Lists:
A collection of items of any type:  [10, 3, 22], ["sh", "ch"]
A mixture of types is okay:  ["apple", 25, "q"]
Nested lists are allowed:  [["nj", "trenton"], ["md", "annapolis"]]
Use indexes in square brackets to refer to items within the list. Numbering starts at 0.

Unlike strings, lists are mutable -- that is, they can be changed.

List Methods and Functions

If L is a list, here are some methods and operators you can use with L.

These methods and operators change L.

# List elements can appear on the left hand side of assignments
L = ["red", "blue", "green"]
print L  # Prints ["red", "blue", "green"]
L[1] = "orange"
print L  # Prints ["red", "orange", "green"]

# Adding things to the end of the list: append, extend methods
# Note that + concatenates lists: L = L + ["foo", "bar"]
# Thus, + can be used as an alternative to extend.

L = ["apple", "banana"]
print L  # Prints ["apple", "banana"]
L.append("peach")
print L  # Adds "peach" to the end of L
print L  # Prints ["apple", "banana", "peach"]

L.extend(["plum", "peach"])
print L  # Adds "plum" and "peach" to the end of L
print L  # Prints ["apple", "banana", "peach", "plum", "peach"]

# Deleting items: pop method, remove method, del operator
L = ["a", "b", "c", "d", "e", "f", "g"]
item = L.pop(2)
print L  # Remove the item in position 2 and return it
print item  # Prints ["a", "b", "e", "f", "g"]

del L[1]
print L  # Remove the item in position 1
print L  # Prints ["a", "d", "e", "f", "g"]

del L[1:3]
print L  # Remove the items in positions 1 through 2
print L  # Prints ["a", "e", "f", "g"]

L.remove("f")
print L  # Remove (the first occurrence of) "f"
print L  # Prints ["a", "e", "g"]
Nested Lists

Lists can contain other lists as elements.

[ ["apple", "banana"], ["cat", "dog"], ["elephant"] ]

Nested lists can be used to represent data in grid form:

_A grid of numbers:_

```
2  4  6  8
3  7  9 11
```

[ [2, 4, 6, 8], [3, 7, 9, 11] ]

_A board for a board game:_

```
X  X  O
X  O  X
O  O  X
```

[ ['X', 'X', 'O'], [ 'X', 'O', 'X' ], [ 'O', 'O', 'X' ] ]

Simple references and assignments:

```python
table = [ [2, 4, 6, 8], [3, 7, 9, 11] ]

print table[0][2]  # Prints 6
print table[1][1]  # Prints 7

table[1][0] = 15  # Now the list is [ [2, 4, 6, 8], [15, 7, 9, 11] ]
```