**Unit 4: for loops and lists** | **Application: A Toss of the Dice**

In this lesson, you will make a simulation of dice tossing using random numbers.

**Objectives:**

- Use **randint()** to simulate dice tosses
- Use a list to keep track of totals
- Analyze the totals

---

When you toss a pair of fair 6-sided dice, the sum of the two dice will be a value from 2 to 12. Which sum is most likely? Least likely? Write a program to simulate the dice tosses and analyze the frequency of each of the eleven sums.

1. Start a new Python file using the *Random Simulations* template as in the last lesson.

   Use a list of 11 elements, all 0, to keep track of the 11 sums (2…12). An effortless way to create this list in Python is:

   **totals = [0] * 11**

   This means 'make a list of 11 elements, all 0'. Remember 'replication' from the last lesson?

   (Try this expression in a Python Shell to see that it works.)

2. Use an **input** statement to ask how many times to toss the dice:

   **trials = int(input("# of trials?"))**

   You can find '#' and '?' on the punctuation key.

   Now make a **for** loop to perform each trial (dice toss):

   **for i in range(trials):**

3. Now, in the loop block, 'toss the dice'. Use two variables, die1 and die2. Assign each a random integer from 1 to 6. Try it yourself first and then move to the next step…

4. Did you write **= randint(1,6)** for each?

   Add the two die values together:

   **sum = die1 + die2**

   The sum ranges from 2 to 12 but the index of the list totals ranges from 0 to 10. We must subtract 2 from the sum to get the proper list index. To add 1 to the corresponding element in the list of totals looks like this:

   **totals[sum - 2] = totals[sum - 2] + 1**

   You may recognize this statement as a counter like c=c+1. Each element of the list is a separate counter.

5. That's it for the loop *block*. When the trials are complete, we want to see the list of **totals**.

   Dedent (un-indent) to the beginning of the next line and print the totals list:

   **print(totals)**

6. Test the program now: When you run the program, try a small number like 10 for the number of trials. Check to make sure the sum of the elements in the list is 10. You will probably not get these same numbers (they are random, right?) but the sum of the numbers should equal the number of trials. Re-run with the same number of trials and you will see a different list.

   When you are satisfied that the program runs as expected, try a large number of trials, say, 500, 1000, 5000. Which element of the totals list is the largest? Smallest?

7. Your task: Write a **for** loop that divides each value in the list by the number of trials. Print the result. What is the significance of these numbers?

Start with:

**for i in range(11):**

You might want to **round** the long decimals to a fewer number of places. What would be appropriate? Use **round(n, #places)**.