

## Контрольная работа: Симулятор Экономической Системы (СЭС)

Разработать систему классов для моделирования базового производства и управления ресурсами в виртуальной экономике.

### Требования к реализации

#### Часть 1: Абстракция и наследование (4 балла)

1. **Абстрактные Классы:** Создайте два абстрактных базовых класса, используя модуль abc:
  - **Resource:** Базовый класс для всех ресурсов. Должен иметь поля name (название ресурса) и amount (количество).
  - **Building:** Базовый класс для всех производственных зданий. Должен иметь поле name и словарь storage для хранения ресурсов.
2. **Наследование и Полиморфизм:**
  - Наследуйте от Resource конкретные классы: **Wood** (Древесина) и **Food** (Еда).
  - Наследуйте от Building конкретные классы: **Farm** (Ферма) и **LumberMill** (Лесопилка).
  - В Building определите **абстрактный метод** produce(self), который будет реализован в подклассах. Farm производит Food, а LumberMill производит Wood.

#### Часть 2: Магические методы и перегрузка Операторов (3 балла)

1. **\_\_str\_\_:** В классе **Resource** реализуйте этот метод для удобного строкового представления в формате, например: Resource: Wood, amount: 150
2. **\_\_add\_\_:** В классе **Resource** реализуйте перегрузку оператора сложения (+). Сложение двух экземпляров ресурса должно создавать новый экземпляр того же типа с суммарным количеством. Если типы ресурсов разные (например, Wood + Food), должно быть сгенерировано исключение (TypeError).

### Часть 3: Геттеры, Сеттеры и property (3 балла)

1. **Свойство amount:** В классе **Resource** используйте **@property** для поля **amount**.
2. **Сеттер:** Напишите **сеттер** (**@amount.setter**) для этого свойства, который обеспечивает **безопасность данных**:
  - Значение **amount** должно быть всегда **целым числом**. Присвоение нецелого числа должно быть преобразовано (например, округлено) или отклонено.
  - Запрещается устанавливать **отрицательное** значение. При попытке присвоить отрицательное число должно генерироваться исключение (**ValueError**).

### Часть 4: Декрипторы (4 балла)

1. **Декриптор ResourceLimiter:** Создайте класс **декриптора** для управления максимальной емкостью склада для конкретного ресурса в здании. Декриптор должен принимать при инициализации максимальное значение (**max\_value**).
2. **Применение:** Используйте этот декриптор в подклассах **Building** (например, **max\_wood = ResourceLimiter(100)** в **LumberMill**).
3. **Логика:** Модифицируйте методы **produce()** в зданиях так, чтобы они **не могли добавить** ресурс в хранилище, если его общее количество превысит установленный лимит, управляемый декриптором.

### Часть 5: Статические методы и методы класса (4 балла)

1. **Статический Метод (@staticmethod):** В классе **Building** реализуйте статический метод **calculate\_production\_cost(resource\_type)**. Этот метод должен принимать тип ресурса (например, класс **Wood** или **Food**) и возвращать базовую "стоимость" его производства (например, 10 для еды, 5 для древесины). Метод не должен зависеть ни от экземпляра, ни от класса.

2. **Метод Класса (@classmethod):** В классе **Building** реализуйте метод класса **create\_initial\_setup(name)**. Этот метод должен принимать только имя и возвращать новый экземпляр здания с этим именем и **фиксированным начальным запасом** ресурсов, который задан внутри метода.

## Часть 6: Декораторы (3 балла)

1. **Декоратор log\_production:** Создайте функцию-декоратор, которая принимает метод produce здания.
2. **Логирование:** Декоратор должен **автоматически выводить** в консоль сообщение **до** вызова метода производства и **после** его завершения. Сообщение должно включать имя здания. Пример: "--- НАЧАЛО ПРОИЗВОДСТВА: Farm [Green Acres] ---" и "--- ПРОИЗВОДСТВО ЗАВЕРШЕНО: Farm ---".
3. **Применение:** Примените этот декоратор к методу produce() в классах Farm и LumberMill.

## Тестирование

Напишите блок кода, который демонстрирует работу всех реализованных элементов:

- Создайте экземпляры Farm и LumberMill с помощью метода класса.
- Покажите, как работает сложение ресурсов (+).
- Продемонстрируйте срабатывание ValueError при попытке установить отрицательный amount ресурсу.
- Вызовите produce() для обоих зданий и покажите, что декоратор сработал, а лимиты дескриптора соблюдаются.
- Вызовите статический метод для расчета стоимости.

**Общее количество баллов: 21**