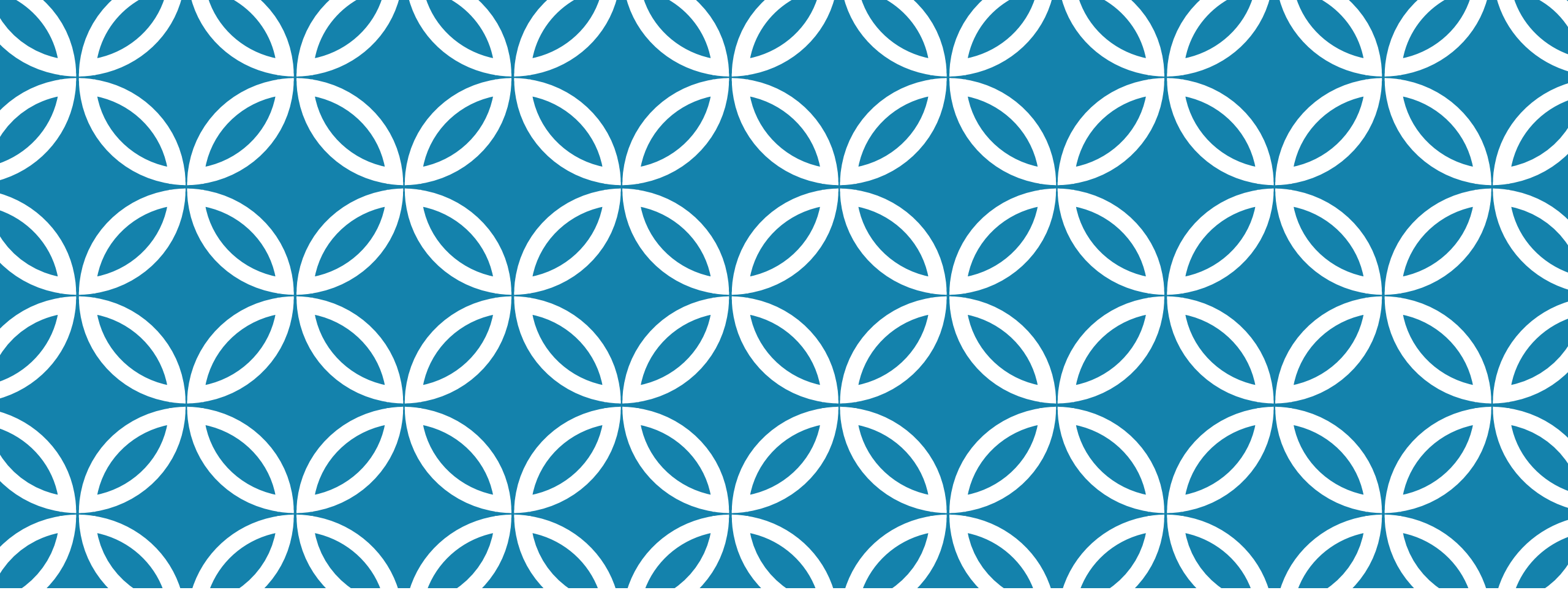
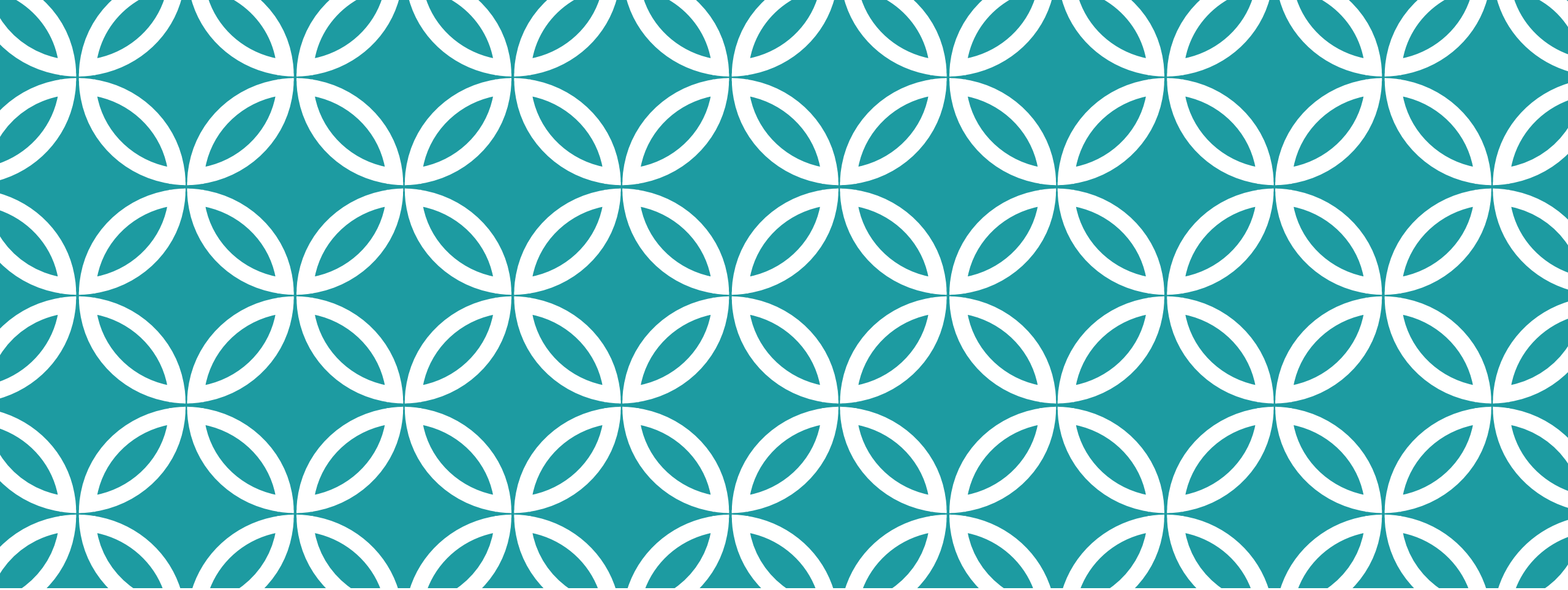


