

# String Calculator Kata

Source: <https://github.com/ardalis/kata-catalog>

## Instructions

This kata is designed to help you learn test-first coding and refactoring. To that end, try not to read ahead or guess what the next requirements might be. Work incrementally, and complete as many steps as you can in a 30 minute period. Continue trying the kata from scratch, until you can complete it entirely within 30 minutes.

## Steps

1. Create a `StringCalculator` with a method `Add(string numbers)` that returns an integer.
  1. Start with the simplest test case of an empty string, then 1 number, then 2.
  2. Solve things as simply as possible!
  3. An empty string should return a sum of 0.
  4. *numbers* can include 0, 1, or 2 integers (e.g. "", "1", "1,2").
  5. `Add` returns the sum of the integers provided in the string *numbers*.
  6. Remember to refactor after each test.
2. Allow the `Add` method to handle an unknown number of numbers (in the string).
3. Allow the `Add` method to handle new lines between numbers (instead of commas):
  1. Example: "1\n2,3" returns 6.
  2. Example: "1,\n" is invalid, but no need to test for it. For this kata we are only concerned with testing correct inputs.
4. Allow the `Add` method to handle a different delimiter:
  1. To change the delimiter, the beginning of the string should be a separate line formatted like this: "[delimiter]\n[numbers]"
  2. Example: ";;\n1;2" returns 3 (the delimiter is ";").
  3. This first line is optional; all existing scenarios (using "," or "\n") should work as before.
5. Calling `Add` with a negative number will throw an exception "Negatives not allowed: " and then listing all negative numbers that were in the list of numbers.
  1. Example: "-1,2" throws "Negatives not allowed: -1".
  2. Example: "2,-4,3,-5" throws "Negatives not allowed: -4,-5".
6. Numbers greater than 1000 should be ignored.
  1. Example: "1001,2" returns 2.
7. Delimiters can be any length, using this syntax: "[| | |]1 | |2 | |3" returns 6.
8. Allow multiple delimiters, using this syntax: "[| |][%]\n1 |2%3" returns 6.
9. Handle multiple delimiters of any length.

## Resources

- [Original Version by Roy Osherove](#)