Lecture 5

Repetition: While Loop

Making Decisions in a Program (revisit)

- Three basic control structures
  - Sequence
  - Selection
  - Repetition
- All procedures in an application are written using one of more of these structures.

Repetition Structure

- A loop is one of the most important structures in programming.
- Used to repeat a sequence of statements a number of times.
- The loop repeats sequence of statement either as long as or until a certain condition is true.
- Also referred to as **ITERATION** or loop.

Repetition Structure

- There are 2 repetition structures are introduce in Python programming language:
  - While Loop
  - For Loop
  - Nested Loop

While Loop

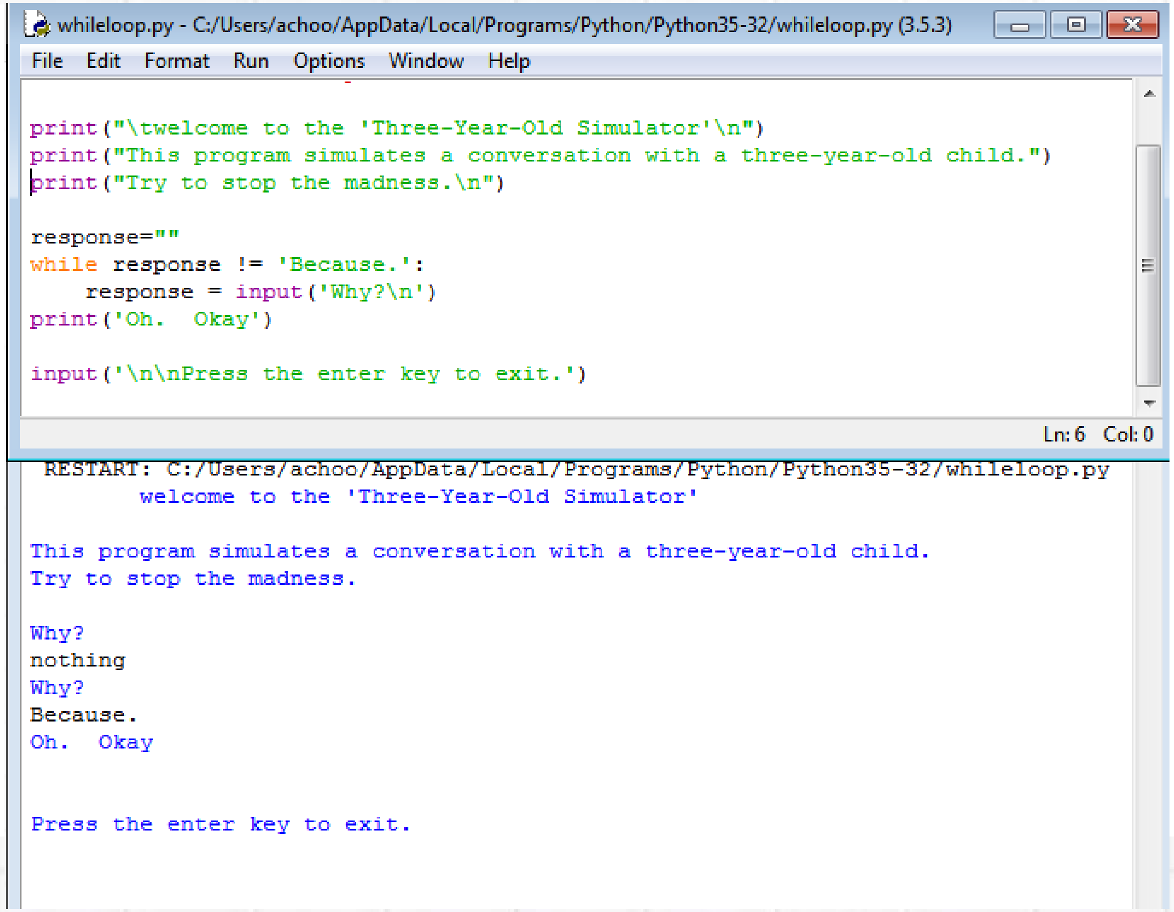
- Loops are used to repeatedly execute a block of program statements.
- The basic loop structure in Python is While Loop.
- Syntax:

```
While (expression):
    statement_1
    statement_2
    …
```

While Loop

- The while loop runs as long as the expression (condition) evaluates to True and execute the program block.
- The condition is checked every time at the beginning of the loop and the first time when the expression evaluates to False, the loop stops without executing any remaining statement(s).

