

## Week 2: Labs

```
#####  
x=123 + 222 # Integer addition  
print(x)  
y=1.5 * 4 # Floating-point multiplication  
print(y)  
z=2 ** 100 # 2 to the power 100, again  
print(z)
```

---

```
import math  
math.pi  
print(math.pi)  
print(math.sqrt(81))
```

---

```
import random  
print(random.random())  
print(random.choice([1, 2, 3, 4]))
```

---

String

---

```
S = 'Python'  
print(S[-1]) # The last item from the end in S  
print(S[-2]) # The second-to-last item from the end
```

```
print(S[-1]) # The last item in S
print(S[len(S)-1]) # Negative indexing, the hard way
print(S[1:3]) # Slice of S from offsets 1 through 2 (not 3)
print(S[1:]) # Everything past the first (1:len(S))
print(S[0:3]) # Everything but the last
print(S[:3]) # Same as S[0:3]
print(S[:-1]) # Everything but the last again, but simpler (0:-1)
print(S[:]) # All of S as a top-level copy (0:len(S))
```

---

```
S = 'Toronto'
print(S.find('ro')) # Find the offset of a substring in S
print(S.replace('To', 'Mo')) # Replace occurrences of a string in S with another
```

-----

List

```
L = [123, 'spam', 1.23] # A list of three different-type objects
print(len(L))
```

```
print(L[0]) # Indexing by position
```

```
print(L[:-1]) # Slicing a list returns a new list
```

---

```
L = [123, 'spam', 1.23] # A list of three different-type objects
```

```
print(L[1])
```

```
print(L[5])
```

---

dictionaries

```
D = {'food': 'Spam', 'quantity': 4, 'color': 'pink'}
```

```
print(D['food'])
```

```
print(D['quantity'] += 1)
```

```
--
```

Tuples

```
T = (1, 2, 3, 4) # A 4-item tuple
```

```
print(len(T)) # Length
```

```
print(T + (5, 6)) # Concatenation
```

```
print(T[0]) # Indexing, slicing, and more
```

---

---

**What is the difference between tuples and lists?**

---

## Chapter 05

```
print(int(3.1415)) # Truncates float to integer
```

```
print(float(3)) # Converts integer to float
```

---

```
a = 3 # Name created: not declared ahead of time
```

```
b = 4
```

```
print(b * 3, b / 2) # Multiplication (4 * 3), division (4 / 2)
```

```
print(a % 2, b ** 2) # Modulus (remainder), power (4 ** 2)
```

```
num = 1 / 3.0
```

```
print(num) # Auto-echoes
```

```
print('%e' % num) # String formatting expression
```

```
print('{0:4.2f}'.format(num)) # String formatting method: Python 2.6, 3.0, and later
```

---

```
print(1 < 2) # Less than
```

```
print(2.0 >= 1) # Greater than or equal: mixed-type 1 converted to 1.0
```

```
print(2.0 == 2.0) # Equal value
```

```
print(2.0 != 2.0) # Not equal value
```

```
import math
```

```
print(math.floor(2.5)) # Closest number below value
```

```
print(math.floor(-2.5))
```

```
print(math.trunc(2.5)) # Truncate fractional part (toward zero)
```

```
print(math.trunc(-2.5))
```

---

```
print(oct(64), hex(64), bin(64))
```

```
import math
```

```
print(abs(-42.0), sum((1, 2, 3, 4))) # Absolute value, summation
```

```
print(min(3, 1, 2, 4), max(3, 1, 2, 4))
```

---

```
x = set('abcde')
```

```
y = set('bdxyz')
```

```
print(x)
```

```
print(x-y) #difference
```

```
print(x | y) #union
```

```
print(x&y) # intersection
```

---

```
x = set('abcde')
```

```
print('e' in x) # Membership (sets)
```

```
print('e' in 'Gameplan', 22 in [11, 22, 33])
```

---

```
sets remove duplicates
```

```
L = [1, 2, 1, 3, 2, 4, 5]
```

```
print(set(L))
```

---

```
engineers = {'bob', 'sue', 'ann', 'vic'}
```

```
managers = {'tom', 'sue'}  
print('bob' in engineers)  
print(engineers & managers)  
print(engineers | managers)  
print(engineers - managers)  
print(managers - engineers)
```