PYTHON

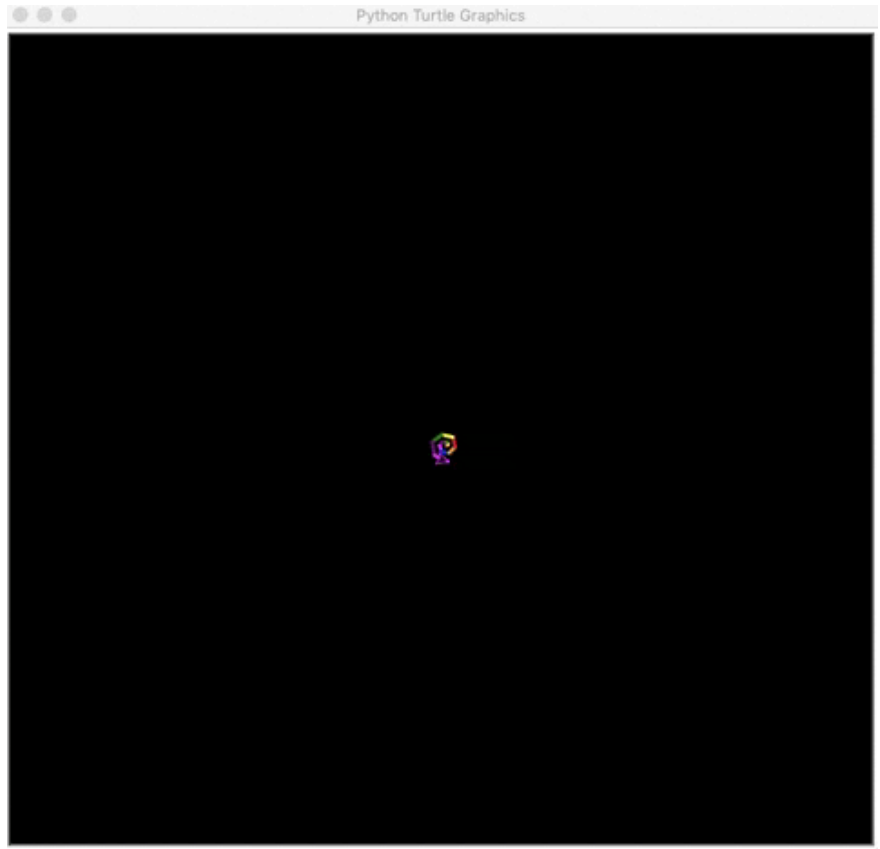


TURTLE

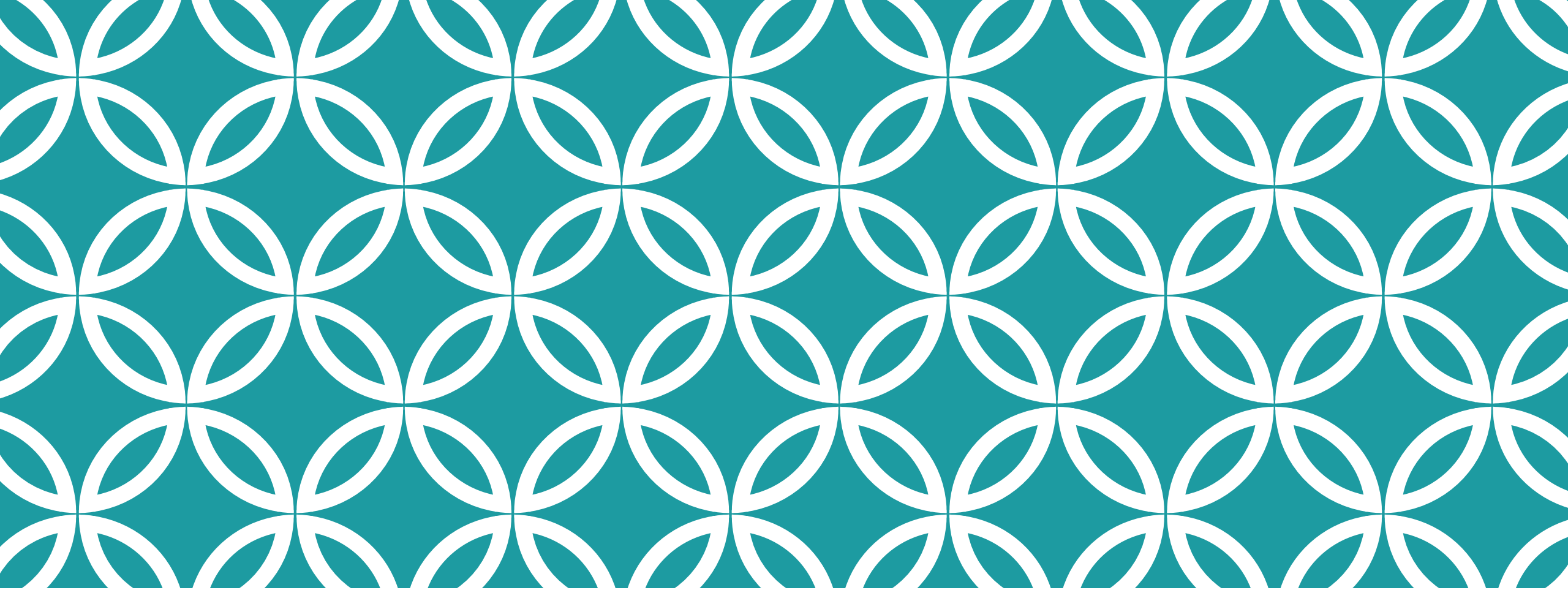
turtle is a pre-installed Python library that enables users to create pictures and shapes by providing them with a virtual canvas. The onscreen pen that you use for drawing is called the turtle and this is what gives the library its name. In short, the Python turtle library helps new programmers get a feel for what programming with Python is like in a fun and interactive way.

turtle is mainly used to introduce children to the world of computers. It's a straightforward yet versatile way to understand the concepts of Python. This makes it a great avenue for kids to take their first steps in Python programming. That being said, the Python turtle library is not restricted to little ones alone! It's also proved extremely useful for adults who are trying their hands at Python, which makes it great for Python beginners.

With the Python turtle library, you can draw and create various types of shapes and images. Here's a sample of the kinds of drawings you can make with turtle:


Python Turtle Graphics

Cool, right? This is just one of many different drawings you can make using the Python turtle library. Most developers use turtle to draw shapes, create designs, and make images. Others use turtle to create mini-games and animations, just like the one you saw above.



GETTING STARTED WITH TURTLE

Before you continue, there are two important things that you'll need to do to make the most of this tutorial:

1. **Python Environment:** Make sure that you're familiar with your programming environment. You can use applications like IDLE or Jupyter Notebook to program with turtle.
2. **Python Version:** Ensure that you have version 3 of Python on your computer. If not, then you can download it from the Python website.