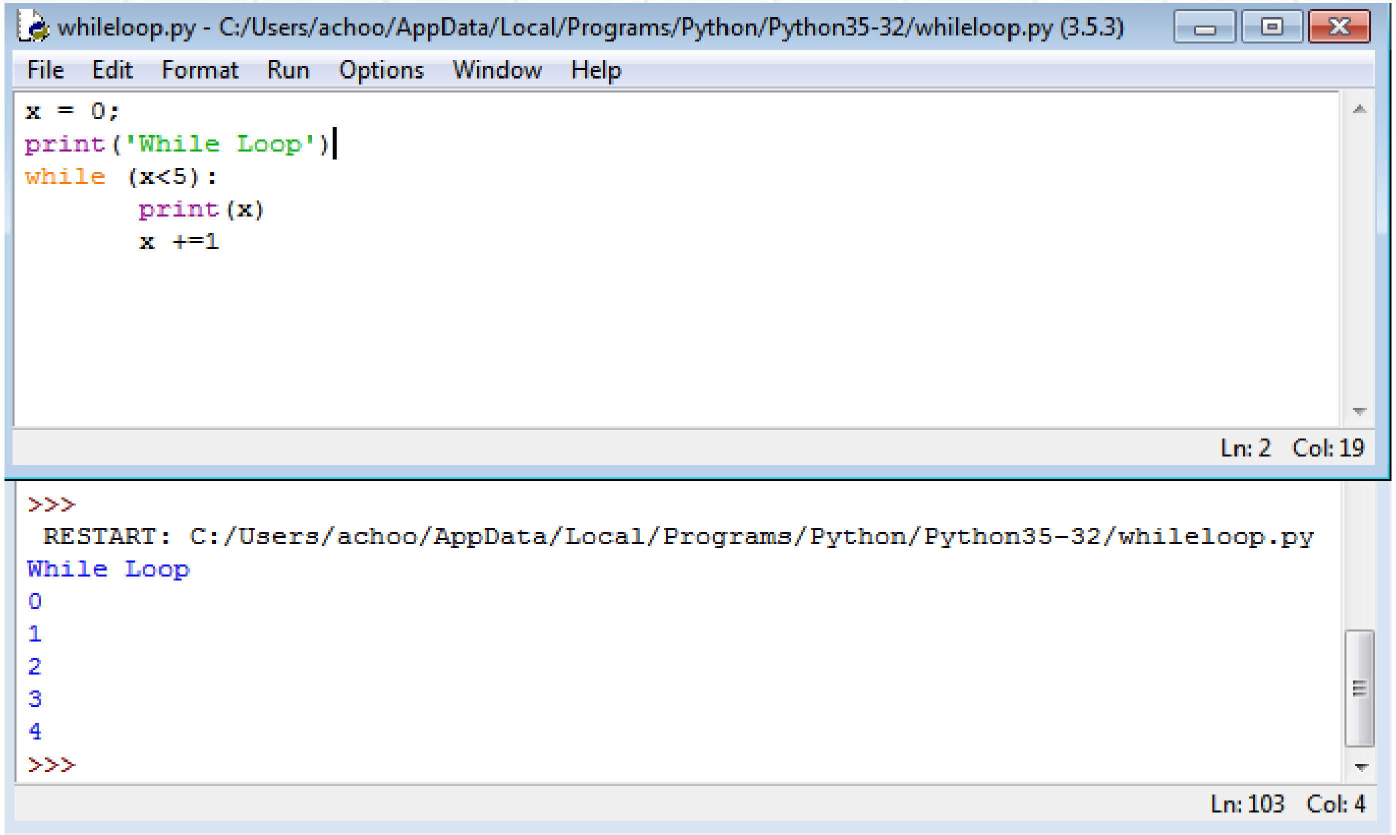
While Loop


whileloop.py - C:/Users/achoo/AppData/Local/Programs/Python/Python35-32/whileloop.py (3.5.3)
File
Edit
Format
Run
Options
Window
Help
Ln: 6
Col: 0

```
print("\twelcome to the 'Three-Year-Old Simulator'\n")
print("This program simulates a conversation with a three-year-old child.")
print("Try to stop the madness.\n")

response=""
while response != 'Because.':
    response = input('Why?\n')
print('Oh.  Okay')

input('\n\nPress the enter key to exit.')
```

```
 RESTART: C:/Users/achoo/AppData/Local/Programs/Python/Python35-32/whileloop.py
