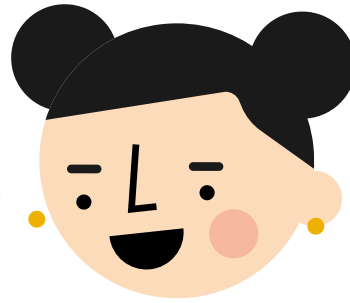# MAKE RANDOM SPARKLES ON THE SENSE HAT

For this activity, you can either use the physical Sense HAT hardware, the desktop emulator in Raspbian, or the web-based emulator on **trinket.io**

- If you're using the Sense HAT, attach it to your Raspberry Pi before booting.
- If you're using the Trinket emulator, open a web browser and go to **trinket.io/sense-hat**

Using set_pixel

First, we'll think up some random numbers and use the `set_pixel` function to place a random colour on a random location on the Sense HAT display.

1 If you're using a Raspberry Pi, open Python 3 and create a new file. If you're using the web emulator, delete the example code before you begin.

2 In the new file, start by importing the Sense HAT module.

If you're using a physical Sense HAT or the Trinket emulator, the import line is:

```
from sense_hat import SenseHat
```

If you're using the desktop emulator, the import line is:

```
from sense_emu import SenseHat
```

The rest of the code will be identical for all versions.

3 Next, create a connection to your Sense HAT by adding:

```
sense = SenseHat()
```

4 Now think of a random number between 0 and 7 and assign it to the variable `x`, for example:

```
x = 4
```

5 Think of another random number between 0 and 7, then assign it to `y`:

```
y = 5
```

6 Think of three random numbers between 0 and 255, then assign them to `r`, `g`, and `b`:

```
r = 19
g = 180
b = 230
```

7 Now use the set_pixel function to place your random colour at your random location on the display:

```
sense.set_pixel(x, y, r, g, b)
```

8 Now run your code by pressing **F5** (or the **Run** button in Trinket). You should see a single pixel light up.

9 Now pick a new set of random numbers, change them all, and run the program again. A second pixel should appear on the display!

Using the random module

So far you have picked your own random numbers, but you can let the computer choose them instead.

1 Add another `import` line at the top of your program, below `import SenseHat`:

```
from random import randint
```

2 Now change your `x =` and `y =` lines to automatically select a random position:

```
x = randint(0, 7)
y = randint(0, 7)
```

3 Run your program again, and you should see another random pixel being placed on the display. It will be the same colour as the last pixel you placed.

4 Now change your colour value lines to:

```
r = randint(0, 255)
g = randint(0, 255)
b = randint(0, 255)
```

Now your program will automatically select a random colour.

5 Run it again, and you should see another pixel appear in a random location with a random colour.

Run it a few more times, and you should see more of the grid fill up with random pixels.

Add a loop

Rather than having to keep running your program, you can add a loop so that it will keep going.

1 First, add an `import` to the top of your file:

```
from time import sleep
```

You'll use this to pause the program between pixels.

2 Add a `while True:` to your code so that the random lines, `set_pixel` and `sleep` are all within the loop:

```
while True:
    x = randint(0, 7)
    y = randint(0, 7)
    r = randint(0, 255)
    g = randint(0, 255)
    b = randint(0, 255)
    sense.set_pixel(x, y, r, g, b)
    sleep(0.1)
```

3 Run the code and you should see random sparkles in action!

4 Try changing the sleep time to make it even shorter.