python 3

data types  
integer  
float  
Boolean (True & False) ~~(true & false)~~

print ("this will display")

#this is a comment in python

result = 7/2 is 3

result 7.0 / 2 is 3.5

float(x) returns x converted to float  
int(x) converts to integer  
str(x) converts to string

string\_var = """by using triple quotes a string

can span multiple lines.

this is just an example"""

\*\* indicated exponents  
3\*\*2 = 9 #(3^2)

using triple quotes without assigning the string to a variable is also a way to have a multi line comment

backslash allows special characters. a quotation mark may close a quote even if you didn’t want it to. \" or \' will prevent that

"STRING"[3] will return the 3rd indexed character (the 4th character), "I"

**String Methods**

len(x) returns length of string

x.lower() returns string, all lowercase

x.upper() returns string, all uppercase

str(x) converts x to a string

methods that use dot notation (x.lower()) only work with strings

insert strings into a larger string without concatenation

"Let's not go to %s. 'Tis a silly %s." % (string\_1, string\_2)

%s is the placeholder in a string. % after the string indicates replacement string.

date/time

from datetime import datetime

print datetime.now()

%02d this is replaced with an integer padded with zeroes to 2 characters wide. d means the number is a signed integer

from datetime import datetime

now = datetime.now()

print '%02d-%02d-%04d' % (now.month, now.day, now.year)

print '%02d:%02d:%02d' % (now.hour, now.minute, now.second)

**Control Flow**

6 comparitors   
== equal to  
!= not equal to  
<  
<=  
>  
>=

Boolean operators and, or, not

not is evaluated first, and second, or last

conditionals  
if 8 < 9:  
 print "Eight is less than nine!"

the indentation lets the program know what should follow the conditional

if answer > 5:  
 return 1  
elif answer< 5:   
 return -1  
else:  
 return 0