First we import the turtle module. Then create a window, next we create turtle object and using turtle method we can draw in the drawing board.

METHOD	PARAMETER	DESCRIPTION
Turtle()	None	It creates and returns a new turtle object
forward()	amount	It moves the turtle forward by the specified amount
backward()	amount	It moves the turtle backward by the specified amount
right()	angle	It turns the turtle clockwise
left()	angle	It turns the turtle counter clockwise
penup()	None	It picks up the turtle's Pen
pendown()	None	Puts down the turtle's Pen
up()	None	Picks up the turtle's Pen
down()	None	Puts down the turtle's Pen
color()	Color name	Changes the color of the turtle's pen

fillcolor()	Color name	Changes the color of the turtle will use to fill a polygon
heading()	None	It returns the current heading
position()	None	It returns the current position
goto()	x, y	It moves the turtle to position x,y
begin_fill()	None	Remember the starting point for a filled polygon
end_fill()	None	It closes the polygon and fills with the current fill color
dot()	None	Leaves the dot at the current position
shape()	shapename	Should be ‘arrow’, ‘classic’, ‘turtle’ or ‘circle’
stamp()	None	Leaves an impression of a turtle shape at the current location

PROGRAM 1

```
# import turtle library
import turtle
my_window = turtle.Screen()
my_window.bgcolor("blue")       # creates a graphics window
my_pen = turtle.Turtle()
my_pen.forward(150)
my_pen.left(90)
my_pen.forward(75)
my_pen.color("white")
my_pen.pensize(12)
```

PROGRAM 2

```
# import turtle library
import turtle
my_pen = turtle.Turtle()
for i in range(4):
    my_pen.forward(50)
    my_pen.right(90)
turtle.done()
```

PROGRAM 3

```
# import turtle library
import turtle
my_pen = turtle.Turtle()
for i in range(50):
    my_pen.forward(50)
    my_pen.right(144)
turtle.done()
```

PROGRAM 4

```
# import turtle library
import turtle
polygon = turtle.Turtle()
my_num_sides = 6
my_side_length = 70
my_angle = 360.0 / my_num_sides
for i in range(my_num_sides):
    polygon.forward(my_side_length)
    polygon.right(my_angle)
turtle.done()
```

PROGRAM 5

```
# import turtle library
import turtle
my_wn = turtle.Screen()
my_wn.bgcolor("light blue")
my_wn.title("Turtle")
my_pen = turtle.Turtle()
my_pen.color("black")
def my_sqrfunc(size):
    for i in range(4):
        my_pen.fd(size)
```

```
        my_pen.left(90)
        size = size - 5
my_sqrfunc(146)
my_sqrfunc(126)
my_sqrfunc(106)
my_sqrfunc(86)
my_sqrfunc(66)
my_sqrfunc(46)
my_sqrfunc(26)
```

PROGRAM 6

```
# import turtle library
import turtle
my_wn = turtle.Screen()
turtle.speed(2)
for i in range(30):
    turtle.circle(5*i)
    turtle.circle(-5*i)
    turtle.left(i)
turtle.exitonclick()
```

PROGRAM 7

```
# import turtle library
import turtle
colors = [ "red","purple","blue","green","orange","yellow"]
my_pen = turtle.Pen()
turtle.bgcolor("black")
for x in range(360):
    my_pen.pencolor(colors[x % 6])
    my_pen.width(x/100 + 1)
    my_pen.forward(x)
    my_pen.left(59)
```

PROGRAM 8

```
from turtle import *
color('red', 'yellow')
begin_fill()
while True:
    forward(200)
    left(170)
    if abs(pos()) < 1:
        break
end_fill()
done()
```

PROGRAM 9

```
import turtle

spiral = turtle.Turtle()

for i in range(50):
    spiral.forward(i * 10)
    spiral.right(144)

turtle.done()
```

PROGRAM 10

```
import turtle
ninja = turtle.Turtle()
ninja.speed(10)
for i in range(180):
    ninja.forward(100)
    ninja.right(30)
    ninja.forward(20)
    ninja.left(60)
```

```
 ninja.forward(50)
    ninja.right(30)
    ninja.penup()
    ninja.setposition(0, 0)
    ninja.pendown()
    ninja.right(2)

turtle.done()
```

