**Class 7**

**Key Ideas**

* **While** loops are conditional loops, in which the action is repeated until something happens
* How many times to execute the action is not known ahead of time
* The action of a **while** loop might not be executed at all if the condition is **False** to begin with
* The condition must eventually become **False**; otherwise, the loop is infinite
* An optional **else** clause can be added to a **while** loop, which specifies an action to be executed when the loop becomes **False**
* A useful application of while loops is to error-check: prompt the user and then loop to repeat the prompt until the user enters a valid value
* Since the number of times the action will be executed is not known ahead of time, it is sometimes useful to use a running count variable to count how many times the action is executed

**Built-ins**

**Functions**

**Statements**

**while**: repeats an action as long as a condition is **True**

**else**: optional clause; specifies an action to be executed when the action becomes **False**

**Assessment Questions**

For the following for loop:

numb = 10

while numb < 12:

numb = numb + 1

print('!')

How many times will the action be executed?

0

1

2

3

12

True/False: The action of a **while** loop will always be executed