        welcome to the 'Three-Year-Old Simulator'

This program simulates a conversation with a three-year-old child.
Try to stop the madness.

Why?
nothing
Why?
Because.
Oh.  Okay


Press the enter key to exit.
```

While Loop



whileloop.py - C:/Users/achoo/AppData/Local/Programs/Python/Python35-32/whileloop.py (3.5.3)
File Edit Format Run Options Window Help
Ln: 2 Col: 19
Ln: 103 Col: 4

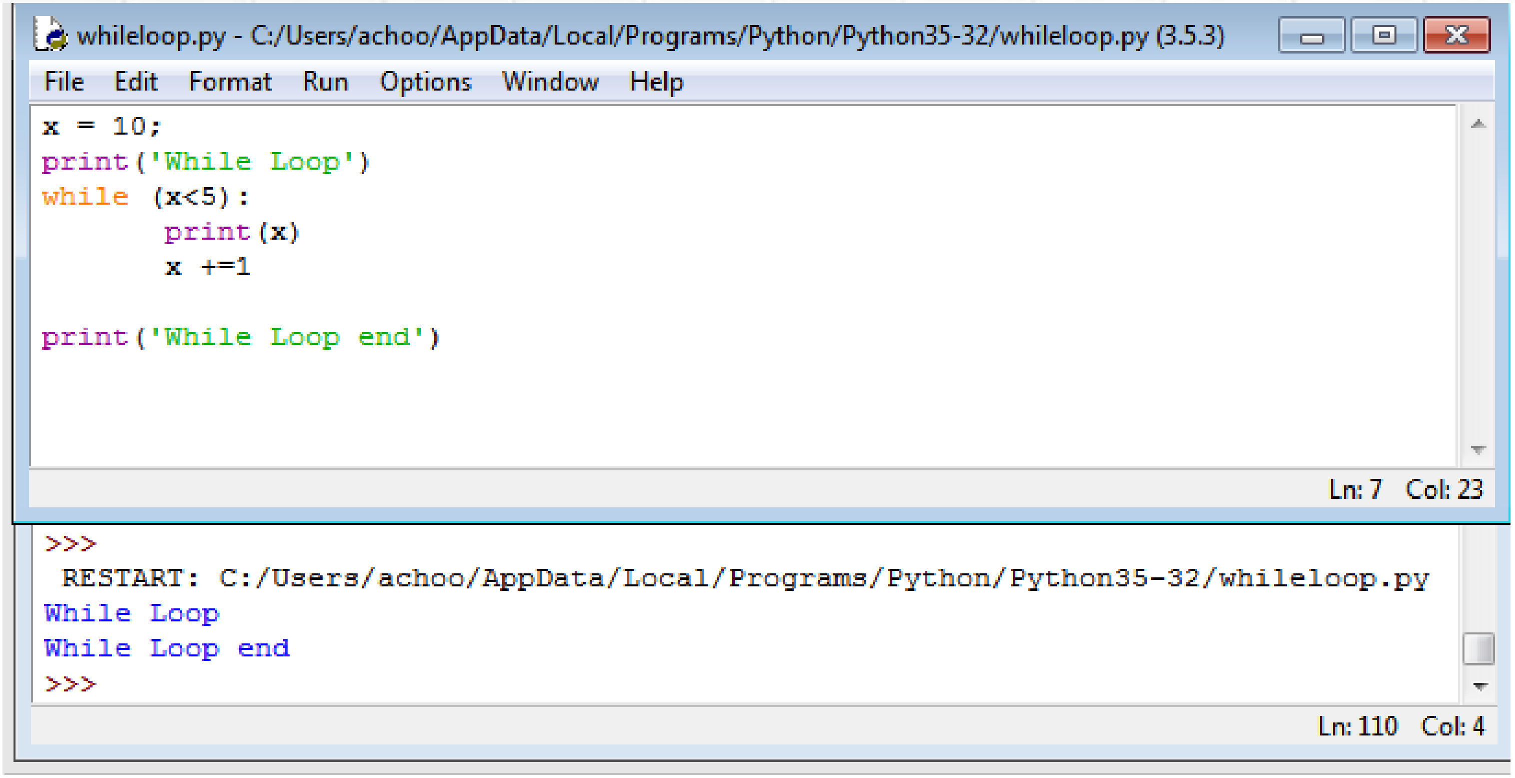
```
x = 0;
print('While Loop')
while (x<5):
        print(x)
        x +=1
```

```
>>>
 RESTART: C:/Users/achoo/AppData/Local/Programs/Python/Python35-32/whileloop.py
While Loop
0
1
2
3
4
>>>
```

