University Mostafa Benboulaid Batna 2
Faculty of Mathematics and Computer Science
Computer Science Department

Practical Work 01
Beginning Python

Part A – "Area and perimeter calculator project"

Tasks:

1. **Basic Input/Output**
   - Ask the user to choose a shape: "circle", "rectangle", "square".
   - Ask for the required dimensions.
2. **Functions for Calculations**
   - Write a function to calculate the **area** and the **perimeter** for each shaper.

Circle		Rectangle		Square	
Area	Perimeter	Area	Perimeter	Area	Perimeter
$A = \pi r^2$	$C = 2\pi r$	$A = L \times W$	$P = 2(l + w)$	$A = s^2$	$P = 4s$

3. **Control Flow & Data Structures**
   - Use (**if**/**elif**/**else**) to decide which shape to compute.
   - Store each result in a **dictionary**:
   - {"shape": "circle", "radius": 5, "area": 78.5, "perimeter": 31.4}
   - Keep all results inside a **list** called ***calculations***.
4. **Loops**
   - Allow the user to perform multiple calculations until they type "exit".
5. **File Handling**
   - Save each result to a file named "**results.txt**".
   - Each line should contain the shape and its results.
   - Example of a result: " ***{'shape': 'Circle (radius=20.0)', 'area': 1256.6370614359173, 'perimeter': 125.66370614359172}*** "

Part B – Object-Oriented Version

Tasks:

1. **Base Class**
    - Create a class ***Shape*** with methods:
        - **area()**
        - **perimeter()**
        - **__str__()** (returns "Generic Shape")
2. **Circle Class**
    - Inherit from Shape.
    - Constructor takes radius.
    - Implement **area()** and perimeter().
    - Override **__str__()** to return "Circle (radius=5)".
3. **Rectangle Class**
    - Constructor takes length and width.
    - Implement **area()** and perimeter().
    - Override **__str__()** to return "Rectangle (length=4, width=6)".
4. **Square Class**
    - Inherit from ***Rectangle***.
    - Constructor takes side.
    - Override **__str__()** to return "Square (side=4)".
5. **Main Program Refactor**
    - Instead of calling functions, create **objects** (Circle, Rectangle, Square).
    - Call **shape.area()** and **shape.perimeter()** on them.
    - Store results in the same **dictionary + list** format as before.
    - Save results to file just like in **Part A**.