



Lesson 4: More Colour Geometry

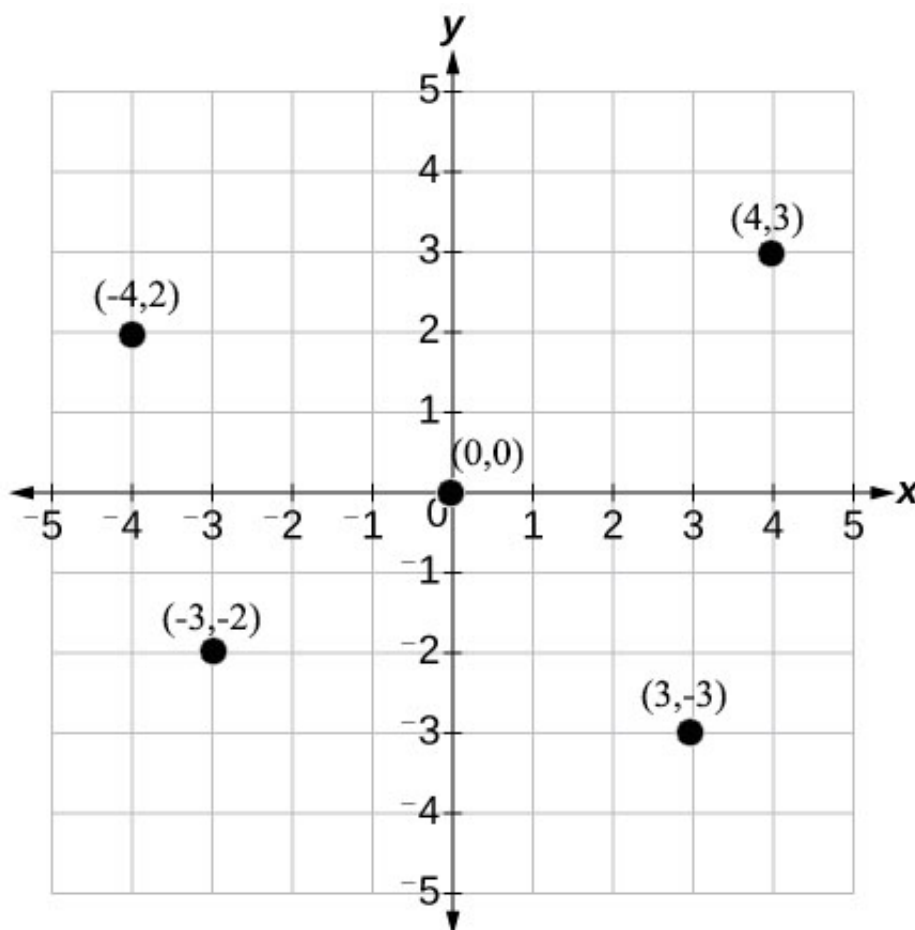
Summary:

Code Instruction	What it does
<code>t.goto(x,y)</code>	Send the turtle to the coordinates x and y
<code>t.setposition(x,y)</code>	Send the turtle to the coordinates x and y
<code>t.showturtle()</code>	Show the turtle
<code>t.hideturtle()</code>	Hide the turtle()
<code>t.stamp()</code>	Stamp a copy of the turtle shape onto the screen at the current turtle position
<code>t.speed('fastest')</code>	Set the fastest speed of the turtle
<code>T=turtle.Turtle('arrow')</code>	Set turtle shape as an arrow

To place the turtle on the turtle screen, we need to understand the x- and y-coordinate system used in our Turtle environment. X,Y coordinate system is drawn in Figure below. In the graph the dark horizontal line is called the *x-axis*, and it runs from left to right. The dark vertical line is the *y-axis*, running from bottom to top. We call the point (initial position of the turtle) where these lines meet, (0, 0), the *origin* because all other points on the grid are labeled with coordinates measured from, or *originating* from, that point. Think of the origin, (0, 0), as the center of your screen. Every other point you want to find can be labeled with an x- and y-coordinate by starting at the origin and moving left or right, down or up. We label points on a graph with this pair of coordinates inside parentheses, separated by a comma: (x, y). The first number, the x-coordinate, tells us how far to move left or right, while the second number, the y-coordinate, tells us how far to move up or down. Positive x-values tell us to move right from

the origin; negative x-values tell us to move left. Positive y-values tell us to move up from the origin, and negative y-values tell us to move down. Look at the points labelled in the Figure. The point in the upper right is labelled with the x and y-coordinates (4, 3). To find the location of this point, we start at the origin (0, 0) and move 4 spaces to the right (because the x-coordinate, 4, is positive) and then 3 spaces up (because the y-coordinate, 3, is positive). To get to the point in the lower right, (3, -3), we go back to the origin and then move right 3 spaces or units. This time, the y-coordinate is -3, so we move *down* 3 units. Moving right 3 and down 3 puts us at (3, -3). For (-4, 2), we move *left* 4 units from the origin and then up 2 units to the point in the upper left. Finally, for (-3, -2), we move left 3 units and then down 2 units to the lower-left point.

