

Laboratorium 2 – 10.10.2018

Refleksja

1. Utworzyć / otworzyć projekt. Można wykorzystać projekt z poprzednich zajęć.
2. W dowolnej metodzie, która wykona się po starcie programu napisać kod, który:
 - 2.1. Dla bieżącej klasy wypisze wszystkie jej metody oraz typy ich argumentów, pola oraz konstruktory.
 - 2.2. Zmienić kod tak, aby uwzględnił również metody prywatne.
 - 2.3. Stworzyć instancję klasy.
3. Stworzyć instancję klasy, której nazwa podana jest jako argument do metody.
4. Wywołać prywatny konstruktor klasy.
5. Wywołać prywatną metodę klasy.
6. Zmienić wartość w prywatnym polu instancji klasy.

Serializacja

7. Stworzyć klasę implementującą Serializable.
8. Zapisać obiekt tej klasy do pliku.
9. Odczytać obiekt z pliku.
10. Czy odczytany obiekt jest identyczny z oryginalnym?
11. Dodać w klasie pole `private static final long serialVersionUID` i nadać mu wartość.
12. Odczytać poprzednio zapisany obiekt. Co się stało?
13. Dodać w klasie pole typu `java.util.function.Supplier`. Zapisać obiekt do pliku. Co się stało?
14. Odczytać obiekt z pliku `lab02/object.dat` w repozytorium.
 - 14.1. (Opcjonalnie) Przygotować plik z podobnymi danymi.

Singletony (opcjonalnie)

15. Utworzyć implementację wzorca Singleton.
16. Wykonać atak refleksją na singleton.
17. Wykonać atak serializacji na singleton.
18. Wykonać atak wielowątkowy na singleton. (Podpowiedź: wykonanie kodu w konstruktorze musi trwać „trochę dłużej”, albo wyniki będą trudno powtarzalne).
19. Czy można zabezpieczyć Singleton przed tymi atakami?