Exercise 1: The "Shouting" String

Goal: Create an extension method for the string type called ToShout().

- **Task:** The method should convert the string to uppercase and add "!!!" to the end.
- **Example Use:** "hello".ToShout() should return "HELLO!!!".

Exercise 2: Currency Formatter

Goal: Create an extension method for the double type called ToPrice().

- **Task:** The method should take the number and return a string formatted with a dollar sign and two decimal places.
- **Example Use:** 19.5.ToPrice() should return "$19.50".

Exercise 3: Integer Boundaries

Goal: Create an extension method for int called IsBetween().

- **Task:** This method needs **two parameters** (min and max) in addition to the this parameter. It should return a boolean if the number is within that range (inclusive).
- **Example Use:** 5.IsBetween(1, 10) should return true.

Exercise 4: List Wrapper

Goal: Create an extension method for List<string> called PrintAll().

- **Task:** The method should loop through every string in the list and use Console.WriteLine() to print them to the console.
- **Example Use:** myList.PrintAll();

Exercise 5: The "Safe" Division

Goal: Create an extension method for int called DivideBy().

- **Task:** The method should take another int as a divisor. If the divisor is 0, return 0 (to avoid a crash). Otherwise, return the result of the division.
- **Example Use:** 10.DivideBy(2) returns 5; 10.DivideBy(0) returns 0.

Exercise 6: The User Name Formatter

Goal: Transform it into "[@GEMINI_USER]" using a single chain of extensions.

Hint: You can mix in C#'s native .ToUpper() method anywhere in that chain!

Task: Create **three separate extension methods** (you can put them in one or multiple static classes) and chain them together in Main.

Step 1: Create the Extensions

1. **ToClean()** (Extends string):
    - Removes any leading or trailing whitespace.
    - Example: " admin ".ToClean() →"admin"
2. **ToUserTag()** (Extends string):
    - Adds an "@" symbol to the front.
    - Example: "admin".ToUserTag() → "@admin"
3. **WithAlert()** (Extends string):
    - Wraps the string in brackets to make it look like a notification.
    - Example: "@admin".WithAlert() → "[@admin]"

Step 2: The Chain

In your Main method, start with a messy string:

string rawInput = " gemini_user ";