

**Bodgers**



**1. Python Programming is  
Perfect for Beginners.  
Simple uncomplicated syntax.  
Writing code in Python is  
similar to writing commands in  
plain English.**

```
print('Hello, world!')
```

```
#include  
  
int main()  
{  
    std::cout << "Hello, world!  
";  
    return 0;  
}
```



## 2. It's Easy to Set up and use.

We'll be using Mu, a Python code editor for beginner programmers

A screenshot of the Mu Python code editor interface. At the top is a toolbar with 16 icons in circular buttons: Mode (a mouse cursor), New (+), Load (upload), Save (download), Play (game controller), Images (brush), Fonts (A), Sounds (speaker), Music (musical note), Zoom-in (magnifying glass with plus), Zoom-out (magnifying glass with minus), Theme (moon), Check (thumbs up), Tidy (three horizontal lines), Help (?), and Quit (power button). Below the toolbar is a tab labeled "flappybird.py" with a close button (X). The main area shows a code editor with the following Python code:

```
1 import pgzrun
2 import random
3
4
5 TITLE = 'Flappy Bird'
6 WIDTH = 400
7 HEIGHT = 708
```



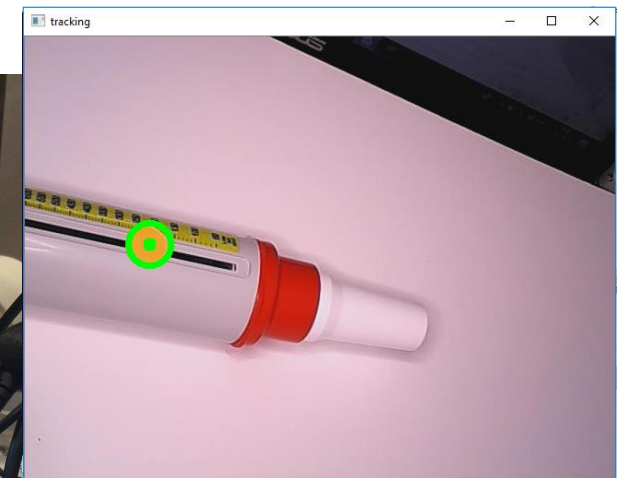
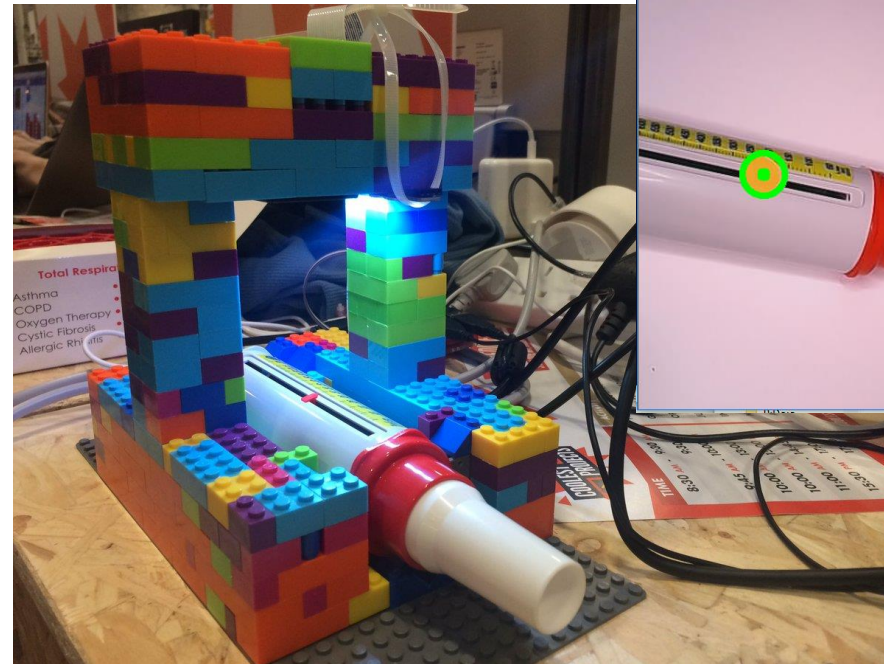
**3. Python is Kid-Friendly**  
With Python, you can transform your ideas into reality using loads of different technologies and online resources.





