

Przygotowujemy aplikacje do zarządzania załadunkiem kontenerów.

Kontenery mogą być później transportowane za pomocą różnego rodzaju pojazdów – statków, pociągów, ciężarówek itp.

Projektowany przez nas system będzie się zajmować załadunkiem kontenerów na kontenerowiec – statek wyposażony w specjalne prowadnice pozwalające na przewóz kontenerów.

Kontenery mogą być różnych typów w zależności od ładunku. Banany powinny być transportowane w kontenerach chłodniczych; mleko powinno być transportowane w kontenerach na płyny; hel powinien być transportowany w kontenerach na gaz. Wszystkie te kontenery mają pewne cechy wspólne:

#### Wszystkie kontenery mają:

- Masę ładunku (w kilogramach)
- Wysokość (w centymetrach)
- Waga własna (waga samego kontenera, w kilogramach)
- Głębokość (w centymetrach)
- Numer seryjny
  - Format numeru to KON-C-1
  - Pierwszy człon numeru to zawsze "KON"
  - Drugi człon reprezentuje rodzaj kontenera
  - Trzeci człon to liczba. Liczby powinny być unikalne. Nie powinno być możliwości powstania dwóch kontenerów o tym samym numerze. Numery powinny być generowane przez system.
- Maksymalna ładowność danego kontenera w kilogramach

#### Wszystkie kontenery powinny pozwolić na:

- Opróżnienie ładunku
- Załadowanie kontenera daną masą ładunku
  - Jeśli masa ładunku jest większa niż pojemność danego kontenera powinniśmy wyrzucić błąd `OverfillException`

## Kontenery na płyny (L)

Kontenery na płyny pozwalają na przewożenie ładunku niebezpiecznego (np. paliwo) i ładunku zwykłego (np. mleko).

- Kontenery tego typu powinny implementować interfejs `IHazardNotifier`
  - Interfejs ten pozwala na wysłanie notyfikacji tekstowej w trakcie zajścia niebezpiecznej sytuacji wraz z informacją o numerze kontenera.
- W momencie uruchomienia metody ładującej towary do kontenera powinniśmy:
  - Jeśli kontener przechowuje niebezpieczny ładunek – możemy go wypełnić jedynie do 50% pojemności
  - W innym wypadku możemy go wypełnić do 90% jego pojemności
  - Jeśli naruszymy dowolną z opisanych reguł – powinniśmy zgłosić informacje o próbie wykonania niebezpiecznej operacji.

## Kontenery na gaz (G)

Kontenery przechowujące gaz przechowują dodatkową informację na temat ciśnienia (w atmosferach).

- W momencie kiedy opróżniamy kontener na gaz – pozostawiamy 5% jego ładunku wewnątrz kontenera.
- Powinien zaimplementować interfejs `IHazardNotifier`. Metoda powinna pozwolić na informowanie o zajściu niebezpiecznego zdarzenia wraz z numerem seryjnym danego kontenera.
- Jeśli masa ładunku przekroczy dopuszczalną ładowność – chcemy zwrócić błąd.

## Kontener chłodniczy (C)

Kontener chłodniczy zawiera informacje na temat:

- Rodzaj produktu, który może być przechowywany w danym kontenerze.
- Temperatura utrzymywana w kontenerze.
- Kontener może przechowywać wyłącznie produkty tego samego typu.

- Temperatury kontenera nie może być niższa niż temperatura wymagana przez dany rodzaj produktu.

Example of possible products and temperatures.

Product	Temperature
Bananas	13,3
Chocolate	18
Fish	2
Meat	-15
Ice cream	-18
Frozen pizza	-30
Cheese	7,2
Sausages	5
Butter	20,5
Eggs	19

Nasza aplikacja powinna pozwolić na przygotowanie danego kontenerowca do rejsu. O samym kontenerowcu chcielibyśmy pamiętać:

- Wszystkie kontenery jakie dany statek transportuje
- Maksymalna prędkość jaką kontenerowiec może rozwijać (w węzłach)
- Maksymalna liczba kontenerów, które mogą być przewożone
- Maksymalna waga wszystkich kontenerów jakie mogą być transportowane poprzez statek (w tonach)

Chcemy, aby aplikacja wspierała następujące operacje:

- Stworzenie kontenera danego typu
- Załadowanie ładunku do danego kontenera
- Załadowanie kontenera na statek
- Załadowanie listy kontenerów na statek
- Usunięcie kontenera ze statku
- Rozładowanie kontenera
- Zastąpienie kontenera na statku o danym numerze innym kontenerem
- Możliwość przeniesienia kontenera między dwoma statkami
- Wypisanie informacji o danym kontenerze
- Wypisanie informacji o danym statku i jego ładunku

Następnie w metodzie Main spróbuj wykorzystać przygotowane przez siebie klasy i metody. Sprawdź czy jesteś w stanie wykonać wszystkie opisane w tekście akcje.

## Rozszerzenie zadania – symulacja działania aplikacji

Zadanie dla chętnych. Spróbuj przygotować interfejs konsolowy, który pozwoliłby na realizację wszystkich funkcji. Przykład działania interfejsu został pokazany poniżej.

Użytkownik uruchamia aplikację. System wyświetla:

```
Lista kontenerowców:  
Brak  
  
Lista kontenerów:  
Brak  
  
Możliwe akcje:  
1. Dodaj kontenerowiec
```

Użytkownik wybiera 1. W tym momencie system prosi po kolei o podanie wszystkich niezbędnych danych. Po zakończeniu system wyświetla ponownie ekran główny.

```
Lista kontenerowców:  
Statek 1 (speed=10, maxContainerNum=100, maxWeight=40000)  
  
Lista kontenerów:  
Brak  
  
Możliwe akcje:  
1. Dodaj kontenerowiec  
2. Usun kontenerowiec  
3. Dodaj kontener
```

Po dodaniu kontener pojawia się na liście kontenerów. Następnie użytkownik ma możliwość umieszczenie kontener'a na statku, usunięcie danego kontenera....