Exercise: Coin toss

What is the probability of getting at least 7 heads when a coin is flipped 10 times?

1. Construct a model that uses random numbers

First we want to import a function for making random numbers. We'll use `random` from the module `random`.

```
from random import random
```

`random()` gives us a random number in the range $0 \leq x < 1$. Let's try it out.

```
print(random())
```

Creating one random number isn't the most illustrative. Here we use a loop to print 10 random numbers.

```
for i in range(10):
    print(random())
```

We can use this is simulate a coin toss by testing the value and taking this to be heads if it is less than 0.5 and tails otherwise.

Edit the code inside your `for` loop so that instead of printing out the random number, it does this test.

```
if random() < 0.5:
    print("H")
else:
    print("T")
```

If you run this, you should get a print out of a set of H's and T's. Did you get at least 7 heads?

Really, we want to count how many times we got H and how many times we got T. We can use a variable to count heads, since that's the one we're interested in. We start the counter at `0` outside our `for` loop, and instead of printing `"H"` and `"T"` we want to count heads. At the end, we print our counter to find out how many heads there were.

```
heads = 0
for i in range(10):
    if random() < 0.5:
        heads += 1
print(heads)
```

The question we are interested is whether `heads` is at least 7. We can test this.

```
if heads >= 7:
    print("yes")
```

If you run this a few times, you may find the number of times you get a `"yes"` is pretty low. But how low?

2. Run this many times

We have written code that simulates tossing a coin ten times and counts whether at least seven of the outcomes were heads.

Now we want to run this many times, for now let's say ten times. To do this, we put the whole code block inside another loop. It's important to use a different variable for our loop counter - if we use `i` again, the inner `for` loop will overwrite the value of the outer `for` loop and our simulation will never end.

One more thing - we don't want a big screen of `yes`es. Instead we want to use another counter to count how many times the number of heads was at least 7.

```
counter = 0
for j in range(10):
    heads = 0
    for i in range(10):
        if random() < 0.5:
            heads += 1
    if heads >= 7:

        counter += 1
print(counter)
```

3. Statistically analyse the results

We have output the number of times the simulation tossed at least seven heads. What statistics might we like to display?

Instead of the raw count, we could display the proportion of the time the condition was met. This is an approximation of the probability of tossing a coin ten times and getting at least four heads.

```
print(counter/10)
```

If you run your code a few times, it may give different estimates for this probability. When I ran it, I mostly got `0.2`, but sometimes `0.1` or even `0.0`.

We can improve our estimate by increasing the number of times we run our

simulation. Edit your outer loop to run the simulation more times, and remember to edit the calculation at the end to match (or you could use a variable for this number in both places to save re-editing the code if you want to change this again). (For example, I edited mine to run a million times (`10**6`) and the results from multiple runs agreed to two decimal places.)