

# WPROWADZENIE DO J. JAVA –

## ZADANIE DOMOWE 1

### ZADANIE 0 - EASY

Jeżeli nie udało Ci się ukończyć wszystkich zadań na zajęciach – ukończ je (zadania dot. Testów, typów, autoboxingu).

### ZADANIE 1 - MEDIUM

Jeżeli na zajęciach nie udało Ci się uzupełnić klasy Calculator to zrób to tak, aby można było wykonać na niej następujące operacje:

- Dodawanie,
- Odejmowanie,
- Dzielenie,
- Mnożenie,
- Podnoszenie do kwadratu,

Pamiętaj, aby wszystkie metody klasy Calculator miały swoje metody testowe w klasie CalculatorTest.

### ZADANIE 2 - MEDIUM

Utwórz klasę CalculatorFloatingPoint. Utwórz klasę testową CalculatorFloatingPointTest (pamiętaj o konwencji lokalizacji klas źródłowych i testowych).

Dodaj test sprawdzający metodę `float add(float a, float b)` klasy CalculatorFloatingPoint – test się nie skompiluje, bo klasa nie ma jeszcze takiej metody!

Dodaj metodę `float add(float a, float b)` zwracającą 0 (zero) – test powinien się skompilować, ale nie zakończy się powodzeniem, bo metoda nie jest jeszcze poprawnie zaimplementowana (zawsze zwraca zero).

Uzupełnij ciało metody `add` tak, aby zwracała poprawny wynik. Uruchom test – powinien zakończyć się powodzeniem.

Postępuj podobnie dla pozostałych operacji (odejmowanie, dzielenie, mnożenie, dzielenie, podnoszenie do kwadratu).

### ZADANIE 3 - EASY

Utwórz nowy projekt (maven). Dodaj pakiet `pl.sda.home`. W pakiecie dodaj klasy: `Home`, `Room`, `Door`. W klasie `Door` dodaj:

- pole `boolean isOpen`
- metodę `boolean isOpen()` (zwracającą wartość pola `isOpen`)

- metody: `void open()` oraz `void close()`, które przypisują odpowiednie wartości pola `isOpen`.

W klasach `Room` oraz `Home` dodaj:

- pole `Door door`; (w domu są drzwi wejściowe oraz te do pokoju).

W klasie `Home` dodaj:

- pole `Room room`; (w pokoju jest jeden pokój),
- metodę `boolean isAllOpen()` sprawdzającą czy wszystkie drzwi są otwarte (tj. drzwi w domu `this.door.isOpen()` oraz drzwi w pokoju `this.room.door.isOpen()`)
- analogicznie metodę `boolean isAllClosed()` sprawdzającą czy wszystkie drzwi są zamknięte

#### ZADANIE 4 – MEDIUM

W projekcie z zadania 3 do klasy `Home` dodaj:

- metodę sprawdzającą czy którekolwiek drzwi są zamknięte,
- metodę sprawdzającą czy którekolwiek drzwi są otwarte.

Napisz klasy testowe `DoorTest` oraz `HomeTest` sprawdzające zaimplementowane metody.

#### CO KONIECZNIE MUSISZ WIEDZIEĆ/UMIEĆ?

1. Potrafisz utworzyć klasę.
2. Potrafisz utworzyć metodę `main` rozpoczynającą program.
3. Potrafisz wypisać tekst na konsolę.
4. Potrafisz zadeklarować zmienne oraz stałe różnych typów.
5. Potrafisz zadeklarować pola klasy.
6. Potrafisz napisać metody w klasie.
7. Potrafisz napisać konstruktor bezargumentowy i konstruktory argumentowe.
8. Potrafisz ograniczyć widoczność metody/pola.
9. Potrafisz napisać klasę dziedziczącą po innej klasie.
10. Potrafisz dodać dependencje (zależności) w projektach mavenowych.
11. Potrafisz napisać test.