By default, turtle screen occupies 0.5 of full width and 0.75 of full height computer screen.





Python + Math

Code

Output

for Kids



```
import turtle
t=turtle.Turtle('arrow')
t.shapesize(5)
t.speed('fastest')

# Draw Arrow
t.up()
t.goto(60,70)
t.pensize(35)
t.down()
t.color('black')
t.setheading(90)
t.fd(180)
t.left(90)
t.fd(150)
t.showturtle()
t.stamp()
t.up()

#Draw Circle with cross Line
t.hideturtle()
t.color('red')
t.goto(0,0)
t.setheading(0)
t.pensize(50)
t.down()
t.circle(200)
t.circle(200,142)
t.left(90)
t.fd(400)
```



```
import turtle

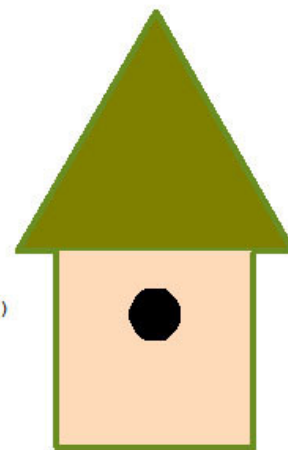
#Bird House
bird = turtle.Turtle()
bird.speed(10)
bird.pensize(4)
bird.color('oliveDrab','olive')
bird.setposition(-50, 0)

#Roof of the house
bird.begin_fill()
bird.forward(175)
bird.left(120)
bird.forward(175)
bird.left(120)
bird.forward(175)
bird.left(120)
bird.end_fill()

#House
bird.forward(25)
bird.right(90)
bird.fillcolor("peachpuff")
bird.begin_fill()
bird.fd(125)
bird.left(90)
bird.fd(125)
bird.left(90)
bird.fd(125)
bird.left(90)
bird.end_fill()

#Window
bird.penup()
bird.fd(62.5)
bird.left(90)
bird.forward(25)
bird.right(90)

bird.pendown()
bird.color('black','black')
bird.begin_fill()
bird.circle(15)
bird.end_fill()
bird.hideturtle()
```



LESSON 4

To see examples, images, and challenges

www.python.kidsgo.ca



1. Example #1 (Road Signs)

```
import turtle
t=turtle.Turtle('arrow')
t.shapesize(5)
t.speed('fastest')

# Draw Arrow
t.up()
t.goto(-60,70)
t.pensize(35)
t.down()
t.color('black')
t.setheading(90)
t.fd(180)
t.right(90)
t.fd(150)
t.showturtle()
t.stamp()
t.up()

#Draw Circle with cross Line
t.hideturtle()
t.color('red')
t.goto(0,0)
t.setheading(0)
t.pensize(50)
t.down()
t.circle(200)
t.circle(200,232)
t.left(90)
t.fd(400)
```



```
import turtle
t=turtle.Turtle('arrow')
t.shapesize(5)
t.speed('fastest')

# Draw Arrow
t.up()
t.goto(60,70)
t.pensize(35)
t.down()
t.color('black')
t.setheading(90)
t.fd(180)
t.left(90)
t.fd(150)
t.showturtle()
t.stamp()
t.up()

#Draw Circle with cross Line
t.hideturtle()
t.color('red')
t.goto(0,0)
t.setheading(0)
t.pensize(50)
t.down()
t.circle(200)
t.circle(200,142)
t.left(90)
t.fd(400)
```



2. Example #1 (Draw house)

```
import turtle

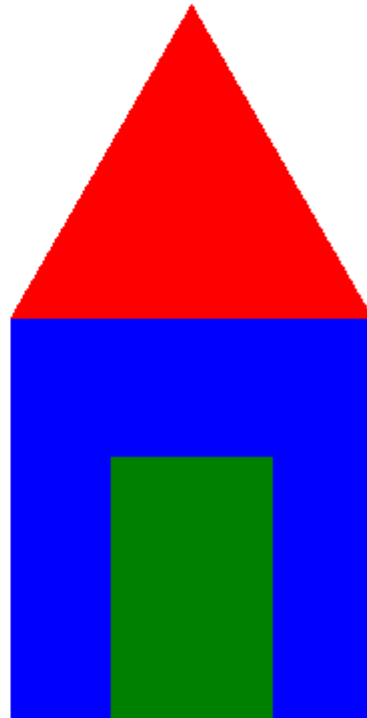
#Build a House

t1=turtle.Turtle()
t1.hideturtle()
t1.up()
t1.goto(-350,-200)
t1.setheading(0)
t1.down()
t1.color('blue','blue')
t1.begin_fill()
t1.fd(180)
t1.left(90)
t1.fd(200)
t1.left(90)
t1.fd(180)
t1.left(90)
t1.fd(200)
t1.left(90)
t1.end_fill()

#Door
t1.up()
t1.goto(-300,-200)
t1.setheading(0)
t1.down()
t1.color('green','green')
t1.begin_fill()
t1.fd(80)
t1.left(90)
t1.fd(130)
t1.left(90)
t1.fd(80)
t1.left(90)
t1.fd(130)
t1.left(90)
t1.end_fill()

#-----
#Roof
t1.up()
t1.goto(-350,0)
t1.setheading(0)
t1.down()
t1.color('red','red')
t1.begin_fill()
t1.fd(180)
t1.left(120)
t1.fd(180)
t1.left(120)
t1.fd(180)
t1.end_fill()

#-----
```



