**Describe two of the functions/methods and one listed in Table 2.3 on page 39. Provide an example of how each one is used. Provide an example of the use of chained methods.**

The lower method takes a string and converts it to lowercase. For example:

>>> print("CityU Seattle".lower())  
cityu seattle

The count method searches the string for the entered substring, and counts the number of occurrences. For example:

>>> print("Fuzzy Wuzzy was a bear".count('z'))  
4

Chained methods involve the combining of multiple methods into a single line of code. An example would be chaining the two methods discussed above to count the number of m’s in an email address.

>>> email = "FIRSTNAMELASTNAME@gmail.com"  
>>> email.count('m')  
2  
>>> email.lower().count('m')  
4

As you can see, the results are different from two seemingly similar lines of code.

**In your own words, describe the difference between immutable and mutable objects.**

Mutable objects are able to be changed without changing their memory location. Lists are mutable, as they can be appended with additional information. Immutable objects cannot be changed in place, and changes will result in the object moving to a new memory location. Strings are immutable, and invoking a method on a string will result in a new string being created—not changing the existing string.

**What is the difference between a tuple and a list? When might a programmer use one instead of the other? Provide an example.**

A tuple is immutable, and its contents cannot be modified without changing the entire tuple. This would be useful when storing related values of different types, such as an employee’s name, address, phone number, etc. The items in a tuple can also be assigned their own variable. A list is mutable, and its contents can be appended, deleted, modified, and changed as desired. Lists are useful for storing information of the same type. One use for a list would be groceries, such as bread, cheese, lettuce, etc.