Exercise 1: The Shape Inspector (is keyword)

Goal: Use the is keyword to identify specific types within a collection.

- **Scenario:** Create a base class Shape and two subclasses: Circle (with a Radius property) and Square (with a Side property).
- **Task:** Create a List<Shape> containing both types. Iterate through the list and print "This is a circle with radius X" or "This is a square with side Y" by checking the type at runtime.

Exercise 2: Safe Downcasting (as keyword)

Goal: Use the as keyword to perform a safe conversion that avoids throwing exceptions.

- **Scenario:** You have an object variable that might contain a string.
- **Task:** Use the as keyword to try and cast the object to a string.
    - If the result is not null, print the string in uppercase.
    - If the result is null, print "Conversion failed: Object is not a string."

Exercise 3: The Data Processor (Pattern Matching)

Goal: Use the modern C# pattern matching syntax (combining is with variable declaration).

- **Scenario:** Write a method ProcessData(object data).
- **Task:** * If data is an int, print its square.
    - If data is a string, print its length.
    - If it’s neither, print "Unknown data type."
    - **Constraint:** Do this using the if (data is int number) syntax.

Exercise 4: Numerical Precision (Explicit Casting)

Goal: Practice manual type conversion and understand data loss.

- **Scenario:** You are building a financial app that calculates averages.
- **Task:** * Create a double variable representing a high-precision price (e.g., 99.99).
    - Create an int variable.
    - Explicitly cast the double to an int and store it.
    - Print both values and observe what happens to the decimal points.

Exercise 5: The UI Component Handler (Mixed Techniques)

Goal: Combine is and as in a simulated real-world interface scenario.

- **Scenario:** You have an array of object[] containing a Button class, a Textbox class, and a SecretKey (a simple int).
- **Task:** Loop through the array:
    1. Use is to check if the item is a SecretKey. If so, print "Key found!"
    2. Use as to try and treat the item as a Button. If successful, call a Click() method on it.
    3. If it is none of the above, print "Unknown component."