While Loop

- Reminder: while loop tests the condition before the body of the loop (block of program statements) is executed.
- If the initial test returns false, the body is not executed at all.

While Loop



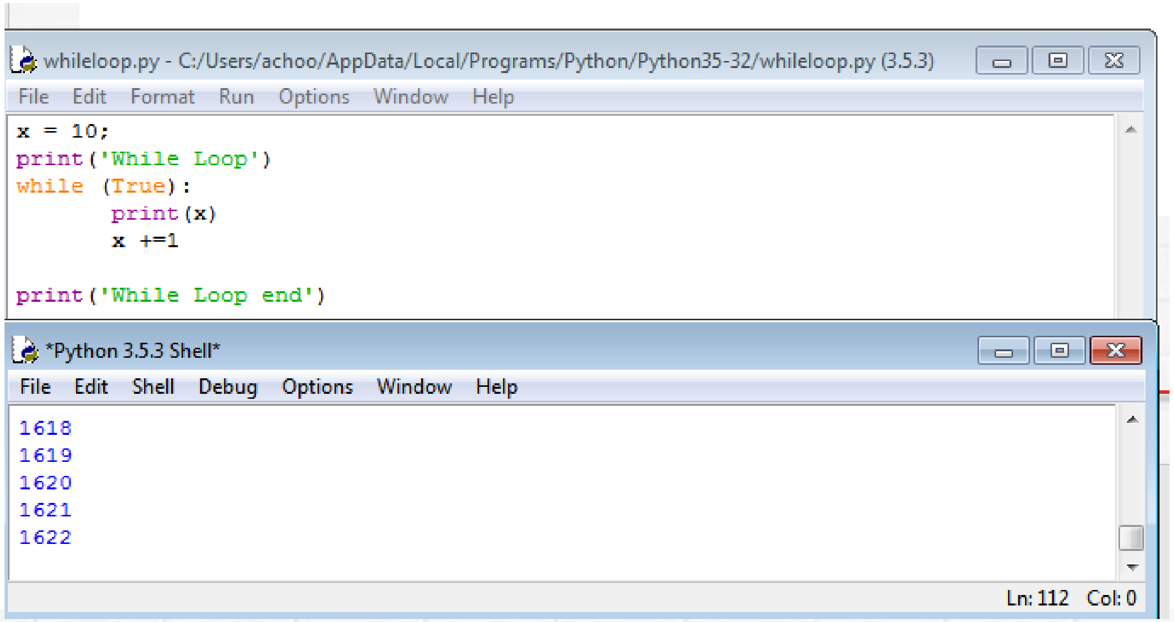
whileloop.py - C:/Users/achoo/AppData/Local/Programs/Python/Python35-32/whileloop.py (3.5.3)
File Edit Format Run Options Window Help
Ln: 7 Col: 23

```
x = 10;
print('While Loop')
while (x<5):
        print(x)
        x +=1

print('While Loop end')
```

```
>>>
 RESTART: C:/Users/achoo/AppData/Local/Programs/Python/Python35-32/whileloop.py
While Loop
While Loop end
>>>
```

