

# Умная система управления финансами

## Функциональные требования

### 1. Базовые дескрипторы

#### 1.1. FinancialDescriptor (Базовый класс)

- **Назначение:** Базовый дескриптор для всех финансовых атрибутов
- **Функциональность:**
  - Валидация минимальных/максимальных значений
  - Ведение истории изменений
  - Автоматическое сохранение временных меток
- **Атрибуты:**
  - name - имя атрибута
  - min\_value, max\_value - ограничения значений
  - История изменений с timestamp

#### 1.2. BalanceDescriptor

- **Наследует:** FinancialDescriptor
- **Особенности:**
  - Запрет отрицательного баланса для дебетовых счетов
  - Разрешение отрицательного баланса для кредитных счетов (в пределах лимита)
  - Автоматическая проверка достаточности средств

#### 1.3. TransactionAmountDescriptor

- **Наследует:** FinancialDescriptor
- **Особенности:**
  - Автоматический расчет комиссии (1%)
  - Проверка минимальной суммы транзакции (0.01)
  - Учет комиссии при проверке достаточности средств

#### 1.4. CategoryDescriptor

- **Наследует:** FinancialDescriptor
- **Особенности:**
  - Валидация категорий расходов (фиксированный набор)
  - Контроль месячных лимитов по категориям
  - Метод для установки лимитов категорий

### 2. Основной класс FinancialAccount

## 2.1. Свойства экземпляра (Properties)

- `age_days` - количество дней с момента создания счета
- `total_commission_paid` - общая сумма уплаченных комиссий
- `monthly_statistics` - статистика за текущий месяц

## 2.2. Методы класса (Class Methods)

- `convert_currency()` - конвертация между валютами
- `set_exchange_rate()` - установка курсов валют
- **Статическое свойство:** словарь курсов валют

## 2.3. Статические методы (Static Methods)

- `_get_current_timestamp()` - получение текущего времени

## 2.4. Методы экземпляра

- `make_transaction()` - проведение транзакции с полной валидацией
- `deposit()` - пополнение счета
- `__init__()` - инициализация с валидацией параметров
- `__str__()` - строковое представление объекта