PROGRAM 11

```
from turtle import *

colors = ['orange', 'red', 'pink', 'yellow', 'blue', 'green'] for x in range(360):

pencolor(colors[x % 6])

width(x / 5 + 1)

forward(x)

left(20)
```

PROGRAM 12

```
from turtle import *
penup()
for a in range(40, -1, -1):
stamp()
left(a)
forward(20)
```

PROGRAM 13

```
from turtle import *


fillcolor('purple')
pensize(10)
pencolor('black')
forward(100)
```

```
begin_fill()
forward(100)
left(90)
forward(100)
left(90)
forward(100)
left(90)
forward(100)
left(90)
end_fill()
```

PROGRAM 14

```
from turtle import *

def up():
    setheading(90)
    forward(100)

def down():
    setheading(270)
    forward(100)
```

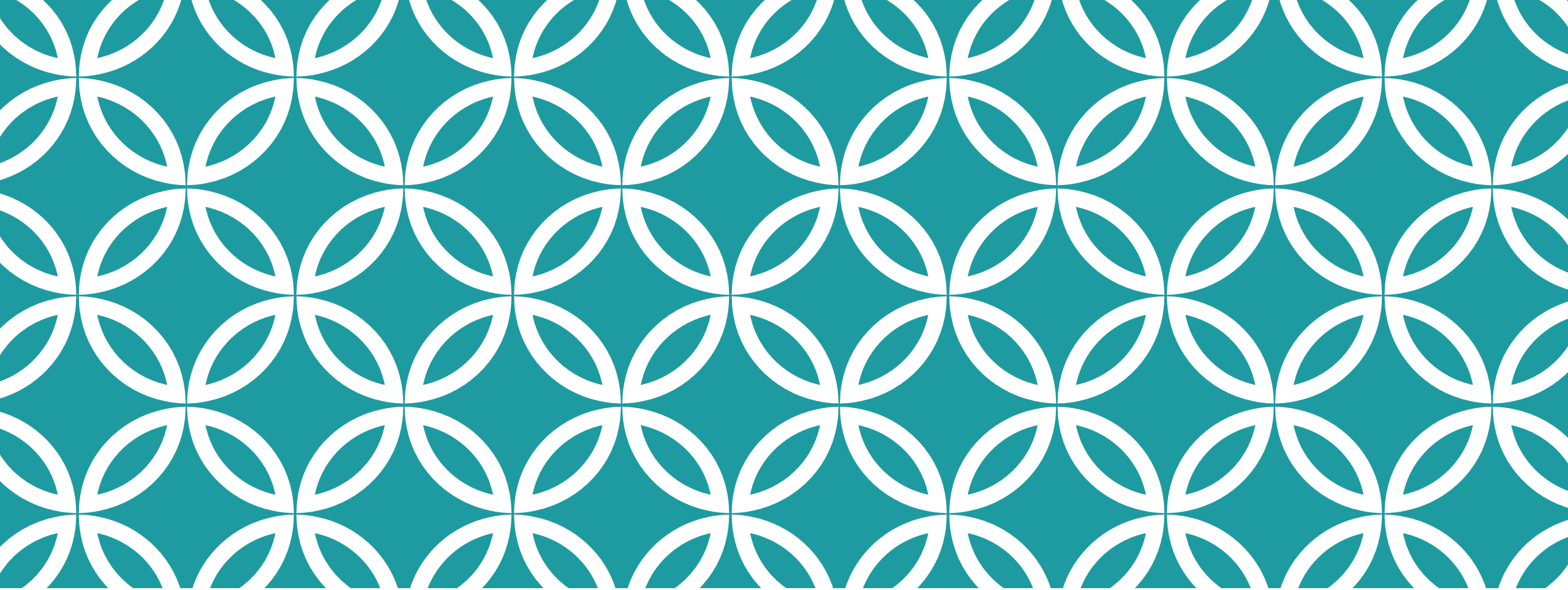
```
def left():
    setheading(180)
    forward(100)

def right():
    setheading(0)
    forward(100)

listen()
```

```
onkey(up, 'Up')
onkey(down, 'Down')
onkey(left, 'Left')
onkey(right, 'Right')

onkey(up, 'w')
onkey(down, 's')
onkey(left, 'a')
onkey(right, 'd')
```



CONTINUE IN NEXT UNIT .....