

Ze środowiskiem Visual Studio Code jestem zaznajomiona z poprzednich laboratoriów z programowania (języki C i Python) jednak z powodu nieznanego mi problemu VS Code po komplikacji wypluwa mi na terminal wynik programu razy trzy i więcej. W związku z tym zdecydowałam się na inne środowisko - IntelliJ IDEA z polecenia znajomego.

### Zadanie 1. Pierwszy program w języku Java – komplikacja i uruchomienie

Po skompilowaniu w katalogu pojawiają się pliki odpowiednie do plików z rozszerzeniem .java

W moim przypadku Main.class

Poniżej zrzut ekranu z po komplikacji i wywołaniu programu członków: Kot, Pies, Owca

```
C:\Users\eviko\.jdks\openjdk-19.0.2\bin\java.exe "-javaagent:C:\Program Files\.
Kot
Pies
Owca

Process finished with exit code 0
```

Następnie sprawdzaliśmy poprawność hermetyzacji w naszym kodzie. Po jego komplikacji otrzymujemy komunikat:

```
C:\Users\eviko\Desktop\java\lab1\Ewa_Wojciechowska_lab1\src>Main.java
java: name has private access in Team
```

Przyczyną niepowodzenia w komplikacji kodu jest źle dobrany poziom hermetyzacji – pole name jest prywatne a próba jego wyświetlenia kończy się wyświetlaniem błędu.

Poniżej widać już wynik komplikacji programu z zaimplementowanymi metodami dostępowymi get.Name i set.Name, które są publiczne a zatem pozwalają na wywołanie nazwy name i jej zmianę.

```
C:\Users\eviko\.jdks\openjdk-19.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains
Zwierzeta
Ssaki
Kot
Pies
Owca

Process finished with exit code 0
```

**Zadanie 2. Zabezpieczanie pól obiektów przed modyfikacją**

```
Exception in thread "main" java.lang.Error: Unresolved compilation
problems:
    members cannot be resolved to a type
    Class<members> cannot be resolved to a type

    at Main.main(lab1.java:19)
```

Wydaje mi się, że nie o ten komunikat chodziło, zwłaszcza, że VS Code nie bardzo chciał współpracować a w IDEA informował tylko o braku przypisania .class, ale nie otrzymałem innego. W przypadku przypisania String tablica[] = zespol.members; na wyjściu pojawiała się pierwotna tablica.

Po zmianie przypisania nastąpiła również zmiana na wyjściu:

```
C:\Users\eviko\.jdks\openjdk-19.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ
Zwierzeta
Ssaki
Żaba
Pies
Owca

Process finished with exit code 0
```

Za drugim razem edycja była możliwa, ponieważ nadpisaliśmy naszą wartość, a nie próbowaliśmy ją zmienić.

**Zadanie 3. Visual Studio Code – edycja, komplikacja, uruchamianie i debugowanie kodu języka Java.**

Na potrzeby zadania wrzuciłem kod do VS Code i wykonałem polecenie, zmienne naszego programu wyglądają następująco:



Po wejściu do wnętrza metody otrzymujemy jeszcze ->getName()

VARIABLES

Local

> →getName(): "Zwierzeta"  
args: String[0]@7

zespol: Team@8

members: String[3]@12

0: "Zwierzeta"  
1: "Owca"  
2: "Pies"

name: "Zwierzeta"

Wydaje mi się, że ten element został wykonany przez program poprawnie mimo wcześniejszych problemów.