

Laboratorium #10

Interfejsy, interfejs Comparable

Organizacja kodu: jeden projekt, każde zadanie oddzielne - oddzielny pakiet; każda klasa/interfejs w oddzielnym pliku.

Informacje wstępne:

- tam gdzie nie wskazano, to implementacja metod można być dowolna (niepusta) o ile kod się kompiluje i wykonuje bez błędów
- zwróć uwagę na rozróżnienie pojęć tablica a lista tablicowa (lista) - te drugie to `ArrayList`
- sortowanie tablic realizuj za pomocą `Arrays.sort(variable)`
- sortowanie list tablicowych możesz zrobić na jeden ze sposobów:

```
Collections.sort(list);  
list.sort(Comparator.naturalOrder());  
list.sort(null)
```

- w metodzie `compareTo` zgodnie “ze zwyczajami” nie dodawaj zabezpieczenia przed `null`. W innych miejscach - opcjonalnie.
- jeśli to wygodne, w klasie możesz dodać konstruktor parametryczny czy nadpisanie metody `toString`.
- w wielu zadaniach na “ćwiczenia” przy implementacji `Comparable` zaburzamy “porządek naturalny” w celach edukacyjnych - należy wykonać tak jak w poleceniu
- w sytuacji gdy chcemy porównywać obiekt w typie klasy bazowej i z obiektem typu klasy pochodnej o tych samych wspólnych polach - przyjmij, że obiekt w typie klasy pochodnej jest “wyższy” w porządku

1. Zadania ze zbioru do wykonania w trakcie bieżących ćwiczeń:

| Ostatnia cyfra numeru albumu: nieparzysta | |
|--|----------------------|
| Nazwa działu | Numer zadania |
| Interfejsy | 5, 10, 11, 15, 17 |
| Interfejs ‘Comparable’ | 1, 9, 11, 12, 18, 21 |

| Ostatnia cyfra numeru albumu: parzysta | |
|---|----------------------|
| Nazwa działu | Numer zadania |
| Interfejsy | 4, 10, 11, 14, 17 |
| Interfejs ‘Comparable’ | 1, 7, 11, 12, 17, 22 |