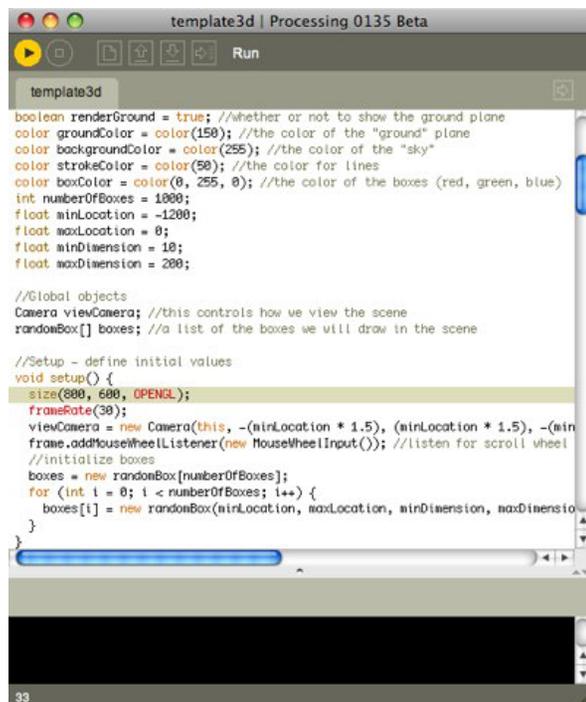


The interface for Processing is very simple but the capabilities for the program can be easy at times and very complex and difficult at others.

If you have had experience in writing code in such programs such as Java, Python and C++, then Processing is no different. If you are not familiar in writing code, think of it as writing a math equation. There are variables in the code like 'x' in a math equation for instance: $3x + 4 = 10$. You will have to write the code to include the variable 'x' and then decide its purpose. To make things more complex here is another equation: $(x + 1) + (2y * 6) = 20$. You must keep the parenthesis organized to keep the equation in order.

The Play button allows you run the code of the program so far for what you have written. The Stop button quits running the program and goes back to the interface.



To start, you must type 'void setup ()' which lets the program know that you are creating a new window to write the code. Typing size (300, 600); will determine the size of the canvas.

```
void setup () {
  size (300, 600);
}
```

Writing ' void draw () ' will create a canvas for you to write the code and draw.

```
void draw () {
}
```

After that, type in the bracket key ' { ', and once you are done typing the code of the program, type in the other bracket key ' } ' to close it. Keeping the code between those brackets allows the program to run. If you forget one or the other, then program will not run. Whenever you finish typing in the code, always press ' ; ' at the end of each sentence because it lets the program know that it is the end of sentence.

Typing ' background(0); ' determines the color of the window. 0 is the color black. The higher the number, the lighter the background.

```
void draw () {
  background (0);
}
```

Typing ' ellipse (0,0,0,0); ' will create a circle. The first two numbers determine the location of the ellipse on the canvas. The last two numbers will determine its size.

```
void draw () {
  background (0);
  ellipse ( 100, 100, 50, 50);
}
```