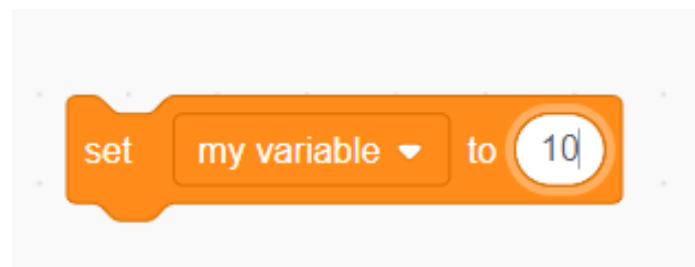
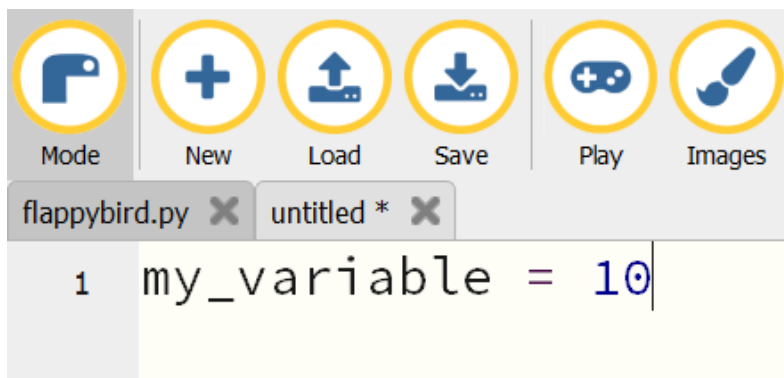
## 4. Python is popular and here to stay

Used for applications like Data Science, Machine Learning and Automation





Compare how things are done in Python Vs. Scratch

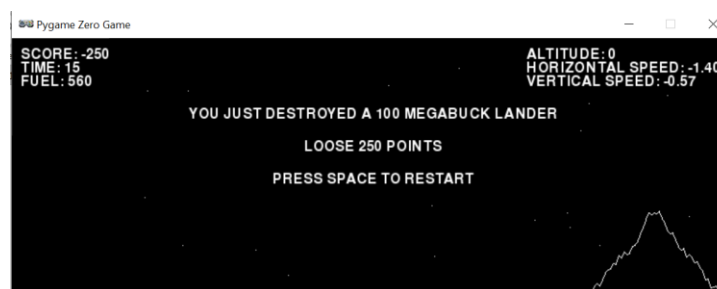
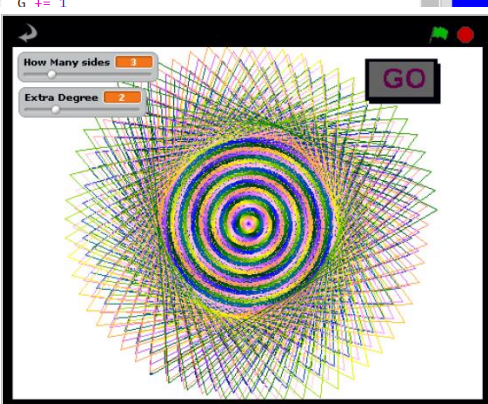
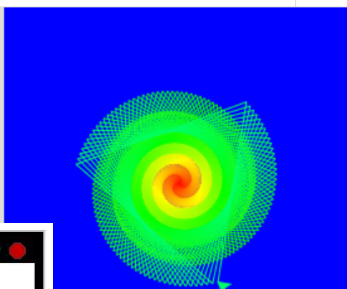




# Bodgers

Create Games & Animations in Python similar to those you created in Explorers & Advancers

```
main.py  
11 screen.bgcolor((0, 0, 255))  
12  
13 turtle.speed(0)  
14  
15 colours = []  
16  
17 while G <= 255:  
18     colours.append((R, G, B))  
19     G += 1  
20  
21  
22  
23  
24  
25  
26  
27
```



```
Pygame Zero Game  
SCORE: -250  
TIME: 15  
FUEL: 560  
ALTITUDE: 0  
HORIZONTAL SPEED: -1.40  
VERTICAL SPEED: -0.57  
YOU JUST DESTROYED A 100 MEGABUCK LANDER  
LOOSE 250 POINTS  
PRESS SPACE TO RESTART  
import pgzrun  
import random  
import colorsys  
from math import copysign  
WIDTH = 600  
HEIGHT = 400  
BALL_SIZE = 10  
MARGIN = 50  
BRICKS_X = 10  
BRICKS_Y = 5  
BRICK_W = (WIDTH - 2 * MARGIN) // BRICKS_X  
BRICK_H = 25  
ball = ZRect(WIDTH / 2, HEIGHT / 2, BALL_SIZE, BALL_SIZE)  
bat = ZRect(WIDTH / 2, HEIGHT - 50, 120, 12)  
bricks = []  
def hsv_color(h, s, v):  
    """Return an RGB color from HSV."""  
    r, g, b = colorsys.hsv_to_rgb(h, s, v)  
    return r * 255, g * 255, b * 255
```

