

#### 4. Домашнее задание

### Задание 1

Ранее вы написали модель приложения для автомобильных гонок.

Теперь добавьте для каждой категории в этой модели перечисления. Для каждого перечисления нужно определить аргументы и метод

`toString` для вывода данных аргументов.

Для класса «Легковые автомобили» добавьте в качестве списка следующие типы кузова:

«Седан»,

«Хетчбэк»,

«Купе»,

«Универсал»,

«Внедорожник»,

«Кроссовер»,

«Пикап»,

«Фургон»,

«Минивэн».

Подсказка

Русское название типа кузова должно быть аргументом для каждой константы перечисления, то есть должно быть передано в параметр конструктора при создании элемента

`enum`. А `toString` должен быть реализован в виде «*Тип кузова*»: «*Название типа кузова на русском языке*».

Для грузовых автомобилей добавьте перечисления по типу грузоподъемности:

N1 (с полной массой до 3,5 тонн);

N2 (с полной массой свыше 3,5 до 12 тонн);

N3 (с полной массой свыше 12 тонн).

#### Подсказки

Верхнюю и нижнюю границы грузоподъемности нужно указать в виде `Float` -значений в `enum`. А `toString` должен выводить значения вида *«Грузоподъемность»: от «нижняя граница» тонн до «верхняя граница» тонн.*

Если одной из границ не существует, ее можно передать как `null` -значение и обработать `null` в `toString`. Если какая-то из границ `null`, то фраза должна изменяться. Например, если нижняя граница `null`, то возвращаемое значение в `toString` будет *«Грузоподъемность»: до «верхняя граница» тонн.*

А для автобусов добавьте классификацию с помощью перечислений по типу вместимости:

особо малая (до 10 мест);

малая (до 25);

средняя (40–50);

большая (60–80);

особо большая (100–120 мест).

## Подсказки

В этом перечислении нижняя и верхняя границы вместимости должны быть реализованы с помощью `Integer`, а `toString` должен выводить значения в виде *«Вместимость»*: *«нижняя граница»* — *«верхняя граница» мест*.

Если одной из границ нет, то реализация `toString` должна работать так же, как в случае с грузоподъемностью.

## Критерии проверки

Реализовано перечисление для класса «Легковые автомобили».

Реализовано перечисление для класса «Грузовые автомобили».

Реализовано перечисление для класса «Автобусы».

## Задание 2

В каждый класс транспортного средства добавьте поле, которое содержит тип транспортного средства, а также геттер и сеттер для этого поля. В разных классах транспортных средств должны быть разные типы.

В абстрактный класс

`Transport` добавьте абстрактный метод `printType`, который выводит в консоль либо значение типа транспортного средства, либо, если тип транспортного средства не указан, строку вида *«Данных по транспортному средству недостаточно»*.

## Критерии проверки

В класс «Легковые автомобили» добавлено поле `enum` .

Созданы геттеры и сеттеры для данного поля.

В класс «Грузовые автомобили» добавлено поле `enum` .

Созданы геттеры и сеттеры для данного поля.

В класс «Автобусы» добавлено поле `enum` .

Созданы геттеры и сеттеры для данного поля.

Создан абстрактный метод `printType` в абстрактном классе `Transport` .

Если тип транспортного средства не указан, то метод `printType` выводит строку вида «*Данных по транспортному средству недостаточно*».