```
# -*- coding: utf-8 -*-
"""
Created on Wed Nov 12 18:30:24 2025

@author: Marty
"""

import turtle

#Create a turtle screen,  with a yellow background
wn=turtle.Screen()
wn.bgcolor("lightyellow")

#our turtle is named tess
tess = turtle.Turtle() # Create tess and set some attributes
tess.color("blue")
tess.pensize(5)

#got a lot of drawing ahead - let's speed us up some
tess.speed(0)

# Parallelogram to take a length & width of each side, and internal angle and
# draw a parallelogram
def parallelogram(length, width, angle):
    """parallelogram function takes in 3 arguments: length, width, & angle
    (L, W, & A), it is expecting a numerical for all, no error control. This
    function is a general application of 4-sided geometric objects within a 2D
    Cartesian plane."""
    for i in range(2):
        tess.forward(length)
        tess.left(angle)
        tess.forward(width)
        tess.left(180-angle)

def rectangle(length, width):
    """Inhereted attributes from Homework_4_11_3.parallelogram (P). Pass
    dimensional variables and set P.angle=90 to illustrate a rectangle"""
    parallelogram(length, width, 90)

def rhombus(length, angle):
    """Inhereted attributes from Homework_4_11_3.parallelogram (P). Pass
    dimensional variable and set user defined angle to illustrate a rhombus"""
    parallelogram(length, length, angle)

#tess.penup()
#tess.forward(120)
#tess.pendown()

parallelogram(100, 50, 90)

#hop to a different spot to draw the next shape
tess.penup()
tess.forward(120)
tess.pendown()

#draw a rectangle
rectangle(50, 20)
```

```
#hop to a different spot to draw the next shape
tess.penup()
tess.forward(120)
tess.pendown()

#draw a rhombus
rhombus(200, 65)

# stop drawing
turtle.done()
turtle.bye()

#close the turtle window after viewing it and before running another turtle
# program
```