**Задание: Reactive Calculator (deadline:** implement in one week after seminar**)**

**Part 1. Implement Reactive Sum** (it can be implemented as special case of Part 2)

The requirements the program must fulfill:

1. It will be an application that runs in the terminal (console).
2. Once started, it will run until the user enters exit.
3. If the user enters a:<number>, the *a* collector will be updated to the *<number>*.
4. If the user enters b:<number>, the *b* collector will be updated to the *<number>*.
5. If the user enters anything else, it will be skipped.
6. When both the *a* and *b* collectors have initial values, their sum will automatically be computed and printed on the standard output in the format *a + b = <sum>*.

On every change in *a* or *b*, the sum will be updated and printed.

**Part 2. Implement Reactive Calculator**

Requirements 5 and 6 above should be replaced with the following:

1. If user enters ‘+’ and both *a* and *b* collectors have initial values, then the result of the expression *a + b* will be printed as “a + b = <result>”, and any subsequent change in *a* and *b* will force the expression result be updated and printed.

If user enters ‘-’ and both *a* and *b* collectors have initial values, then the result of the expression *a - b* will be printed as “a - b = <result>”, and any subsequent change in *a* and *b* will force the expression result be updated and printed.

If user enters ‘\*’ and both *a* and *b* collectors have initial values, then the result of the expression *a \* b* will be printed as “a \* b = <result>”, and any subsequent change in *a* or *b* will force the expression result be updated and printed.

If user enters ‘:’ and both *a* and *b* collectors have initial values, then the result of the expression *a : b* will be printed as “a : b = <result>”, and any subsequent change in *a* and *b* will force the expression result be updated and printed.

1. If the user enters anything else, it will be skipped.

**Additional notes for assessments:**

* Using Java Enumerations advanced abilities is a plus;
* Providing Javadoc-comments is a plus;
* Supplying JUnit-tests is a plus.