3. Example #2 (Tree)

```

import turtle
tl=turtle.Turtle()
tl.hideturtle()
#Crismas Tree

tl.up()
tl.goto(100,-200)
tl.setheading(0)
tl.down()
tl.color('brown','brown')
tl.begin_fill()
tl.fd(50)
tl.left(90)
tl.fd(100)
tl.left(90)
tl.fd(50)
tl.left(90)
tl.fd(100)
tl.end_fill()

tl.up()
tl.goto(50,-100)
tl.setheading(0)
tl.down()
tl.color('green','green')
tl.begin_fill()
tl.fd(150)
tl.left(120)
tl.fd(150)
tl.left(120)
tl.fd(150)

tl.up()
tl.goto(50,-50)
tl.setheading(0)
tl.down()
tl.fd(150)
tl.left(120)
tl.fd(150)
tl.left(120)
tl.fd(150)

tl.up()
tl.goto(50,0)
tl.setheading(0)
tl.down()
tl.fd(150)
tl.left(120)
tl.fd(150)
tl.left(120)
tl.fd(150)
tl.end_fill()

```



4. Example #2 (Bird House)

```
import turtle

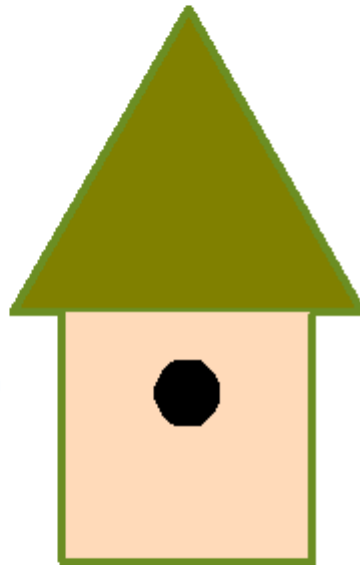
#Bird House
bird = turtle.Turtle()
bird.speed(10)
bird.pensize(4)
bird.color('oliveDrab','olive')
bird.setposition(-50, 0)

#Roof of the house
bird.begin_fill()
bird.forward(175)
bird.left(120)
bird.forward(175)
bird.left(120)
bird.forward(175)
bird.left(120)
bird.end_fill()

#House
bird.forward(25)
bird.right(90)
bird.fillcolor("peachpuff")
bird.begin_fill()
bird.fd(125)
bird.left(90)
bird.fd(125)
bird.left(90)
bird.fd(125)
bird.left(90)
bird.end_fill()

#Window
bird.penup()
bird.fd(62.5)
bird.left(90)
bird.forward(25)
bird.right(90)

bird.pendown()
bird.color('black','black')
bird.begin_fill()
bird.circle(15)
bird.end_fill()
bird.hideturtle()
```



5. Example #4(Traffic Light)

```
import turtle
t=turtle.Turtle()
t.hideturtle()

#Traffic Light
t.color('grey','grey')
t.up()
t.goto(-50,400)
t.down()
t.begin_fill()
t.fd(100)
t.right(90)
t.fd(300)
t.right(90)
t.fd(100)
t.right(90)
t.fd(300)
t.end_fill()

#Red traffic Light
t.penup()
t.goto(0,300)
t.setheading(0)
t.color('red','red')
t.begin_fill()
t.circle(50)
t.end_fill()

#Yellow Traffic Light
t.penup()
# it is possible to use t.up() instead t.penup()
t.goto(0,200)
t.color('yellow','yellow')
t.begin_fill()
t.circle(50)
t.end_fill()

#Green Traffic Light
t.penup()
# it is possible to use t.up() instead t.penup()
t.goto(0,100)
t.color('green','green')
t.begin_fill()
t.circle(50)
t.end_fill()
```



Challenges: write codes to create the following geometry shapes with circle code:

1. Expected output



2. Expected output



3. Expected output

