Cadette Coding for Good, Badge 1:
meme, functions, pseudocode, shareable

1. Learn about functions and arguments.
2. Explore how memes are created.
3. Write pseudocode for a meme.
4. Write shareable code.
5. Share your meme.

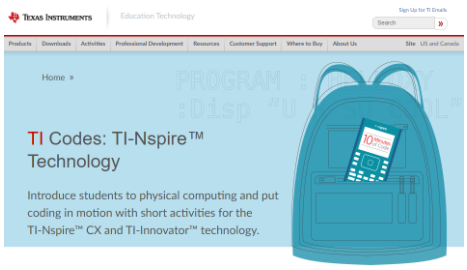
Be sure to review the "Cadette Coding for Good" guide for the badge requirements and badge steps provided by your leader and the Girl Scouts. It also includes relevant vocabulary and interesting background information to spark your interest in coding. This lesson will allow you to earn the **Badge 1: Coding Basics,** using your TI-Nspire™ CX II graphing calculator.

Introduction

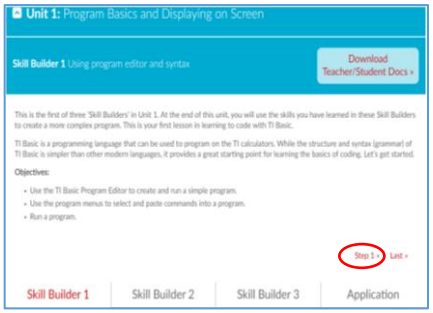
The programming language **BASIC** (stands for **B**eginner's **A**ll-purpose **S**ymbolic **I**nstruction **C**ode) was developed in the 1960s as an easy system for teaching computer programming. TI-Basic is similar to other flavors of BASIC, but you must select the programming words and commands from the onboard menus, as you will soon see.

Note: The DRAW features used in this project require a TI-Nspire™ CX II graphing calculator.

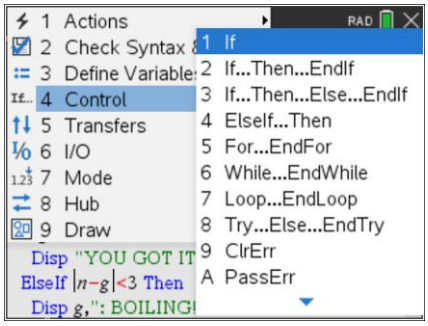
1. For an introduction to programming on the TI-Nspire™ CX II graphing calculator see the TI Codes lessons at **education.ti.com > Activities > TI Codes > TI-Basic**. Units 1 through 4, and Unit 6 (Drawing), should be enough to get started. Then return to this lesson for the **Cadette Coding, Badge 1** project. Go to TI Codes (for TI-Nspire™ technology).



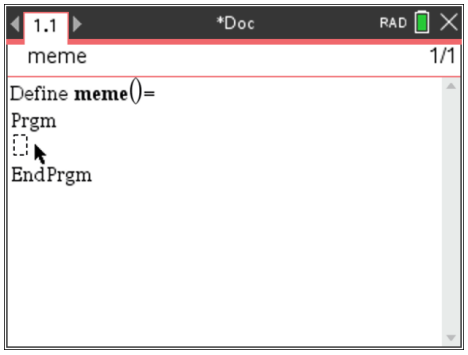
How to navigate TI Codes: Be sure to notice that each unit has three Skill Builders (SB) and one Application activity, as you can see here. You will navigate through each Skill Builder by clicking on the "Step 1" in the right bottom corner, which will take you through the content. This screenshot shows you are currently in Skill Builder 1 > Step 1. After you complete all the steps in SB 1, you will then move to SB 2, and so on, until you complete all of Unit 1. You will then move on to Units 2, 3, 4 … and do the same thing.



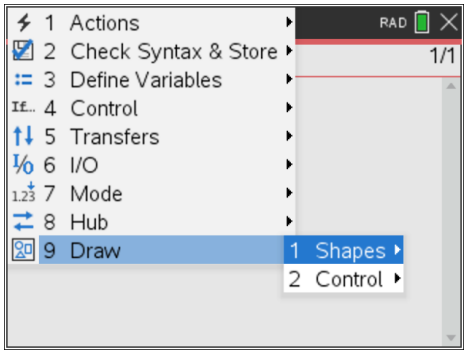
2. After you have completed the TI Codes units, you know about:
   - The programming process (planning, coding, testing and debugging)
   - Using the TI-Nspire™ Basic Program Editor and its menus
   - Running a program
   - Some important pieces of TI-Basic code: Input, Disp, variables and assignment statements, **If…Then…Else** structures and loops, like **For** and **While** (some of these statements are shown to the right)
   - **Draw** features in the programming world of **TI-Nspire™ CX II**



3. To start a new program:
   - From the Home screen, select **New** to start a new document
   - From the menu shown select **Add Program Editor**
   - Type the name of the program: ***meme***
   - Press **[enter]**



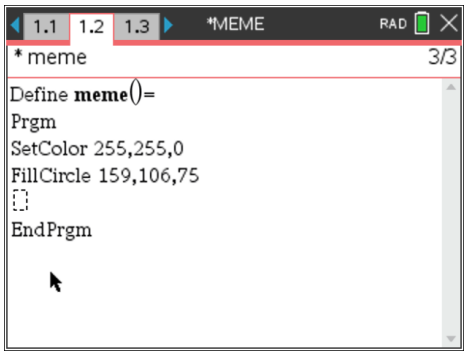
4. When designing a screen for graphics display, the drawing commands are all found on **[menu] > [draw]**.