Infinite Loop

- Infinite loop that using True as the condition.



whileloop.py - C:/Users/achoo/AppData/Local/Programs/Python/Python35-32/whileloop.py (3.5.3)
File Edit Format Run Options Window Help

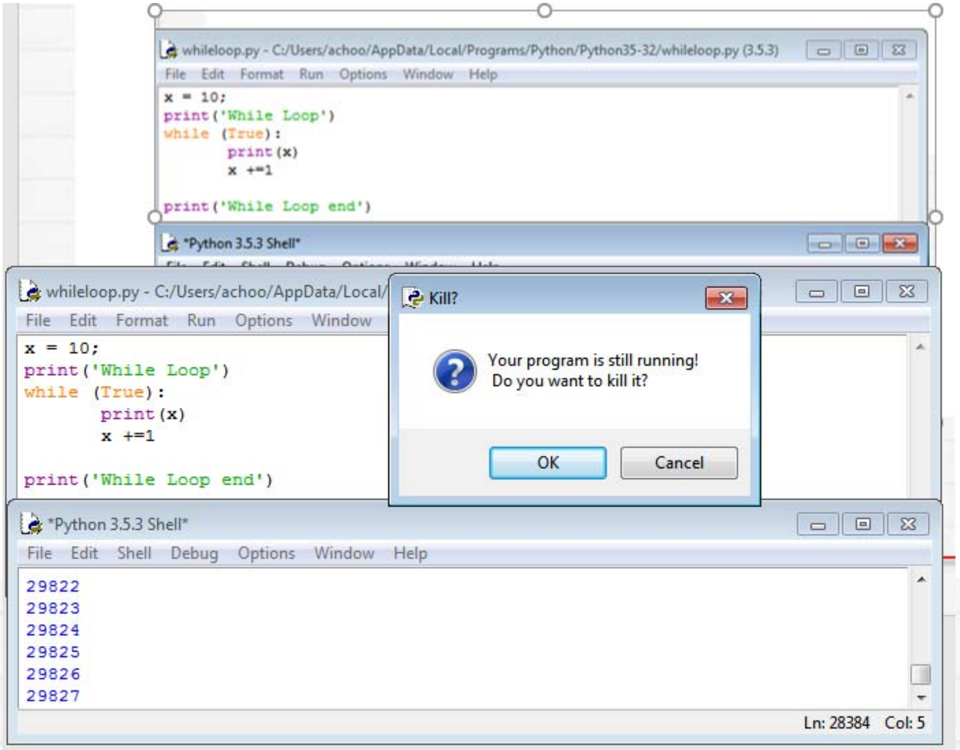
```
x = 10;
print('While Loop')
while (True):
        print(x)
        x +=1

print('While Loop end')
```

Python 3.5.3 Shell
File Edit Shell Debug Options Window Help

```
1618
1619
1620
1621
1622
```

Infinite Loop

- In order to stop an infinite loop. Click x – close button at the menu.



whileloop.py - C:/Users/achoo/AppData/Local/Programs/Python/Python35-32/whileloop.py (3.5.3)
File Edit Format Run Options Window Help

```
x = 10;
print('While Loop')
while (True):
        print(x)
        x +=1

print('While Loop end')
```

Python 3.5.3 Shell
whileloop.py - C:/Users/achoo/AppData/Local/
File Edit Format Run Options Window

```
x = 10;
print('While Loop')
while (True):
        print(x)
        x +=1

print('While Loop end')
```

Your program is still running!
Do you want to kill it?

