Python 3 reference (advanced)

Repeat a block while a condition is met

Repeat a block 5 times

```
x = 0
while x < 5:
    print("Going round")
    x = x + 1
```

Change the condition inside the loop to make the loop stop.

Repeat a block while a condition is met

```
driver_response = ""

driver_response = input("Are we there yet ? ")
while driver_response != "yes":
    driver_response = input("Are we there yet ? ")

print("Hooray ! Finally !")
```

Lists (variables that hold multiple items)

Creating a list

```
#Create a shopping list
shopping_list = ["bread","milk", "cheese","ham"]
```

Printing a list

Lists start at 0 (not 1).

```
#Print the whole list
print(shopping_list)
#Print the second element of the list
print(shopping_list[1])
#Print a slice of the list
print(shopping_list[2:4])
```

Changing the data in a list

```
#Adds an item to the end of the list
shopping_list.append("eggs")
#Removes only the first instance of an item
shopping_list.remove("cheese")
#Inserts an item into a list at a given index
shopping_list.insert(1,"sausages")
#Overwrite a list element
#Replace word bread with rolls in the list
stock = ["cheese","bread","juice","butter"]
stock[1] = "rolls"
```

Cycling through a list

```
#List of items in stock at shop
stock = ["cheese","bread","juice","butter"]
for item in stock :
    print (item)
```

Searching in a list

```
#in operator checks for a value in a list.
#Returns a boolean
print("rolls" in stock)
#index function returns the index position
#of a value is in a list
print(stock.index("rolls"))
#max function returns the highest value in a list
print(max(stock))
#min function returns the lowest value in a list
print(min(stock))
```

Use a built in module

Generate a random number

```
#Must import module you want to use
import random

#Generate a number between 1 and 65
number = random.randint(1,65)

#Generate a random even number using a step value
EvenNumber=random.randrange(0,65,2)
```

Suspend processing for a specified time

```
#import Time module
import time

#Suspend processing for 10 seconds
time.sleep(10)
```

String manipulation (advanced)

Using parts of a string

Strings start at 0 (not 1). Many functions that work on lists work on strings.

```
text = "sandwich"
print(text[2])
#Print part of a string
print(text[0:4])
#Print every other letter in a string
print(text[::2])
#Print every other letter in a string
#starting at end and working to start
print(text[::-2])
```

Looping through a string

```
sentence = 'The cat sat on the mat.'
for letter in sentence:
        print(letter)
```

Writing to and reading from text files

Write to a file (add \n at end of text to start a new line)

```
myFile = open('example.txt', 'wt')
myFile.write('I have written to a file.')
myFile.close()
```

Using 'a' instead of 'wt' opens the file in append mode.

Read the whole of a file

```
myFile = open('example.txt', 'rt')
contents = myFile.read()
print (contents)
myFile.close()
```

Using a For loop to read all lines in a file.

```
myFile = open('example.txt', 'rt')
for line in myFile:
    print (line)
myFile.close()
```

The text file needs to be in in the same folder as your program for this to work.