5. When designing a screen for graphics display, the setup commands like **SetColor** and **SetPen** are found on **[menu] > [draw] > [Control]**.

   The shapes , like **FillCircle** are on **[menu] >Draw > Shapes**.

   The code to the right (note the numbers entered) draws a large yellow dot when you run the program by pressing **ctrl-R** and press **[enter]**.



   Examples: **SetColor** *red_value, blue_value, green_value*
   values between 0 and 255
   FillCircle x-center, y-center, radius
   values depend on window setting

   Does your image remind you of anything?

   For a refresher on more **Draw** tools, see Unit 6 of the TI-Nspire™ CX II technology lessons at TI Codes.

6. To draw text (words) on the graphics screen use

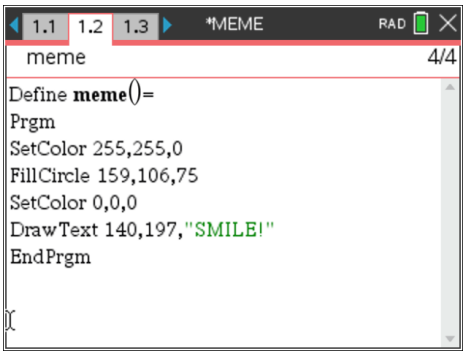
DrawText x, y, "*message*"

Found on **[menu] > Draw > Shapes**.

If you use the *two* commands (**SetColor** 0,0,0 is black) in the image to the right, you will see the *black* word

SMILE!

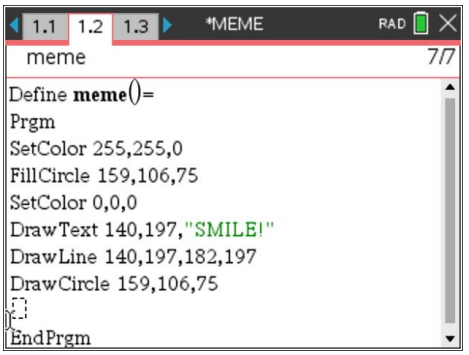
starting at the pixel in column x=140, row y=197 of the graphics canvas when you run the program.



Recall that the screen is 318 pixels wide and 212 pixels high and that the default window has (0,0) in the upper left corner of the screen. Change the numbers in the DrawText command to see where the message is drawn.

7. Other Draw commands are on **[menu] > Draw > Shapes**:
   - **DrawLine** a, b, c, d draws a segment between the points (a,b) and (c,d)
   - **DrawCircle** a, b, r draws a circle with center at (a,b) and a radius r

Both of these commands depend on the current window setting for the coordinates. Other commands on the **Draw** menu have similar arguments.

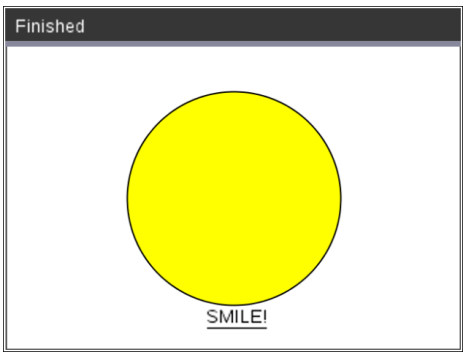


8. If you have entered the code presented so far, and run the program, you will see the screen shown to the right. Note the **SetColor** controls as well.

When designing your meme screen, you will be adjusting the numbers in these command a lot to get things just right. The numbers used here,

DrawLine 140, 197, 182, 197

were found by "guess and check" to get the word SMILE! underlined just right.

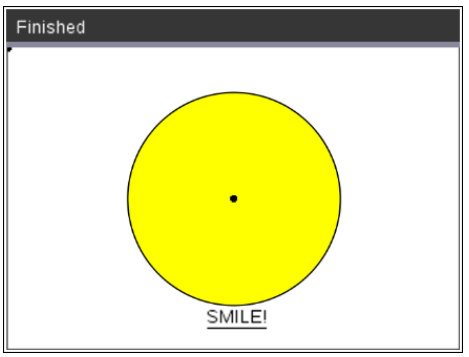


9. Individual points can be plotted using the command

PlotXY *x, y, style*

PlotXY 0, 0, 1 makes a dot in the upper left corner of the screen. It may be hard to see, so try some other coordinates.

PlotXY 159, 106, 1 makes a dot in the center of the screen.

