Loops continued
Lecture 15

Week 8

for loop

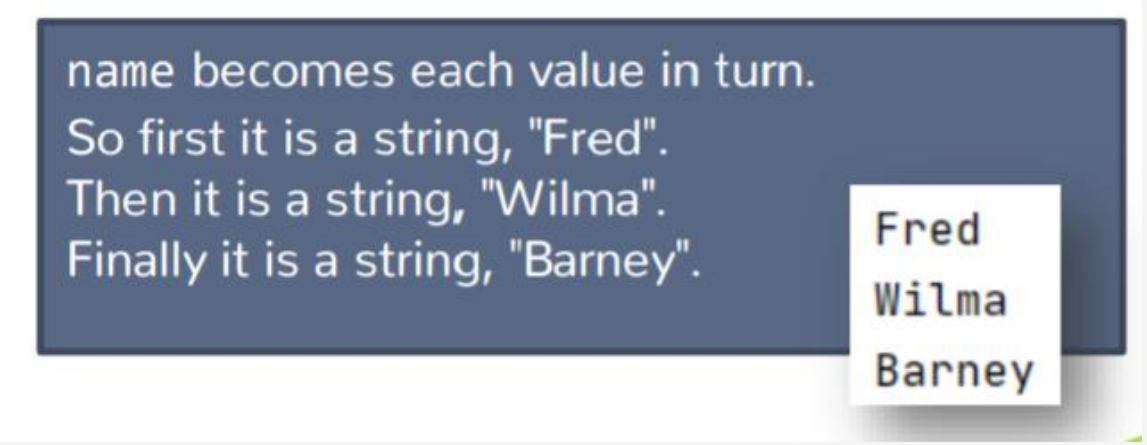
for loops allow us to move through a set of values

The variable in the loop "becomes" each of the values during the iterations of the loop

```
for number in (4, 6, 1, 9):
    print(number, end=' ')
print()
```

for loop

```
for name in ("Fred", "Wilma", "Barney"):
    print(name)
```



for loop – mixed datatypes

```
for item in (4, "Fred", 4.5):
    print(type(item), item)
```

Example: counting vowels

Write code that asks the user for a sentence.

Loop through each character in the input string and

- count the number of vowels that appear
- count the number of spaces “ ”
- count the number of punctuation characters

for looping through a string

A for loop can iterate* through strings

*iterate means to journey through a multi-value object one value at a time

```
sentence = "Once upon a time"
for character in sentence:
    print(character, end="*")
```

Example: counting e's

```
sentence = "Once upon a time"
count_es = 0
for character in sentence:
    if character.lower() == 'e':
        count_es = count_es + 1
print()
print(f"There are {count_es} e's in '{sentence}'")
```

This will hold the number of e's

The character from the sequence is lower cased. If the character is 'e' , add 1 to the count.

Example: counting e's

```
sentence = input("What is the sentence? ")
count_es = 0
for character in sentence:
    if character.lower() == 'e':
        count_es = count_es + 1

if count_es == 1:
    print(f"There is 1 e in '{sentence}'")
else:
    print(f"There are {count_es} e's in '{sentence}'")
```

Getting the grammar right!

range

Range is a Python datatype created from a function

When we create this type we must specify 3 pieces of information: start, stop and step.

```
range(start, stop, step)
```

range

```
for number in range(0,10,1):
    print(number, end=' ')
print()
```

```
0 1 2 3 4 5 6 7 8 9
```

- **range** provides a sequence of integer numbers, starting at 0 and stopping before 10
- **number** becomes each of these values in turn

range

- range is preferred over tuples
- It does not store all the numbers
- Instead it requires only 3 values no matter how large the list of numbers is

```
# 20 values stored
for x in (1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20):
```

```
# 3 values stored
for x in range(1, 21, 1):
```

range(stop)

You must provide the stop value

- range() assumes that the start value is 0
- range() assumes that the step value is 1

```
for number in range(10):
    print(number, end=' ')
print()
```

```
0 1 2 3 4 5 6 7 8 9
```

range(start, stop)

You can provide just the start value and the stop value

- range() assumes that the step value is 1

```
for number in range(3, 10):
    print(number, end=' ')
print()
```

```
3 4 5 6 7 8 9
```

range(start, stop, step)

You can provide just the start value and the stop value, and the step value

```
for number in range(1, 10, 2):
    print(number, end=' ')
print()
```

```
1 3 5 7 9
```

```
for number in range(10, 1, -1):
    print(number, end=' ')
print()
```

```
10 9 8 7 6 5 4 3 2
```

Example: display numbers

Write code to display the numbers between 3 and 99.

Add a second for loop to display the odd numbers between 3 and 99 by changing the step size.