**Задача 8**

Цели домашнего задания

* Рассмотреть на практике особенности работы с абстрактными классами.
* Отработать навык создания иерархии классов банковских карт с помощью наследования и абстрактных классов.

Что нужно сделать

1. Создайте абстрактный класс BankCard, который будет описывать сущность «Банковская карта». Этот класс должен предоставлять свойство «Баланс», а также функции:
   * «Пополнить» — пополняет карту на переданную сумму;
   * «Оплатить» — списывает с карты переданную сумму и возвращает результат типа Boolean;
   * «Получить информацию о балансе»;
   * «Получить информацию о доступных средствах» — возвращает информацию о балансе, кредитном лимите и любых других средствах.
2. Создайте два производных от BankCard класса DebitCard и CreditCard.
3. В классе DebitCard реализуйте функции «Пополнить» и «Получить информацию о балансе».
4. Класс CreditCard должен иметь дополнительное свойство «Кредитный лимит». Основные средства кредитной карты состоят из двух частей: кредитная часть и собственные средства.  
   При пополнении кредитной карты необходимо сначала погасить кредитную часть, и только затем увеличивать собственные средства.  
   При оплате сначала списываются собственные средства, затем кредитные.

**Пример**

Кредитная карта с лимитом 10 000.

Кредитные средства: 10 000.

Собственные средства: 0.

После пополнения карты на 5 000:

Кредитные средства: 10 000.

Собственные средства: 5 000.

   После оплаты на 5 000:

   Кредитные средства: 10 000.

   Собственные средства: 0.

   После оплаты на 3 000:

   Кредитные средства: 7 000.

   Собственные средства: 0.

   После пополнения на 2 000:

   Кредитные средства: 9 000.

   Собственные средства: 0.

   После пополнения на 2 000:

   Кредитные средства: 10 000.

   Собственные средства: 1 000.

Реализуйте функции «Пополнить», «Оплатить» и «Получить информацию о балансе». Должны учитываться как собственные, так и кредитные средства.

1. Создайте несколько производных классов от DebitCard и CreditCard.
2. Добавьте для конкретных дебетовых и кредитных карт различные бонусные программы в виде отдельных переменных.

**Примеры**

Бонусные баллы в размере 1% от покупок.

Потенциальный кэшбэк 5% от покупок при условии трат больше 5 000 тыс.

Накопление в размере 0.005% от суммы пополнений.

1. Для конкретных дебетовых и кредитных карт скорректируйте функции «Пополнить» и «Оплатить» так, чтобы они учитывали конкретные бонусные программы.
2. Для конкретных дебетовых и кредитных карт реализуйте функцию «Получение информации обо всех доступных средствах». Функция должна учитывать собственные и кредитные средства, а также соответствующие бонусы.
3. Проанализируйте получившиеся классы. При необходимости перенесите реализацию функций на уровень выше.
4. Убедитесь, что все функции ваших классов работают корректно. Для этого создайте экземпляры классов и проверьте работу каждой из описанных функций.

Что оценивается

* Программа запускается, корректно работает и выводит необходимую информацию.
* Выполнены все пункты задания.
* Соблюдена инкапсуляция, доступны только публичные члены классов.
* Функции соответствуют принципам чёрного ящика, выполняют атомарные операции и независимы друг от друга.
* Функции возвращают неизменяемые коллекции.
* Названия переменных отражают суть данных, на которые они ссылаются.
* Переменные, которые не изменяются в программе, объявлены неизменяемыми.
* Открыты только необходимые для переопределения функции.
* Открыты только необходимые для наследования классы.
* Различные выводы в консоль начинаются с новой строки.