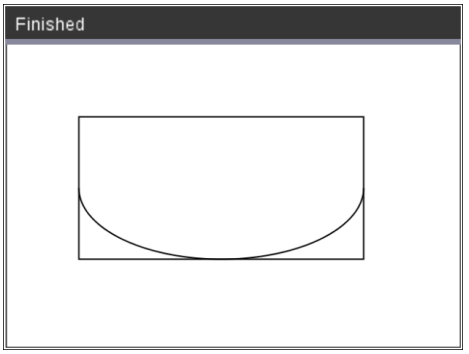


The third value (1 in these examples) is the *style* of the point and can be a number from one to 13. Try them all.

10. **DrawArc** and **FillArc** are a little trickier. We first make a rectangle (surrounding the arc) and you will have an idea:

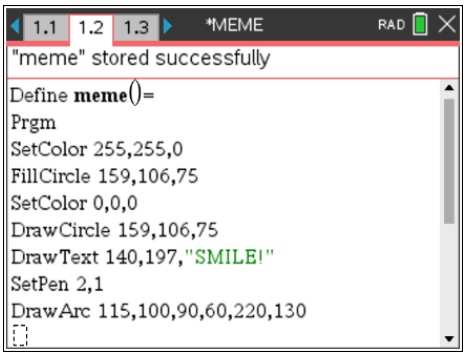
    DrawRect 50, 50, 200, 100
    DrawArc 50, 50, 200, 100, 180, 180

    The arc is *inscribed* in the rectangle, just touching each side. The last two values (180, 180) are the *StartAngle* and the *ArcAngle* to produce the lower portion of the ellipse shown. It starts at 180 degrees ("west") and goes another 180 degrees counterclockwise. A complete circle or ellipse can have any *StartAngle* and should have an *ArcAngle* of 360 degrees. You can draw the arc without drawing the rectangle. Try other values for *StartAngle* and *ArcAngle*.



11. You can draw the arc without drawing the rectangle. We used **SetPen 2,1** to make a thick arc.

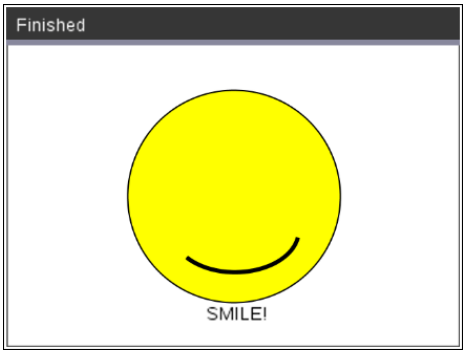
    Try your own values for the bounding rectangle, the *StartAngle* and the *ArcAngle*. Positioning the arc just right is also a matter of guessing well!



12. How about that "Mona Lisa smile"?

    Our smiley face meme needs eyes and a better message. That will be left up to you using two **FillArc** commands.

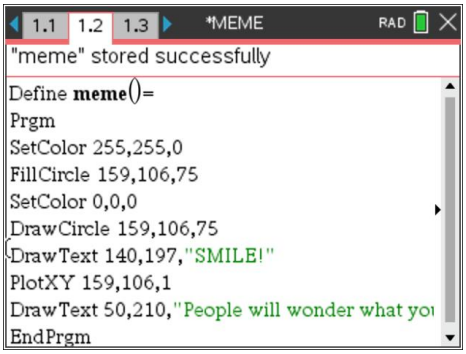
    Or … create your own meme!

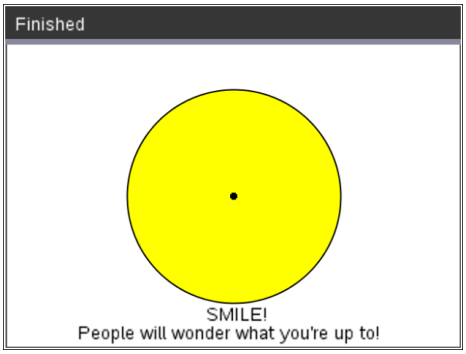


13. **Share your project.**

    You can take a picture of your calculator screen to share with the world! Share it online, and tag it using **@TICalculators**.

14. Here is a sample TI-Nspire™ CX II graphing calculator TI-Basic program and its result:





15. With this start you can complete the smiley face display or create a meme of your own design.

Congratulations!

You have completed the requirements for earning your **Cadette Coding for Good, Badge 1: Coding Basics**. Now that you have completed this requirement, you are challenged to *give service* by *sharing* what you have learned about coding with others. Refer to the Coding for Good Girl Scout guide for suggestions on how to do so.

©2021 Texas Instruments Incorporated

What's next? (Optional extensions)

Ready to try some additional practices with your new skills?

1. With the skills you learned from TI Codes, and these additional ideas, you can probably figure out how to write a program to draw the smiley face, the original emoji (or an emoji of your own design).

2. After completing Unit 6 of the TI Codes lessons, you might be ready to create an animation on your TI-Nspire™ CX II graphing calculator. Use the **Clear** command on the **Draw > Control** menu to erase the screen, and use a loop to draw objects in different locations on the screen.

3. See the **Basketball Game** in the **Beyond Basics** section of TI Codes. These projects are designed using TI-Basic on a TI-84 Plus family graphing calculator, but since the languages are so similar you should be able to "translate" them to TI-Basic on the TI-Nspire™ CX II graphing calculator.

Be sure to share your work on social media and tag it **@TICalculators**!