ROBOTICS CORNER

Task(15.2)

Instructions:

1. **Define the Shape Interface:**
   - Create an abstract class named Shape with a pure virtual function draw(). This function should be responsible for displaying the type of shape being drawn.
2. **Implement Concrete Shape Classes:**
   - Create three classes (Circle, Rectangle, and Triangle) that inherit from the Shape class.
   - Each class should override the draw() method to output a message indicating the specific shape being drawn.
3. **Create a Factory Function:**
   - Implement a standalone function named createShape that takes a std::string parameter, shapeType.
   - The function should return a pointer to a Shape object based on the shapeType provided. If the shape type is unknown, it should return nullptr.
4. **Client Code:**
   - In the main function, use the createShape function to create instances of the shapes.
   - Call the draw() method for each shape created to display the output.
   - Ensure proper memory management by deleting the shape objects after use.
5. **Test Different Shape Types:**
   - Verify the functionality by testing the creation of all three shape types: Circle, Rectangle, and Triangle.
   - Observe the output produced by the draw() method for each shape.
6. **Handle Unknown Shape Types:**
   - Modify the createShape function to handle unknown shape types gracefully, ensuring it returns nullptr and does not create any shape object.

Thank You