

- Chciałabym zmienić żeby wszystkie zmienne nie były public, tylko żeby miały też inne specyfikatory dostępu
- W Vehicle zaimplementowałam metodę reduceSpeed a w klasach Car i Plane ustawiłam final int MAX_SPEED z maksymalną prędkością dla każdego z pojazdów. Chciałabym, żeby ta metoda reduce speed wywoływała się sama przy każdym wyświetleniu wszystkich pojazdów, tak, żebym nie musiała 'ręcznie' patrzeć, czy pojazd ma przekroczoną MAX_SPEED czy nie.

```
public void reduceSpeed(){
    if (speed > MAX_SPEED) {
        System.out.println("muszę zmniejszyć prędkość, ponieważ przekracza ona maksymalną prędkość pojazdu");
        setSpeed(getMAX_SPEED());
    }else
        setSpeed(getSpeed());
}
```

to jest metoda abstrakcyjna z klasy Vehicle, ona nie musi być abstrakcyjna. Prócz tej metody jest jeszcze metoda abstrakcyjna SpeedUp. Wystarczy, że jedna z nich będzie abstrakcyjna

- Zgodnie z wytycznymi do projektu musiałam gdzieś wcisnąć tablice, więc wymyśliłam że obiekty pojazdów będą przechowywane w tablicy i wyświetlane za pomocą pętli, jednak nie działa to :P
Wyświetlają mi się tylko auta, samoloty nie. Bardzo proszę o naprawienie tego
- W klasie Car i w klasie Plane są zmienne, których nie ma w klasie Vehicle, jest to doors w aucie i aerofil_length w samolocie. Chciałabym, żeby one były w konstruktorze prócz tych zmiennych dziedziczonych z Vehicle. Żeby samolot miał aerofil_length, a auto doors dodatkowo do tych zmiennych w konstruktorze dziedziczonym po Vehicle

```
public int aerofil_length; //nie wiem jak to wsadzić do konstruktora
final int MAX_SPEED = 1500;

public int getMAX_SPEED() {
    return MAX_SPEED;
}

public Plane(String brand, int year, String color, int wheels, int speed, Engine engine, Size size) {
    super(brand, year, color, wheels, speed, engine, size);
}
```

Wiem, że dużo rzeczy w tym projekcie nie ma sensu tak logicznie, ale musiałam w projekt o prostym pomysle wcisnąć wszystkiego po trochu, dlatego jest w nim wiele rozwiązań których nie ma sensu, no ale to jest na zaliczenie więc musi być trochę bez sensu.

Tu jest lista wytycznych które muszą znajdować się w projekcie:

ST - Programowanie obiektowe - ćwiczenia 7 (zaliczenie ćwiczeń) - 10.06.2021 (czwartek)

Temat 7 - Omówienie samodzielnie opracowanych programów w podejściu obiektowym - zaliczenie ćwiczeń

Student definiuje sobie dowolną treść programu. Treść tego programu powinna być spójna, tzn. dotyczyć jednego obszaru (fragmentu) rzeczywistości, np. sprzedaż, wypoczynek, transport, zdrowie, statystyka, itp.

Program na zaliczenie powinien zawierać następujące konstrukcje z zakresu programowania obiektowego:

Wymogi na ocenę dost:

- klasy,
- obiekty,
- pola dla obiektów,
- pola dla klasy (static),
- konstruktory,
- konstruktory rekursyjne,
- metody bez void,
- metody z void,
- metody: akcesor, mutator,
- dziedziczenie,
- hermetyzacja,
- operator instanceof,
- toString,
- przeciążenia metod,
- referencje,
- typy obiektowe.

Wymogi na ocenę dobry:

- obiekty tablicowe,
- klasy abstrakcyjne,
- metody abstrakcyjne,
- klasy finalne,
- metody finalne,
- interfejsy,
- abstrakcja,
- polimorfizm,
- wyjątki (try-catch),
- typ wyliczeniowy (enum).

Wymogi na ocenę bardzo dobry:

- kopiowanie klas i obiektów,
- klasy anonimowe,
- Singleton,
- kolekcje (w ujęciu obiektowym),
- destruktor,
- tekstowy / graficzny interfejs użytkownika.

Jak zauważysz części rzeczy które są wymagane nie mam w tym projekcie, bo po prostu nie potrafię ich zrobić. Na zajęciach zrobiliśmy w sumie tylko rzeczy z wymagań na 3, resztę mamy sami się domyślić, więc trochę się domyślałam jak zrobić, poszukałam, zaimplementowałam, ale części rzeczy nie umiem i nie rozumiem. Jakby ci się chciało i miałbyś jakiś ekstra czas, żeby marnować na mnie przy tym projekcie, to proszę dodaj mi coś, ale jeśli nie masz ekstra czasu, albo po prostu ci się nie chce, to tego nie rób i tak wystarczająco mi dupę uratowałeś