

Zaawansowane Programowanie Obiektowe i Funkcyjne

Refleksje i adnotacje

Zadanie oceniane nr 5

25-11-2018

Zadania znajdują się w repozytorium (każdy ma swoje konto w podkatalogu Lab8-zadanie5). Należy zsynchronizować się za pomocą polecenia git pull. Po zakończeniu pracy konieczne jest wgranie zmian do repozytorium (commit + push). Proszę pamiętać o poleceniu add.

"Winda"



Całkiem niedawno w pewnym budynku nastąpiła awaria windy. Kabina pełna pasażerów, zatrzymała się i uwięziła kilka osób. Pojawiły się głosy, iż sytuacja do której doszło jest efektem błędu popełnionego przez programistów implementujących sterownik. Ktoś również wspominał o niewystarczającym pokryciu kodu testami jednostkowymi. Na szczęście pasażerami windy byli zdolni ludzie z branży IT, którzy sami postanowili uzyskać dostęp do komponentów windy i manipulować nimi tak aby zmusić windę do ich uwolnienia. Aby być w stanie rozwiązać problem, trzeba najpierw zebrać pewne informacje o tym dźwigu osobowym. Tłenu w kabinie starczy tylko na 90 minut, więc dobrze by było oswobodzić się przed upływem tego czasu...

Poniższe punkty zadania wykonać **za pomocą mechanizmu** refleksji dostarczonego przez standardową bibliotekę Java. **Nie wolno** używać operatora "**new**", "**instanceof**" ani **niczego zmieniać** w pakiecie: `pl.edu.pw.mini.zpoif.fifth.elevator` (chyba że podpunkt zadania będzie tego wymagał wprost). Należy zabezpieczyć kod na okoliczność wystąpienia wyjątku. Można stosować casting typów.

Kod rozwiązania i jego demonstracji należy umieścić w pakiecie:

`pl.edu.pw.mini.zpoif.gradedtask.fifth.solution`. Działanie każdego punktu należy zaprezentować w klasie `Demonstrator`, wraz z odpowiednimi informacjami wypisywanymi na konsolę. Każdy z punktów powinien mieć swoją metodę, która będzie wywołana.

Prace do wykonania:

1. Utwórz obiekt klasy "pl.edu.pw.mini.zpoif.fifth.elevator.MiNIWinda" używając jej nazwy tekstowej.
2. Ustal za pomocą refleksji liczbę konstruktorów klasy pl.edu.pw.mini.zpoif.fifth.elevator.MiNIWinda. Znajdź konstruktor o największej liczbie parametrów oraz wypisz ich typy i nazwy.
3. Wypisz wartość pola PRODUCENT.
4. Wypisz wartość pola "kolor".
5. Wypisz listę wszystkich metod, które są dziedziczone przez klasę MiNIWinda oraz pełną nazwę nadklasy.
6. Wypisz nazwę klasy (bez pakietu), której instancja jest przypisana do pola sterownikWindy, które to znajduje się wewnątrz instancji klasy MiNIWinda, utworzonej za pomocą konstruktora bezparametrowego.
7. Ustal i wypisz nazwę pakietu nadklasy klasy MiNIWinda.
8. Przygotuj kod wypisujący nazwy wszystkich klas (bez pakietu) wraz z ilością publicznych metod, które są nadklasami danej klasy.
9. Podmień wartość pola sterownikWindy za pomocą instancji klasy: UniwersalnySterownikWindy.
10. Ustaw pole sygnalPowitalny instancji klasy PanelSterowniczy będącej polem klasy: Kabina wartością powitalną: "Guten morgen!". Uruchom metodę: getSygnalPowitalny();
11. Utwórz następujące adnotacje:
 - WaznyElement, która:
 - ✓ zawiera parametr o nazwie stopienWaznosci, któremu można przyporządkować tylko trzy wartości: "Wazny", "Arcywazny", "Kluczowy"
 - ✓ przeznaczona jest dla pól i metod
 - ✓ widoczna dla mech. refleksji tylko w czasie działania aplikacji
 - DomyslneWartosciBezpiecznika, która:
 - ✓ zawiera parametry:
 - ◆ amperaz integer (domyślnie 30)
 - ◆ stan pole przyjmujące dwie wartości: "Włączony", "Wyłączony" (domyślnie "Włączony")
 - ✓ przeznaczona jest dla konstruktorów
 - ✓ widoczna dla mech. refleksji we wszystkich okolicznościach
 - MaksymalnaPredkosc
 - ✓ zawiera parametr o nazwie maksymalnaPredkosc, bez wartości domyślnej
 - ✓ przeznaczona jest dla parametrów
 - ✓ widoczna dla mech. refleksji tylko w czasie działania aplikacji

12. Należy wykonać następujące czynności:

Oznacz adnotacją `DomyslneWartosciBezpiecznika`:

- publiczny konstruktor klasy `BezpiecznikZasilania`, bez podawania wartości jej parametrów

Oznacz adnotacją `WaznyElement`:

- pole `kolor` klasy `MiniWinda` z wartością "Ważny"
- pole `dopuszczalnaIloscOsob` klasy `MiniWinda` z wartością "Kluczowy".
- metodę `jedzDoGory` klasy `MiniWinda` z wartością "Arcywazny".
- metodę `jedzDoDolu` klasy `MiniWinda` z wartością "Arcywazny".

Oznacz adnotacją `MaksymalnaPredkosc` parametry metod:

- `jedzDoGory` (wartość 5)
- `jedzNaDol` (wartość 6)

* Jeśli kompilator nie pozwoli zrobić tak, żeby pozwolił

13. Należy wykonać następujące czynności:

- Wypisz nazwy elementów adnotowanych z klasy : `MiniWinda` wraz z nazwami i parametrami adnotacji.
- Uruchom adnotowane adnotacją `WaznyElement` metody, losując wartości ich parametrów tak, aby wartości te nie przekraczały wartości maksymalnej prędkości podanej w adnotacji.

14. Podmień pole `bezpiecznikZasilania` klasy `Naped` której obiekt przypisany jest do pola `naped` w klasie `MiniWinda`, na bezpiecznik stworzony w oparciu o wartości z adnotowanego konstruktora, za pomocą adnotacji: `DomyslneWartosciBezpiecznika`.

15. Ustaw wszystkie pola adnotowane adnotacją `WaznyElement` (wartość: "KLUCZOWY"), uznana przez siebie wartością stworzoną oczywiście za pomocą mechanizmu refleksji.