Python on Trinket – Lesson 4

Review

- For loops
- Variables
- Random integers
- Angles of rotation

Discussion

- Functions
  - A block of organized, reusable code that is used to perform a certain action
  - User-defined functions: written by the programmer
    - Avoid re-writing multiple lines of code (for tasks you want to perform more than once)
    - Define code to perform some action within a function, and use a command to call that function every time you want that action to run.
    - Pass in function parameters to create flexible programs
      - Create varying outputs depending on values passed in
      - i.e. Snowflakes with various colours and shapes, by passing in different values for colours and lengths.

User-defined function	Function call

`def functionname(parameters):`
`  #function body`
`  return`
	`functionname(parameters)`

- Variables that change over time
  - Have many inputs without having to provide them manually

Lesson & practice

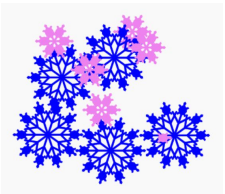
1a. Snowflake (user-defined function)

```
def snowflake():
  size = 10
  for count in range(size):
    t.color(random.randint(0, 255), random.randint(0, 255), random.randint(0, 255))
    t.forward(50)
    t.stamp()
    t.backward(20)
    t.left(45)
    t.forward(20)
    t.backward(20)
    t.right(90)
    t.forward(20)
    t.backward(20)
    t.left(45)
    t.backward(30)
    t.left(360/size)
  return
```

```
#function call:

snowflake()
```

1b. Shapes using user-defined functions with parameters

```
def snowflake_parameters(length, number_of_rays, colour):
  for count in range(number_of_rays):
    t.color(colour)
    t.forward(length)
    t.stamp()
    t.backward(length/3)
    t.left(45)
    t.forward(length/3)
    t.backward(length/3)
    t.right(90)
    t.forward(length/3)
    t.backward(length/3)
    t.left(45)
    t.backward(length*(2/3))
    t.left(360/number_of_rays)
  return

t.penup()
t.goto(0, -100)
t.pendown()

for count in range (5):
  snowflake3(40, 10, "blue")
  t.penup()
  t.goto(random.randint(-100, 100), random.randint(-100, 100))
  t.pendown()
  snowflake3(15, 6, "violet")
  t.penup()
```



```
  t.goto(random.randint(-100, 100), random.randint(-100, 100))
  t.pendown()
```

2. Variables that change over time (Snowflakes with varying RGB)

```
r = 0
g = 21
b = 255

n_snowflakes = 10
for count in range (n_snowflakes):
  snowflake_parameters(30, 10, (r, g, b))
  t.penup()
  t.goto(random.randint(-150, 150), random.randint(-150, 150))
  t.pendown()
  b = b - 25
  r = r + 25
```