1. The Shape Shifter (Property Pattern)

Goal: Calculate the area of different shapes using property patterns.

Task: Write a method double GetArea(object shape) that uses a switch expression.

- If it's a Rectangle with properties Width and Height, return $Width \times Height$.
- If it's a Circle with property Radius, return pi r^2.
- If it's a Square with property Side, return Side^2.
- Handle null or unknown shapes by returning 0.

2. The Cinema Usher (Relational & Logical Patterns)

Goal: Determine ticket pricing based on age using relational operators (<, >, <=, >=) and logical patterns (and, or).

Task: Create a method string GetTicketCategory(int age).

- 0-2: "Infant (Free)"
- 3-12: "Child"
- 13-19: "Teenager"
- 20-64: "Adult"
- 65+: "Senior"
- Negative numbers: Throw an ArgumentException.

3. The Delivery Logistics (Tuple Pattern)

Goal: Determine shipping costs based on a combination of destination and weight.

Task: Create a method decimal CalculateShipping(string zone, double weight). Switch on a **tuple** (zone, weight).

- ("International", > 10): $50.00
- ("International", <= 10): $25.00
- ("Domestic", > 5): $10.00
- ("Domestic", <= 5): $5.00
- Everything else: $0.00

4. The Smart Thermostat (Positional Pattern)

Goal: Use deconstruction to evaluate a Point or Vector object.

Task: Suppose you have a record TemperatureReading(double Temp, bool IsFahrenheit). Write a switch expression to:

- Convert the temp to Celsius if IsFahrenheit is true (using the formula $C = (F - 32) \times \frac{5}{9}$).
- Return the temp as-is if IsFahrenheit is false.
- **Bonus:** Add a "Guard" (when) to return a warning string if the resulting Celsius temperature is above 100.

5. The File Processor (List Patterns)

Goal: Match the structure of an array or list.

Task: Write a method string AnalyzeData(int[] numbers).

- Match an empty array []: "No data".
- Match an array with exactly one element [var single]: "Single value: {single}".
- Match an array starting with 1, 2 followed by any number of elements: "Sequence starts with 1 and 2".
- Match any array that ends with 0: "Sequence ends in zero".