age + 1
7 print("Age next year " + str(age_next_year))
8
9 # Use String to float
10 height = float(input("Height? (m) "))
11 twice_height = height * 2
12 print("Twice your height: " + str(twice_height))
```

```
Age? 21
Age next year 22
Height? (m) 4.2
Twice your height: 8.4
```



Simple Converter

```
# Weight Calculator
# Author: Angelica Lim
# Date: Jan 24, 2018

# Convert pounds to kilograms
pounds = float(input("How many pounds?"))

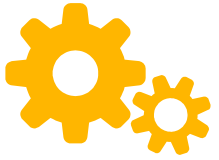
# Output weight in kilograms
kg = pounds*0.45

print("That's " + str(kg) + "kg.")
```

Use `float()` to convert to float type.

<http://interactivepython.org/runestone/static/thinkcspy/SimplePythonData/OperatorsandOperands.html>

<http://interactivepython.org/runestone/static/thinkcspy/SimplePythonData/Typeconversionfunctions.html>



More on data types

```
input ↗ clear ✕  
  
: type(2.5)  
=> <class 'float'>  
: type(0)  
=> <class 'int'>  
: type("hello")  
=> <class 'str'>  
: age = 120  
: type(age/3)  
=> <class 'float'>  
: █
```

Dividing automatically converts integers to float.

Get the type of a variable using:
`type(my_variable)`

<http://interactivepython.org/runestone/static/thinkcspy/SimplePythonData/ValuesandDataTypes.html>