**Финальный проект 2 спринта**

Впереди твой первый проект. Чтобы вспомнить весь материал, загляни в шпаргалки: [первый спринт](https://praktikum.notion.site/Java-dabd350cfc0a449d8d828ebfd42d9b46), [второй спринт](https://praktikum.notion.site/Java-3a6eda76d10d4d7394b64ceab234a6eb).

**Иерархия классов**

Создай иерархию классов с продуктами, которые можно положить в корзину. Создай их в отдельном пакете model.

В пакете должны быть:

1. **Класс Food** — абстрактный класс с полями:
   * amount — количество продукта в килограммах (целое число);
   * price — цена за единицу (вещественное число);
   * isVegetarian — флаг, который показывает, вегетарианский ли продукт.

Поля должны быть доступны только из классов-потомков.

1. **Интерфейс Discountable** c единственным методом getDiscount. Он возвращает размер скидки — вещественное число.

Все классы, реализующие этот интерфейс, возвращают значение 0 в методе getDiscount, если в задании не указано иное.

1. **Наследники класса Food: Meat и Apple**

Класс Meat — для мяса. Это класс с конструктором. Конструктор принимает два входных параметра — количество и цену. И устанавливает для флага isVegetarian правильное значение.

Класс Apple — для яблок. У него есть ещё строковое поле colour — цвет яблок. Конструктор принимает три параметра: количество, цену и цвет яблок — и устанавливает флаг isVegetarian.

Для красных яблок (colour равно "red") скидка должна быть равна 60%: соответствующий метод возвращает значение 60.

**Пакеты и подпакеты**

Создай иерархию пакетов.

**Подпакет constants**. Заведи подпакет constants в пакете model. Там создай два класса:

* Discount для значений скидок,
* Colour для цветов red и green — это цвета яблок. В них все поля должны быть public static final.

**Пакет service.** В нём создай класс для корзины — ShoppingCart. Добавь в него поле — массив элементов из объектов типа Food. Понадобится конструктор с одним параметром — массивом элементов Food. С его помощью ты создашь объект корзины.

В классе ShoppingCart реализуй методы для работы с корзиной:

* получить общую сумму товаров в корзине без скидки,
* получить общую сумму товаров в корзине со скидкой,
* получить общую сумму всех вегетарианских продуктов в корзине без скидки.

**Код для функциональности**

Отдельно от пакетов создай класс Main с методом main.

В методе main реализуй следующую логику:

Инициализируй данные. Для этого создай объекты:

* Мясо в количестве 5 кг по цене 100 рублей за кг;
* Яблоки красные в количестве 10 кг по цене 50 рублей;
* Яблоки зелёные в количестве 8 кг по цене 60 рублей.

Создай массив продуктов из этих элементов и проинициализируй корзину с помощью созданного массива.

Посчитай и выведи на экран значения для созданной корзины:

* общую сумму товаров без скидки,
* общую сумму товаров со скидкой,
* сумму всех вегетарианских продуктов без скидки.

Все переменные класса должны быть недоступны извне, если не указано иное. Доступ к ним реализуй через известные тебе методы: создавай только те, которые действительно нужны для реализации описанной функциональности.

**Как сдать проект**

Создай проект в IDE: поможет [инструкция](https://code.s3.yandex.net/qa-automation-engineer/java/track2/cheatsheets/sprint2/ide_project_sprint_2.pdf) и знания из темы «Локальная среда разработки». Обрати внимание: нужно использовать Java 11.

Сдай в тренажёре архивом — вот [инструкция по сдаче](https://code.s3.yandex.net/qa-automation-engineer/java/track2/cheatsheets/sprint2/upload_project_sprint_2.pdf).

**Как будут оценивать твою работу**

* Есть пакет model с правильно реализованной иерархией классов.
* Есть подпакет constants с константами и пакет service с классом ShoppingCart. В ShoppingCart реализованы методы.
* Все классы и интерфейс реализованы по требованиям в задании.
* В коде нет лишних сущностей: только то, что необходимо для реализации функциональности.
* В проекте используется Java 11.