

Zaawansowane Programowanie Obiektowe i Funkcyjne

Strumienie i wyrażenia lambda

Zadanie oceniane nr 4

21-11-2019

Kod wstępny znajduje się w podkatalogu Lab7-zadanie4. Po zakończeniu pracy konieczne jest wgranie zmian do repozytorium.

"Statki powietrzne"



W dniu dzisiejszym należy poddać analizie i przetwarzaniu kolekcję statków powietrznych (samoloty, śmigłowce, szybowce i spadochrony) ze względu na ich własności. Należy dostarczyć klasę implementującą interfejs `Strumienie`, która przyjmuje jako argument konstruktora dwie kolekcje generowanego przez klasę `GeneratorStatkowPowietrznych`. W zależności od wymogów zadania, raz pracuje na jednej, a raz na drugiej kolekcji. Należy również dostarczyć klasę demonstracyjną, której metoda prezentuje na konsoli efekt działania implementacji poszczególnych metod z w/w interfejsu.

Używając mechanizmów opartych na strumieniach i wyrażeniach lambda (jeśli jest to wystarczające) należy dostarczyć implementację następujących metod:

1. Metoda `StatekPowietrzny` `getNajlzejszyStatekPowietrzny()`;
Zwraca najlżejszy statek powietrzny z kolekcji wszystkich statków powietrznych.
2. Metoda `StatekPowietrzny` `getNajciezszySmiglowiec()`;
Zwraca najcięższy śmigłowiec począwszy od 3-go helikoptera w kolekcji wszystkich statków powietrznych.
3. Metoda `Smiglowiec` `getSmiglowiecONajmniejszymWirniku()`;
Zwraca śmigłowiec o najmniejszym wirniku, z typem zaczynającym się na "Mi" (wielkość liter jest nieistotna) począwszy od 2-go helikoptera w kolekcji wszystkich statków powietrznych.
4. Metoda `Set<StatekPowietrzny>` `getSamolotyLatajaceWyzejNiz19000M()`;
Zwraca zbiór samolotów liczący co najwyżej 7, które latają na pułapie wyższym niż 19000 metrów, o powierzchni nośnej nie mniejszej niż 25, począwszy od 5-go samolotu w kolekcji wszystkich statków powietrznych

5. Metoda **public** `List<NapędzanyStatekPowietrzny>`

`getPosortowaneNapędzaneStatkiPowietrznePosortowanePoZasięgu()`

Zwraca co najwyżej 3 z posortowanych malejąco po zasięgu statków powietrznych począwszy od 5-tego, o prędkości wznoszenie nie mniejszej niż 100, pochodzących z kolekcji napędzanych statków powietrznych.

6. Metoda `Szybowiec` `getSzybowiecONajwiększejDokonalosciINajniższejMasie();`

Spośród trzech szybowców o największej doskonałości, zwraca ten o najmniejszej masie.

7. Metoda `Map<String, StatekPowietrzny>` `getSamolotyOPowierzchniNośnejWiększejNiz20();`

Zwraca mapę samolotów <typ, samolot> o powierzchni nośnej większej niż 20, pochodzących z kolekcji napędzanych statków powietrznych. W przypadku wystąpienia duplikatu informujemy o tym stosownym wpisem na konsolę i wybieramy ten o niższej masie.

8. Metoda **int** `getSumeDokonalosciSzybowcowOMasieWiększejNiz350PoczOdDrugiego();`

Zwraca sumę doskonałości szybowców o masie nie mniejszej niż 350, poczynając od drugiego na liście wszystkich statków powietrznych.

9. Metoda `List<StatekPowietrzny>`

`getPierwszePiecSamolotowLubSpadochronowPosortowanychMalejacoPoMasiePoczOd2();`

Zwraca pierwsze pięć samolotów lub spadochronów posortowanych malejąco po masie począwszy od 2-giego na liście wszystkich statków powietrznych.

10. Metoda `Map<Integer, StatekPowietrzny>`

`getMapeSpadochronowODopMasieSkoczkaWiększejNiz85IgnorujDuplikaty();`

Zwraca nam mapę spadochronów o dopuszczalnej masie skoczka większej niż 85. Ew. duplikaty są zawężane.

11. Metoda **void** `zmienNazwyTypowWZaleznosciOdPierwszychElementow();`

Zmienia nam nazwy typów wszystkich statków powietrznych w kolekcji na małe litery, jeśli pierwszy element w kolekcji jest spadochronem, a drugi także oraz plus dopuszczalna masa skoczka (w drugim) jest większa niż 100.