6. Working With Lists
==================================

looping through an entire list
You'll often want to run through all entries in a list, performing the same task with each item

```
magicians = ['alice', 'david', 'carolina']
for magician in magicians:
    print(magician)
```

making numerical lists
Using the range() Function
Python's range() function makes it easy to generate a series of numbers.
For example, you can use the range() function to print a series of numbers like this:
```
for value in range(1,5):
    print(value)
```

```
numbers = list(range(1,6))
print(numbers)
```

```
even_numbers = list(range(2,11,2))
print(even_numbers)
```

```
digits = [1, 2, 3, 4, 5, 6, 7, 8, 9, 0]
>>> min(digits)
0
>>> max(digits)
9
>>> sum(digits)
```

working with Part of a list
==
Slicing a List
To make a slice, you specify the index of the first and last elements you want to work with.

```
players = ['charles', 'martina', 'michael', 'florence', 'eli']
print(players[0:3])
```

```
players = ['charles', 'martina', 'michael', 'florence', 'eli']
print(players[:4])
```

```
players = ['charles', 'martina', 'michael', 'florence', 'eli']
print(players[2:])
```

#copy

```
my_foods = ['pizza', 'falafel', 'carrot cake']
v friend_foods = my_foods[:]
```

tuples

Lists work well for storing sets of items that can change throughout the
life of a program. The ability to modify lists is particularly important when
you're working with a list of users on a website or a list of characters in a
game. However, sometimes you'll want to create a list of items that cannot
change. Tuples allow you to do just that. Python refers to values that cannot
change as immutable, and an immutable list is called a tuple.

Defining a Tuple

A tuple looks just like a list except you use parentheses instead of square
brackets.