CSc 110
Lists and For Loops

Adriana Picoral
(she/her/hers)

Implement print_card

```
def print_card(content):
    '''
    The function should print out a text card, with
    the provided content as the text on the card.
    For example, if content = "Hi There", then this
    function should print out:
    +------------+
    |  Hi There  |
    +------------+
   '''
```

Implement print_card

```
def print_card(content):
    length = len(content)
    print('+--' + '-' * length + '--+')
    print('|  ' + content + '  |')
    print('+--' + '-' * length + '--+')
```

Implement quiz

```
def quiz(words, definitions):
    '''
    This function should select a random word form the
    list to quiz the user about.
    Once a random index has been generated, use the
    print_card function to display the word, then the
    definition.
    Use the random module!
    '''
```

for loops

- Another type of loop!
- Useful for iterating over or through lists, ranges, and data structures in general

Write a for-loop that produces the same output as the while-loop

```
values = [70, 20, 30, 35, 10, 7, 17, 50]
```

```
i = 0
while i < len(values):
    print(i)
    i += 1
```

Write a for-loop that produces the same output as the while-loop

```
values = [70, 20, 30, 35, 10, 7, 17, 50]
```

```
i = 0
while i < len(values):
    print(values[i])
    i += 1
```

Looping through ***indexes*** compared to Looping through ***values***

```
for i in range(len(values)):
    print(i)


for i in values:
    print(i)
```

Write the code

```
# What code goes here?
# Use a for-loop
```

With this input:

```
ben kai jim jane johanness
```

Produce this output:

```
average name length = 4.4
```

Removing Elements

- **`list.remove(value)`**
  - Removes the first occurrence of **`value`** in the list
  - Raises an exception is the value does not exist in the list
- **`list.pop(index)`**
  - Removes the element at **`index`**
  - Raises exception if no value at **`index`**

Write a Python program to remove all elements from a given list present in another list.

```
def remove_items(original_list, items_to_remove):
    # code goes here

def main():
    list1 = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
    list2 = [2, 4, 6, 8]
    remove_items(list1, list2)
    print(list1)

main()
```