OK
Cancel
Python 3.5.3 Shell
File Edit Shell Debug Options Window Help

```
29822
29823
29824
29825
29826
29827
```

Python: While and Else statement

- There is a structural similarity between while and else statement.
- Both have a block of statement(s) which is only executed when the condition is true.
- The difference is that block belongs to if statement executes once whereas block belongs to while statement executes repeatedly.

Python: While and Else statement

- You can attach an optional else clause with while statement.
- Syntax:

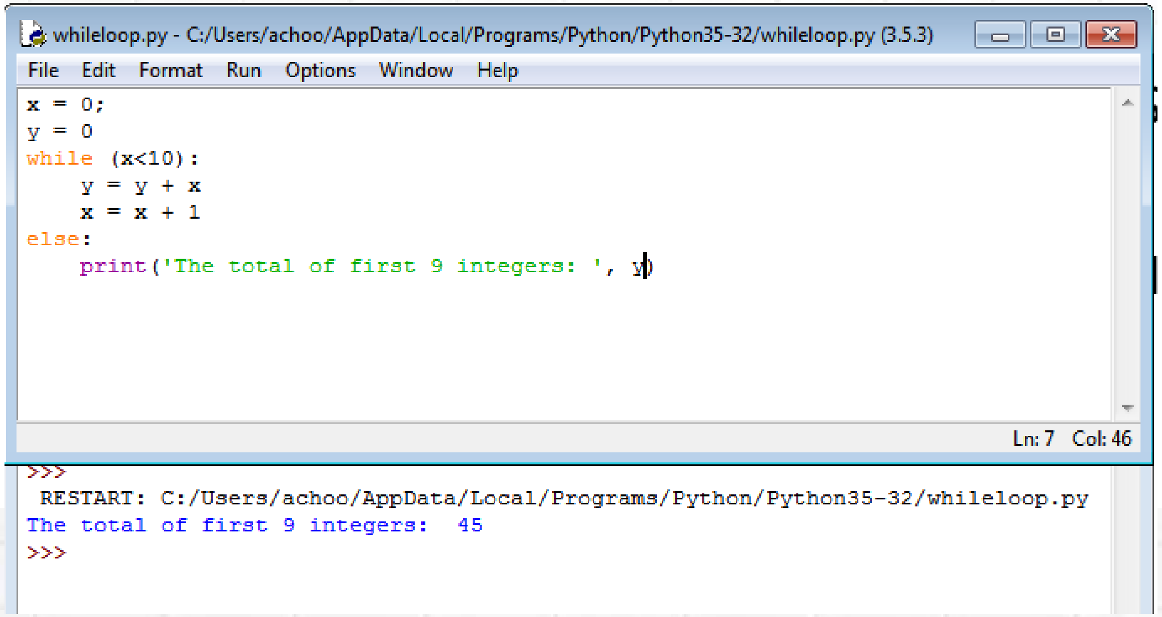
```
While (expression):
    statement_1
    statement_2
    …
Else:
    statement_3
    statement_4
    …
```

Python: While and Else statement

- The while loop repeatedly tests the expression (condition) and, if it is true, executes the first block of program statements.
- The else clause is only executed when the condition is false it may be the first time it is tested and will not execute if the loop breaks, or if an exception if raised.

Python: While and Else statement

- If a break statement executes in first program block and terminates the loop then the else clause does not execute.



