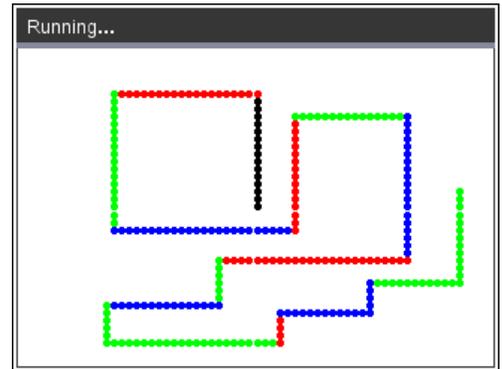




The most versatile tool in the TI-Nspire CX II Python library is the `get_key()` function. In this activity you will make use of this function to make an interactive 'Etch-A-Sketch'® style drawing program.

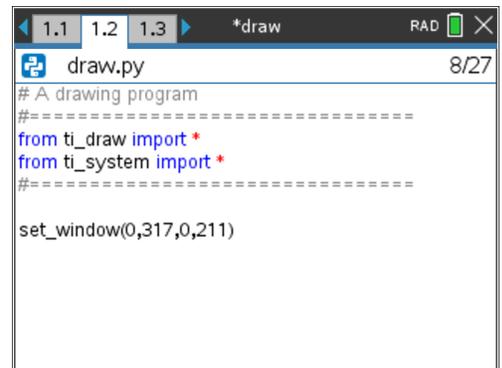
Introduction: Using the `get_key()` function found in the `ti_system` module it is possible to make dynamic, *interactive* graphical applications (and games!). In this activity, you will build a simple drawing program that begins by using the four arrow keys and can be extended to make use of other keys for changing color, points style, making 'stamps' or other features.



1. Begin a new program using the **Geometry Graphics** template and import the `ti_system` module to use the `get_key()` function.

The first statement sets up a comfortable canvas coordinate system with (0,0) in the lower left corner and each pixel is one unit.

`set_window(0, 317, 0, 211)`



2. Start in the center of the screen. Assign 159 to the variable `x` and 105 to the variable `y`. This can be done in a single line:

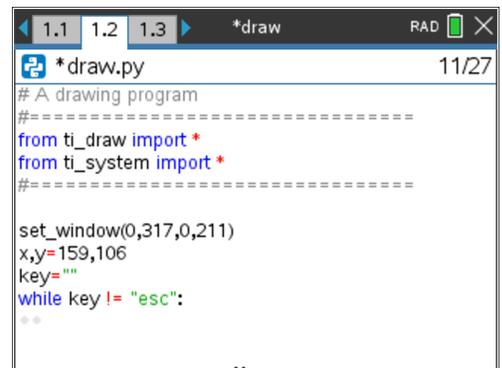
`x, y = 159, 105`

We usually use the statement `while get_key() != "esc"`: but in this project we need to *test* the value of `get_key()` to see which key is pressed and act accordingly, so we will assign `get_key()` to a variable named `key`. First assign an empty string to the variable `key`:

`key = ""` # nothing in the quotes

Then write the `while` loop that terminates when `key` is `"esc"`.

`while key != "esc":`





10 MOC: Python Modules

TI-NSPIRE™ CX II PYTHON

TI DRAW: INTERACTIVE ART

- In the loop body, plot the point (x, y) using the `ti_draw` function

plot_xy(x, y, 1)

Now assign the variable `key` the result of the `get_key()` function:

key = get_key()

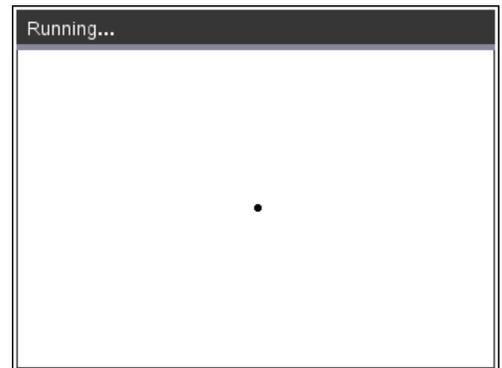
Note that `get_key()` can also be written `get_key(0)`.

- You can run the program now to see a dot in the center of the screen. Press **[esc]** to end the program.

```

1.1 1.2 1.3 *draw RAD 13/22
#-----
from ti_draw import *
from ti_system import *
#-----

set_window(0,317,0,211)
x,y=159,106
key=""
while key != "esc":
    plot_xy(x,y,1)
    key=get_key()
    |
  
```



- Check the value of `key` and cause the point to be repositioned at a new location depending on the key pressed. The four arrow keys on the keypad are called “up”, “down”, “left”, and “right” and each one will impact either the variable `x` or the variable `y`.

