Flow Control (while Loop and break Statement)

while Loop

- Do something WHILE a condition is **True**.
- Loop will STOP when condition is **False**.
- OR when a **break** statement is executed to exit the while loop.

REVIEW: Find the SUM of the numbers from 1 to 10.

```
n = 10
sum = 0
i = 1
while i <= n:
    sum = sum + i
    i = i + 1
print("The sum is ", sum)
```

n	sum	i	Condition (True or False) i < = n

Find the SUM of the numbers from 1 to 100000.

```
theSum = 0
count = 1
while count <= 100000:
    theSum = theSum + 1
    count = count + 1
print("The sum is ", theSum)
```

theSum	count	Condition (True or False) count < = 100000

TASK:

Find the SUM of numbers entered from a user, or just press enter to stop. Loop will stop when user presses <enter> (empty string “”)

REVIEW: Pseudocode Algorithm

set the sum to 0.0

input a string

while the string is not the empty string

    convert the string to a float

    add the float to the sum

    input a string

print the sum

```
theSum = 0.0
data = input("Enter a number or just press enter to quit: ")
while data != "":                    #empty string ""
        number = float(data)
        theSum = theSum + number
        data = input("Enter a number or just press enter to quit: ")

#The next statement is outside the loop
print("The sum is ", theSum)
```

Enter Code: today's_date.py

while Loop using a break Statement

TASK:
Find the SUM of numbers entered from a user, or just press enter to stop.

```
theSum = 0.0
while True:
        data = input("Enter a number or just press enter to quit: ")
        if data == "":                          #empty string ""
                break
        number = float(data)
        theSum = theSum + number

#The next statement is outside the loop
print("The sum is ", theSum)

#The break Statement will cause an exit from the while loop.
```

TASK:

Enter a numeric grade from 0 – 100. Make sure to handle any invalid numeric grades.

Enter Code: today's_date.py

```
while True:
        number = int(input("Enter a numeric grade from 0 – 100 "))
        if number >= 0 and number <=100:
                break
        else:
                print("Error: grade must be between 0 – 100 ")

print("The grade is ", number)
```

#If a user enters a number from 0 to 100, the if condition will be True and the break Statement will cause an exit from the while loop.

#If a user doesn't enter a number from 0 - 100, the else block will execute and the Error Message will print, and the while loop continues prompting the user for a numeric grade again.

```
done = False
while not done:
        number = int(input("Enter a numeric grade from 0 – 100 "))
        if number >= 0 and number <=100:
                done = True
        else:
                print("Error: grade must be between 0 – 100 ")

print("The grade is ", number)

#Same as previous slide without break statement.
```