



Zadanie domowe 2 – implementacja klasy z użyciem zasad hermetyzacji, implementacja metod z użyciem konwersji, test jednostkowy testujący klasę

dr inż. Łukasz Sosnowski
WIT Wyższa Szkoła Informatyki Stosowanej i Zarządzania
pod auspicjami Polskiej Akademii Nauk

1 Kod początkowy klasy

Z materiałów umieszczonych w systemie UBI ściągnij plik Presentation.java
Wgraj go do plików projektu „lab1-ex1” utworzonym w ramach laboratorium 1.
Wgraj go do ścieżki „\src\main\java\pl\wit\lab2\”, następnie odśwież projekt w ECLIPSE poprzez przycisk F5.
Po pojawieniu się klasy „Presentation.java” w pakiecie „pl.wit.lab2” otwórz ją w głównej części okna ECLIPSE.

2 Implementacja

1. Dodaj komentarz do klasy oraz zmiennych składowych klasy. W komentarzu klasy zawrzyj skrótowo opis tworzonej klasy. Nad każdą zmienną składową klasy dodaj 1 liniowy komentarz opisujący daną zmienną. Komentarze tworzymy w języku polskim.
2. Zaimplementuj konstruktor parametryczny 2-argumentowy, w taki sposób aby wartości przekazywanych parametrów odpowiednio inicjalizowały zmienne składowe.
3. Zaimplementuj drugi konstruktor parametryczny (5-argumentowy) stosując jawną konwersję typów
4. Zaimplementuj metodę `getFullName` w taki sposób aby zwracała łańcuch znaków w postaci „imię nazwisko”
5. Zaimplementuj metodę `getBirthdateAsString` zwracającą ręcznie zbudowany łańcuch znaków w postaci daty urodzin w formacie dd.MM.yyyy z uwzględnieniem zer wiodących w numerze dnia i numerze miesiąca.

6. Zaimplementuj metodę `getAge` z użyciem podanego w komentarzu sposobu pobrania bieżącej daty i czasu. Metoda ma zwrócić aktualny wiek względem daty urodzin i chwili obecnej.
7. Zaimplementuj niezbędne gettery i settery dla zmiennych składowych
8. Zadeklaruj i zaimplementuj metodę dodającą do obecnego stanu obiektu kolejny łańcuch znaków przedstawiający odwiedzone miejsca.
9. Zaimplementuj metodę `getPresentationStory` budującą łańcuch znaków na bazie informacji dostępnych wewnątrz klasy. Wynikiem ma być obiekt łańcucha znaków przedstawiający słowny opis prezentacji osoby. Do implementacji użyj instrukcji sterujących *switch*, *if* oraz pętli *for*. Metoda powinna wyświetlać opis bezpośrednio związany z przechowywanymi danymi, lecz również informacje wnioskowane z tych danych, np. liczba znajomych na FB: **5000** = Posiadam **dużą** liczbę znajomych na FB.

3 Test jednostkowy

Utwórz klasę testu jednostkowego `PresentationTest.java`

Przygotuj testy jednostkowe dla każdej metody publicznej (poza setterami i getterami) oraz obu konstruktorów.

Ostatni test wykonaj dla metody `getPresentationStory` który po uruchomieniu powinien wygenerować na konsoli pełny tekst prezentacji a dodatkowo posprawdzać czy łańcuch zawiera wymagane elementy.

Łączna liczba testów nie powinna być mniejsza niż 7 szt.