The first of four `if` statements is:

if key == “right”:
x += 1

You could also write `x = x + 1` (or another number)

Write the other three `if` statements.

```

1.1 1.2 1.3 *draw RAD 16/24
#-----
set_window(0,317,0,211)
x,y=159,106
key=""
while key != "esc":
    plot_xy(x,y,1)
    key=get_key()
    |
    if key=="right":
        x+=1
    |
    |
    |
  
```

- Did you write these?

if key == "left":
x -= 1
if key=="up":
y += 1
if key=="down":
y -= 1

```

1.1 1.2 1.3 *draw RAD 22/22
#-----
plot_xy(x,y,1)
key=get_key()
|
|
|
if key=="right":
    x+=1
if key=="left":
    x-=1
if key=="up":
    y+=1
if key=="down":
    y-=1
|
  
```



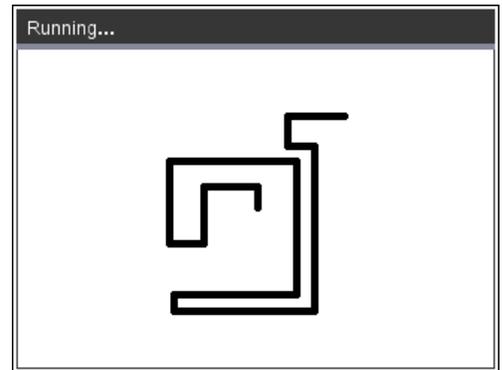
10 MOC: Python Modules

TI-NSPIRE™ CX II PYTHON

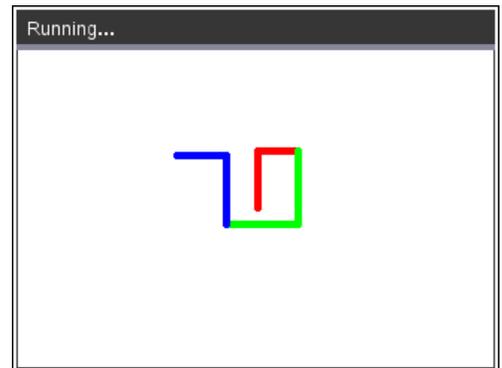
TI DRAW: INTERACTIVE ART

7. **Test your program now.** When you see the dot, use the four arrow keys to draw.

Again, press **[esc]** to end the program.



8. Almost all keys on the keypad return a value like the arrow keys. All values are "strings". For example, the **[tab]** key gives the string "tab"; and the **[3]** key becomes "3" in your code. The letter keys give lowercase letters. **[shift]** and **[ctrl]** do not give values because they are *modifier* keys that change the behaviors of some of the other keys.

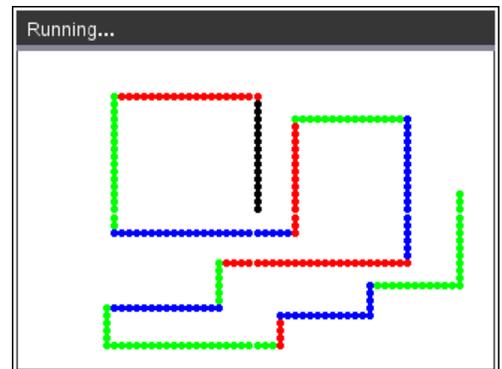


9. You can use other keys to add features to your program. Many other features are possible. Just add an if statement to your program to incorporate the feature. Some possibilities:

- use "r", "g", and "b" and the **set_color(, ,)** function to change the drawing color to red, green, or blue.
- Use the "c" key to **clear()** the screen
- Use the "e" key to switch to erase mode (set the color to white).
- But you will need some other key to return to black. Your choice
- Add a key to make a stamp ("s"?). the stamp can be a polygon or a circle.
- Add a key to perform some random behavior (computer art?).

Add a Notes page in front of your program to explain all the keystrokes that your program uses.

Change the increment values in your code to make the point move further in each keypress as seen here.





10 MOC: Python Modules

TI-NSPIRE™ CX II PYTHON

10. Your drawing point *might* go off the screen. How should you handle that? There are two common options:
- *Stop* at the edge of the screen (like a wall)
 - *Wrap* around to the opposite side of the screen (called 'toral' mode).

“Etch-A-Sketch” is a registered trademark of **SPIN MASTER LTD.**

TI DRAW: INTERACTIVE ART

