Homework 1

1. Draw a Triangle

Task : Write a Python program (using the Turtle library) to draw an **equilateral triangle** .

Requirements :

Each side of the triangle should be **100 units** long.

Use loops or repeated commands as needed.

2. Modify the Pentagon

Task : Starting from the idea of drawing a pentagon, **modify** the logic to draw a **hexagon** (6 sides).

Requirements :

Each side of the hexagon should be **80 units** long.

You may use a function to draw the hexagon or simply write a loop that repeats the drawing commands.

3. Draw and Fill a Rectangle

Task : Write a Python program to draw a **rectangle** .

Requirements :

The rectangle should have a **width of 150 units** and a **height of 75 units** .

Fill the rectangle with the color **blue** (use `begin_fill()`, `fillcolor()`, and `end_fill()`).

4. Draw and Fill a Star

Task : Write a Python program to draw a **5-point star** .

Requirements :

Each side of the star should be **100 units** long.

Fill the star with the color **red** .

Consider the turning angle (144°) for a typical 5-point star.

5. Create a Circle

Task : Modify or create a new program to draw a **circle** using Turtle.

Requirements :

The circle should have a **radius of 100 units** .

Use the `circle(radius)` method from the Turtle library.

Additional Tips

Structure Your Code : Remember to import the Turtle library and create a Turtle object before drawing.

Use Loops : For repeating patterns (like the sides of a shape), loops are very efficient.

Angles Matter : For regular polygons, the external angle is `360 / number_of_sides`.

Pen Control : Use `penup()` and `pendown()` to move the turtle without drawing if you need to change positions.

Colorful Creations : Make sure to use `begin_fill()`, `fillcolor("color")`, and `end_fill()` when filling shapes.

Documentation and Resources :

Python's official Turtle documentation: https://docs.python.org/3/library/turtle.html

Lesson notes and examples from class.

Collaboration and Help

You are encouraged to discuss approaches with classmates, but make sure the final work is your own.
If you get stuck, ask your instructor or classmates for guidance.

Submission

Make sure each shape or drawing is clear.
Place each exercise's code in a separate script file or clearly separate each task in one script.
Follow your instructor's guidelines for how to submit your work (e.g., upload to a shared folder, email, or LMS).

Good luck, and have fun experimenting with Turtle Graphics!