Exercise 1

```python
In [1]:
x = 5.34
    # if you do it like this, python decides how many digits to give you
    print("The value of x is",x)
    print(" ")  # a blank line

    # This is one way to specify 4 digits ("the old way")
    print("The value of x is %8.4f" %x)
    # x.y means use a total of x characters, y characters after the decimal point.
    # The "f" is for a floating number
    print( ' ')

    # The "zero" here is the zero-th argument of "format"
    print("The value of x is {0:8.4f}".format(x) )

    # Yes, it is ugly.  I always forget and have to look it up.

The value of x is 5.34
The value of x is  5.3400
The value of x is  5.3400
```

Exercise 2

```python
In [2]:
x = 8.92
    print("The integer value of x is  ", int(x))
    print("The nearest integer to x is", round(x))

The integer value of x is  8
The nearest integer to x is 9
```

Exercise 3

```python
In [3]:
    import math
    print("pi is ", math.pi)
    print("The sin of 30 degrees is ",math.sin(math.pi/6))
    print("The sin of 30 degrees is %6.4f" % math.sin(math.pi/6))

pi is  3.141592653589793
The sin of 30 degrees is  0.4999999999999994
The sin of 30 degrees is 0.5000
```

Exercise 4

```python
In [4]:
x = True
b = 34.5
c = 45
print(x*b, x*c, x+b, c-x)
```
In [5]:

# There are a zillion ways of doing this..

j = 597
print("j in binary is", bin(j))

bit4Only = 2**4  # This variable has all bits = 0 except for bit 4
print("bit4Only in binary is", bin(bit4Only))

# make a variable that is like j but with all bits zeroed except for bit 4
temp = j & bit4Only
print(bin(temp))

# if bit 4 is zero then temp is zero
# if bit 4 is one then temp is not zero. In fact temp should be = bit4
bit4 = int(temp == bit4Only)
print("Bit 4 of", j, "is", bit4)

j in binary is 0b1001010101
bit4Only in binary is 0b100000
0b100000
Bit 4 of 597 is 1

In [6]:

# Again there are a zillion ways of doing this

# This variable has bits 6,7,8 set to 1
mask = 2**6 + 2**7 + 2**8
print(bin(mask))

# make a variable like j with all bits except 6,7,8 turned off
temp = j & mask
print(bin(j), bin(temp))

# Shift bits by 6 place
result = (temp >> 6)
print("The number between 0 and 7 encoded in bits 6,7,8 is ", result)

0b111000000
0b100101101 0b1000000
The number between 0 and 7 encoded in bits 6,7,8 is 1