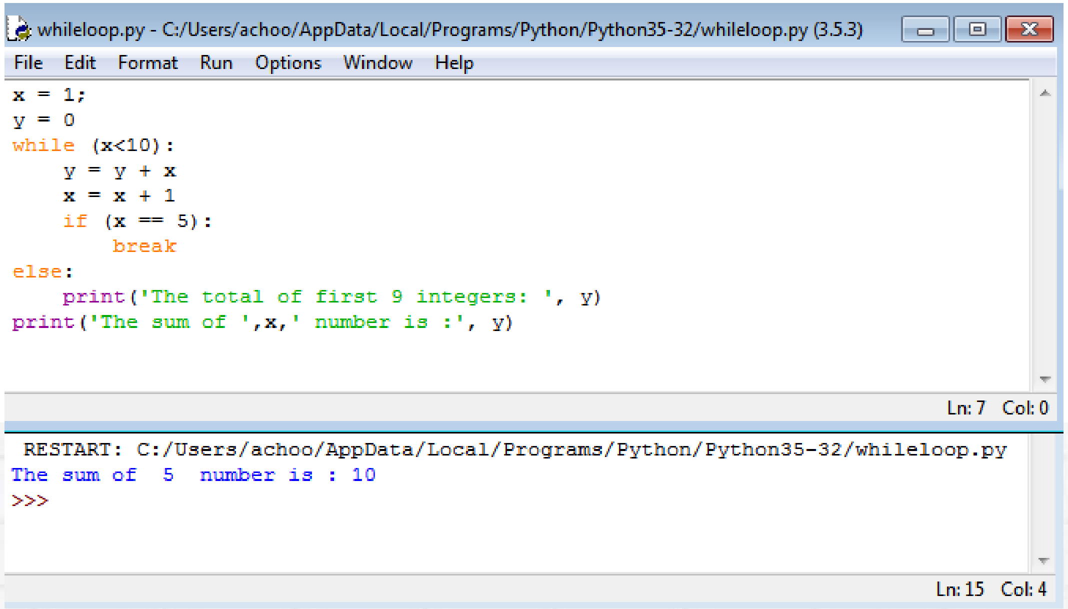
whileloop.py - C:/Users/achoo/AppData/Local/Programs/Python/Python35-32/whileloop.py (3.5.3)
File Edit Format Run Options Window Help
Ln: 7 Col: 46

```
x = 0;
y = 0
while (x<10):
    y = y + x
    x = x + 1
else:
    print('The total of first 9 integers: ', y)
```

```
>>>
 RESTART: C:/Users/achoo/AppData/Local/Programs/Python/Python35-32/whileloop.py
The total of first 9 integers:  45
>>>
```

While loop with if-else and break statement

- The loop is terminated when x becomes 5.
- ‘Break’ statement is used to terminate the while loop without completing it, therefore program control goes to outside the while-else structure and execute the next print statement


whileloop.py - C:/Users/achoo/AppData/Local/Programs/Python/Python35-32/whileloop.py (3.5.3)
File Edit Format Run Options Window Help
Ln: 7 Col: 0
Ln: 15 Col: 4

```
x = 1;
y = 0
while (x<10):
    y = y + x
    x = x + 1
    if (x == 5):
        break
else:
    print('The total of first 9 integers: ', y)
print('The sum of ',x,' number is :', y)
```

```
 RESTART: C:/Users/achoo/AppData/Local/Programs/Python/Python35-32/whileloop.py
The sum of  5  number is : 10
>>>
```

Python Bitwise Operators

Operator	Shorthand	Expression	Description
&	And	X & Y	Bits that are set in both x and y are set.
|	Or	X | Y	Bits that are set in either x or y are set.
^	Xor	X ^ Y	Bits that are set in x or y but not both are set.
~	Not	~X	Bits that are set in x are not set, and vice versa.
<<	Shift left	X << Y	Shift the bits of X, Y steps to the left.
>>	Shift right	X >> Y	Shift the bits of X, Y steps to the right.

KNOW
ING