**Define *AND* give an example for each of the following vocabulary words as related to the Python programming language.**

**Element**

The Element type is a flexible container object, designed to store hierarchical data structures in memory. The type can be described as a cross between a list and a dictionary.

**Example:**

To create an element, call the Element constructor, and pass the tag string as the first argument:

rom elementtree.ElementTree import Element

root = Element("root")

You can access the tag string via the tag attribute:

print root.tag

To build a tree, create more elements, and append them to the parent element:

root = Element("root")

root.append(Element("one"))

root.append(Element("two"))

root.append(Element("three"))

**Immutable**

An immutable object cannot be changed after it is created

**Example:**

Object of types like int, float, bool, str, tuple, and Unicode are immutable

**Index**

index() is an inbuilt function in Python, which searches for given element from start of the list and returns the lowest index where the element appears

**Example**

list1 = [1, 2, 3, 4, 1, 1, 1, 4, 5]

# Will print index of '4' in sublist

# having index from 4 to 8.

print(list1.index(4, 4, 8))

**IndexError**

An IndexError is a common exception found in Python, it is raised whenever attempting to access an index that is outside the bounds of a list

**Example**

def main():

try:

# Create list and populate with Books.

books = list()

books.append(Book("Shadow of a Dark Queen", "Raymond E. Feist", 497, datetime.date(1994, 1, 1)))

books.append(Book("Rise of a Merchant Prince", "Raymond E. Feist", 479, datetime.date(1995, 5, 1)))

books.append(Book("Rage of a Demon King", "Raymond E. Feist", 436, datetime.date(1997, 4, 1)))

# Output Books in list, with and without index.

Logging.line\_separator('Books')

log\_list(books)

Logging.line\_separator('Books w/ index')

log\_list(books, True)

# Output list element outside bounds.

Logging.line\_separator('books[len(books)]')

Logging.log(f'books[{len(books)}]: {books[len(books)]}')

except IndexError as error:

# Output expected IndexErrors.

Logging.log\_exception(error)

except Exception as exception:

# Output unexpected Exceptions.

Logging.log\_exception(exception, False)

**Lists**

A list is a collection which is ordered and changeable, allows duplicate members.

**Example:**

Thislist = [“apple”, “banana”, “cherry”]

Print(thislist)

**Mutable**

A mutable object can be changed after it is created.

**Example:**

Objects of types like list, set, and dict are mutable.

**Sequence**

Sequence is the generic term for an ordered set.

**Example:**

There are several types of sequences in Python, including Lists, Tuples, and Strings.

**Slice**

A slice object is used to specify how to slice a sequence. You can specify where to start the slicing, and where to end. You can also specify the step, which allows you to e.g. slice only every other item.

The slice() function returns a slice object.

**Example:**

a = ("a", "b", "c", "d", "e", "f", "g", "h")  
x = slice(2)  
print(a[x])

**Tuples**

A tuple is a collection which is ordered and unchangeable, allows duplicate members.

**Example:**

thistuple = ("apple", "banana", "cherry")  
print(thistuple)