Unit 5: Lists, Graphics, and Dynamic Programs | **Skill Builder 3: Simulations**

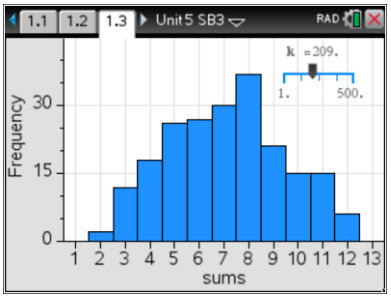
In this lesson, you will learn about using dynamic programs to produce other types of graphs using the **Data & Statistics** app.

Objectives:

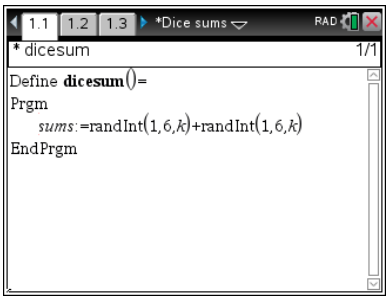
- Write a program that dynamically updates a graphical display
- Write a program that simulates a real-world occurrence (tossing a pair of dice) and plot the results in a meaningful graph

Dynamic programs can produce much more than random (or not-so-random) scatter plots.

In this lesson, we'll develop a demonstration of the distribution of the sum of two fair 6-sided dice when they are rolled a large number of times.



When you toss a pair of 6-sided dice, the sum of the values on the two dice can range from a low of 2 (by rolling two ones) to a high of 12 (by rolling two sixes). But, are the eleven totals all equally likely, or do some totals occur more often than others? Let's develop a TI-Nspire™ CX activity that allows us to simulate the experiment using a program and a graph.

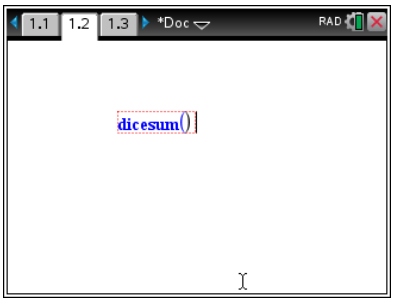


1. Begin a new Problem either by starting a new document or select **doc > Insert > Problem**.
2. Select **Add Program Editor > New…** from the menu, and name the program.
3. The program contains just one statement and *no argument*:

   sums:=randInt(1, 6, k) + randInt(1, 6, k)

   Two global variables will be established when the program runs:

   k represents the number of tosses of the pair of dice (trials).
   sums is a list holding each of the **k** sums of the tosses (outcomes).
4. Press **ctrl+B** to store the program. Do not run it yet.



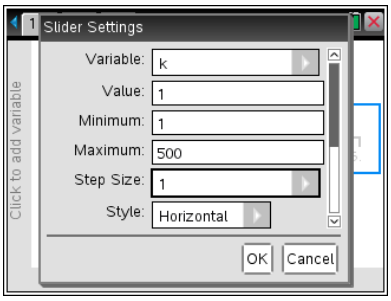
5. Insert a **Notes** app by selecting **doc > Insert > Notes** or by selecting **ctrl+doc** or **ctrl+I** and then selecting **Add Notes**.
6. Insert a Math Box by selecting **menu > Insert > Math Box**.
7. Type the name of the program and a set of parentheses, and press **enter**.

*Don't be surprised to see an error message since the program is being run and the global variable **k** in the program has no value yet.*

8. Press **enter,** or click **OK**.

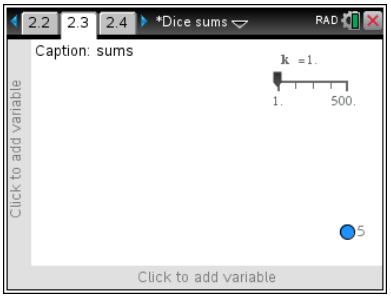
9. Insert a Data & Statistics app by selecting **doc > Insert > Data & Statistics** or by selecting **ctrl+doc** or **ctrl+I** and then selecting **Add Data & Statistics**.
10. Insert a slider (**menu > Actions > Insert Slider**) for the variable *k* with a value of 1, a minimum of 1, a maximum of 500, and a step size of 1.

    The variable *k* represents the number of times the dice are tossed (trials).
11. Click **OK**, and then place the slider near the top of the page.

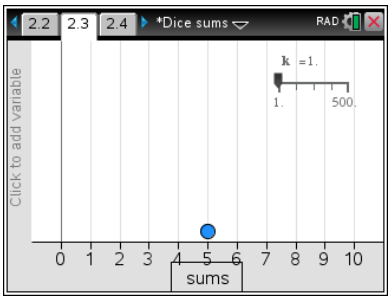


Now that *k* is defined, the program runs from within the Notes app. You will see 'Caption: sums' at the top of the app and a dot representing the sum of a pair of dice.

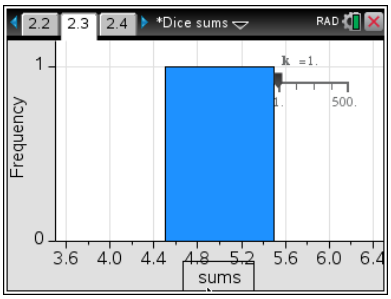
Don't use the slider yet!



12. Click the text 'Click to add variable' at the bottom of the app, and select **sums** for the horizontal axis.
    - The result is a vertical 'dot plot' of the *sums* list.

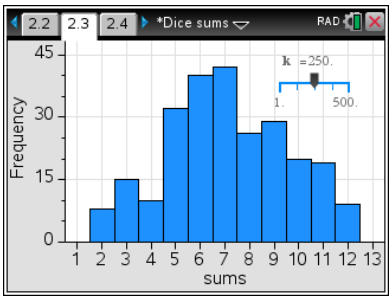


13. Change the plot type to a histogram by selecting **menu > Plot Type > Histogram**.



14. Now activate the slider to generate more (or less) sums.
    - As ***k*** increases, you will have to adjust the viewing window by selecting **menu > Window/Zoom > Zoom Data**.

    Which sum is the most common? Which sums are the rarest? Why?



Teacher Tip: Here's the teachable moment: Why does 7 tend to appear more often than other sums? Because when you toss a pair of dice there are six possible ways of rolling a 7. But there's only one way of rolling a 2 (or a 12). The simulation as described here generates a new set of data each time you change the value of k. A more realistic simulation would simply add another toss to the existing dataset when you increase the slider and remove the last toss when you decrease the slider. This is left as a challenge. See **augment** and **left** to add to and remove elements to/from a list.