More Turtle Graphics, Event Listeners, State and Multiple Instances

Python Higher Order Functions and Event Listeners:

Functions as inputs: When we pass the functions as inputs, we only pass the name; without the brackets at the end.

```
from turtle import Turtle, Screen

tim = Turtle()
screen = Screen()

def move_forward():
    tim.forward(10)



screen.listen()
screen.onkey(key="space", fun=move_forward)
screen.exitonclick()
```

Notice the last 2 lines.
The idea of Higher Order Functions is a function that can work with functions.
In python, it is very commonly used and is helpful when we listen for events and then trigger a particular function.

Etch-A-Sketch App:
Key functions:
W= forwards
S= backwards
A= counter-clockwise/ left
D= clockwise/right
C= clear drawing and put the turtle back in the center.

```
from turtle import Turtle, Screen

tim = Turtle()
screen = Screen()

def move_forward():
    tim.forward(10)

def move_backward():
    tim.backward(10)

def turn_left():
    new_heading = tim.heading() + 10
```

```
    tim.setheading(new_heading)

def turn_right():
    new_heading = tim.heading() - 10
    tim.setheading(new_heading)

def clear():
    tim.clear()
    tim.penup()
    tim.home()
    tim.pendown()

screen.listen()
screen.onkey(key="w", fun=move_forward)
screen.onkey(key="s", fun=move_backward)
screen.onkey(key="a", fun=turn_left)
screen.onkey(key="d", fun=turn_right)
screen.onkey(key="c", fun=clear)

screen.exitonclick()
```

Object State and Instances:
We can create as many objects as we want.
Each object can have different attributes and can be performing different methods at a time.
In programming, this is known as their state.

Turtle coordinate System: