

QUICK START GUIDE

The TI-Innovator™ Rover and TI-84 Plus CE Python Graphing Calculator



Begin a new Python program:

- » Press **[zoom]** to access the **Type** softkey to select the **Rover** template and enter the program name.
(Note: The name of this program is “Rover1.”)
- » Press **[graph]** to select the **OK** softkey.

```
FILE MANAGER
NEW PROGRAM
Name=ROVER1

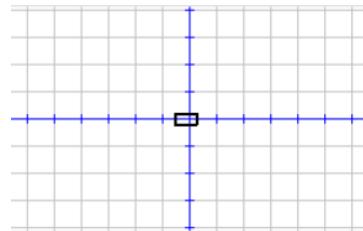
Allowed
- Up to 8 characters
- First character: A-Z
- Remaining characters: A-Z 0-9

Rover
Esc Types Ok

EDITOR: ROVER1
PROGRAM LINE 0005
# Rover
from time import *
from ti_system import *
import ti_rover as rv
```

Rover and the Cartesian coordinate plane:

- » Rover has a built-in coordinate system just like a Cartesian graphing system.
- » When you **import ti_rover as rv**, Rover's position on the coordinate grid is set to (0,0) and its heading is 0 degrees (pointing toward the positive x-axis or east on a map).
- » The default coordinate and driving unit is 10 cm.



Turn on the on-screen coordinate plane:

- » Select the following:
 - » **[math]> 5: ti_plotLib > 1: import ti_plotlib as plt**
 - » **[math]> 5: ti_plotLib > 2: cls()**
 - » **[math]> 5: ti_plotLib > 3: grid(1,1,"dash")**
 - » **[math]> 5: ti_plotLib > 4: window(-10,10,-7,7)**
 - » **[math]> 5: ti_plotLib > 6: axes("on")**

```
EDITOR: ROVER1
ti.plotlib module
Setup Draw Properties
1:import ti_plotlib as plt
2:cls()           clear screen
3:grid(xscl,yscl,"style") 
4>window(xmin,xmax,ymin,ymax)
5:auto_window(xlist,ylist)
6:axes("mode")
7:labels(" xlabel"," ylabel",x,y)
8:title("title")
9:show_plot()    display>[clear]
Esc

EDITOR: ROVER1
PROGRAM LINE 0011
# Rover
from time import *
from ti_system import *
import ti_rover as rv
import ti_plotlib as plt
plt.cls()
plt.grid(1,1,"dash")
plt.window(-10,10,-7,7)
plt.axes("on")
plt.show_plot()
```

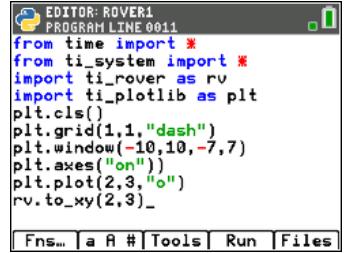
Make Rover drive to a point in the coordinate plane:

From the origin, move Rover to a point in the first quadrant on the calculator and the surface where Rover moves, such as (2,3):

- » Select [math] >5: ti_plotlib > Draw > 6: plot(x, y, "mark")
- » The mark selects how you want the point to appear on the calculator screen.
- » Enter the x and y coordinates as shown.
- » Select [math] > 7: ti_rover > 9: to_xy(x,y)
- » Enter the same coordinates as above.
- » Connect the calculator and Rover, then turn on the Rover.
- » Press [trace] to access the **Run** softkey and run the program.



```
EDITOR: ROVER1
ti.Plotlib module
Setup Draw Properties
1:color(r,g,b) 0-255
2:cls() clear screen
3:show_plot() display>[clear]
4:scatter(xlist,ylist,"mark")>
5:plot(xlist,ylist,"mark")>
6:plot(x,y,"mark")>
7:line(x1,y1,x2,y2,"mode")>
8:lin_reg(xlist,ylist,"disp")>
9:pen("size","style")>
0:text_at(row,"text","align")>
Esc
```



```
EDITOR: ROVER1
PROGRAM LINE 0011
from time import *
from ti_system import *
import ti_rover as rv
import ti_plotlib as plt
plt.cls()
plt.grid(1,1,"dash")
plt.window(-10,10,-7,7)
plt.axes("on")
plt.plot(2,3,"o")
rv.to_xy(2,3)
rv.to_xy(2,3)_
```

Make Rover create a reflection in the y-axis:

- » Press [enter] to create a new program row.
- » Select [math] > 5: ti_plotlib > Draw > 6: plot(x, y, "mark")
- » Enter the x and y coordinates which creates a reflection in the y-axis of the previously entered point.
- » Select [math] > 7: ti_rover > 9: to_xy(x,y)
- » Enter the same x and y coordinates as above.
- » Connect the calculator and Rover, then turn on Rover.
- » Press [trace] to access the **Run** softkey and run the program.
- » Press [trace] again to access the **Edit** softkey and edit the original program.



```
EDITOR: ROVER1
PROGRAM LINE 0013
from ti_system import *
import ti_rover as rv
import ti_plotlib as plt
plt.cls()
plt.grid(1,1,"dash")
plt.window(-10,10,-7,7)
plt.axes("on")
plt.plot(2,3,"o")
rv.to_xy(2,3)
plt.plot(-2,3,"o")
rv.to_xy(-2,3)
rv.to_xy(-2,3)_
```

Sample actions:

- » Move Rover to a point in each quadrant.
- » After plotting a point, create other transformations of the point such as:
 - » Reflection in the x-axis.
 - » Translation of x and y coordinates.