```
Current fruit : apple

Current fruit : mango

Good bye!
```

Here, we took the assistance of the len() built-in function, which provides the total number of elements in the tuple as well as the range() built-in function to give us the actual sequence to iterate over.

Using else Statement with Loops

Python supports to have an **else** statement associated with a loop statement.

- If the **else** statement is used with a **for** loop, the **else** statement is executed when the loop has exhausted iterating the list.
- If the **else** statement is used with a **while** loop, the **else** statement is executed when the condition becomes false.

The following example illustrates the combination of an else statement with a for statement that searches for prime numbers from 10 through 20.

```
#!/usr/bin/python


for num in range(10,20):  #to iterate between 10 to 20

   for i in range(2,num): #to iterate on the factors of the number

      if num%i == 0:      #to determine the first factor

         j=num/i          #to calculate the second factor

         print '%d equals %d * %d' % (num,i,j)
```

```
         break #to move to the next number, the #first FOR

   else:                  # else part of the loop

      print num, 'is a prime number'
```

When the above code is executed, it produces the following result:

```
10 equals 2 * 5
```