

Programowanie funkcyjne, interfejsy funkcyjne - zadania

1. Napisz implementację interfejsu funkcyjnego `java.util.function.BiFunction`, który przyjmie `Integer` oraz `Double`, a zwróci `String`. Implementacja w trakcie wywołania wydrukuję na ekranie 2 przekazane argumenty, a na koniec zwróci dodane do siebie `Integer` oraz `Double` jako `String` stosując konkatenację. Pisząc implementację tego interfejsu użyj mechanizmu `method reference`.
2. Napisz interfejs funkcyjny z metodą, która przyjmie od Ciebie `String`, `Integer` oraz Dowolny obiekt. Typem zwracany z takiej metody będzie `Integer`. Napisz 2 implementacje funkcyjne takiej metody:
 - Pierwsza ma zwracać sumę długości przekazanego `Stringa`, przekazanego `Integera` oraz długości `Stringa` zwróconego z wywołanej metody `toString()` na przekazanym ostatnim obiekcie do metody.
 - Druga ma dodać do siebie liczbę wystąpień litery 'a' w pierwszym `Stringu`, wartość przekazaną jako `Integer` oraz ilość wystąpień przecinków zsumowaną z ilością wystąpień '=' w metodzie `toString()` wywołanej na przekazanym ostatnim obiekcie.
3. Napisz implementację interfejsu `BinaryOperator`, która z 2 przekazanych do niej obiektów zwróci większy. Jeżeli przekazane 2 obiekty są sobie równe, zwróć `null`, a potem obsłuż go wykorzystując `Optional`, który wyrzuci wyjątek, w przypadku gdy obiekty są sobie równe.