

ДИАГРАММЫ КЛАССОВ

Создайте абстрактный класс `AbstractSpace`, реализующий интерфейс `Space`.

Классы `RentedSpace` и `OwnedSpace` должны наследоваться от класса `AbstractSpace`.

Добавьте классу `AbstractSpace` общедоступную статическую операцию `period`. У операции два аргумента: `from` и `to`. Оба аргумента имеют тип `Date`. Операция возвращает значение типа `Period`. Добавьте классу частные атрибуты `person` типа `Person`, `vehicle` типа `Vehicle`, статический атрибут `empty` типа `LocalDate`, предназначенный только для чтения.

Классы `RentedSpace` и `OwnedSpace` состоят из классов `Person` – владелец ТС (транспортное средство) и `Vehicle` – ТС.

Класс `OwnedSpace` включает в себя класс `Position`, который можно определить как упорядоченное множество точек (класс `Point`), который не может меняться.

Отобразите на диаграмме, что `Vehicle` исполняет `Order` с классом ассоциации `FormEntryOrder`.

Ассоциация между классами `Order` и `OrderLine` использует квалификатор. Квалификатор указывает, что в соответствии с заказом для каждого экземпляра продукта (`Product`) может существовать только одна строка заказа. Полюс со стороны `OrderLine` называется `LineItems`.

Класс `Person` характеризуется частным атрибутом `name` (строка) для хранения 3 составляющих. Имеет производный атрибут `age`. Общедоступная константа класса: `unknow` типа `Person` с начальным значением пустая строка.

Для работы класса `Person` нужен интерфейс `Space`.

Создайте перечисление `VehicleTypes` – типа ТС. Константы перечисления: `NONE` `CAR` `CROSSOVER` `MOTOR_BIKE` `SUV` `TRUCK` `OTHER`