Министерство образования и молодежной политики Свердловской области ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

Отчёт по программе «**Практическая 9B**»

Выполнил: Гребенюков Иван

Группа: ПР-22

Преподаватель: Мирошниченко Г.В

2023



**Задание:**

Разработать систему «Платежи». **Клиент** имеет **Счет** в банке и **Банковскую карту (КК)**. **Клиент** может оплатить **Заказ**, сделать платеж на другой **Счет**, заблокировать **КК** и аннулировать **Cчет**. **Администратор** может заблокировать **КК** за превышение платежа.

**Входные и выходные данные**

**Входные данные**

val client = Client ("Иванов Иван Иванович", "ул. Ленина 1", "89777777777")

val bankAccount = BankAccount ("40814567124556784356", 7000.0)

val bankCard = BankCard ("1234567890123456", bankAccount.accountNumber, 5000.0)

val order = Order (1, 1000.0)

val administrator = Administrator ("Андреев Андрей Андреевич", "ул. Первомайская 1", "89555555555")

val tranfer = Payment (client, order, bankAccount, bankCard, administrator)

**Выходные данные**

bankAccount.GetBalance()

tranfer.ProcessPayment()

bankCard.BlockCard()

tranfer.ProcessPayment()

bankCard.UnBlockCard()

bankAccount.Transfer(4000.0, targetBankAccount)

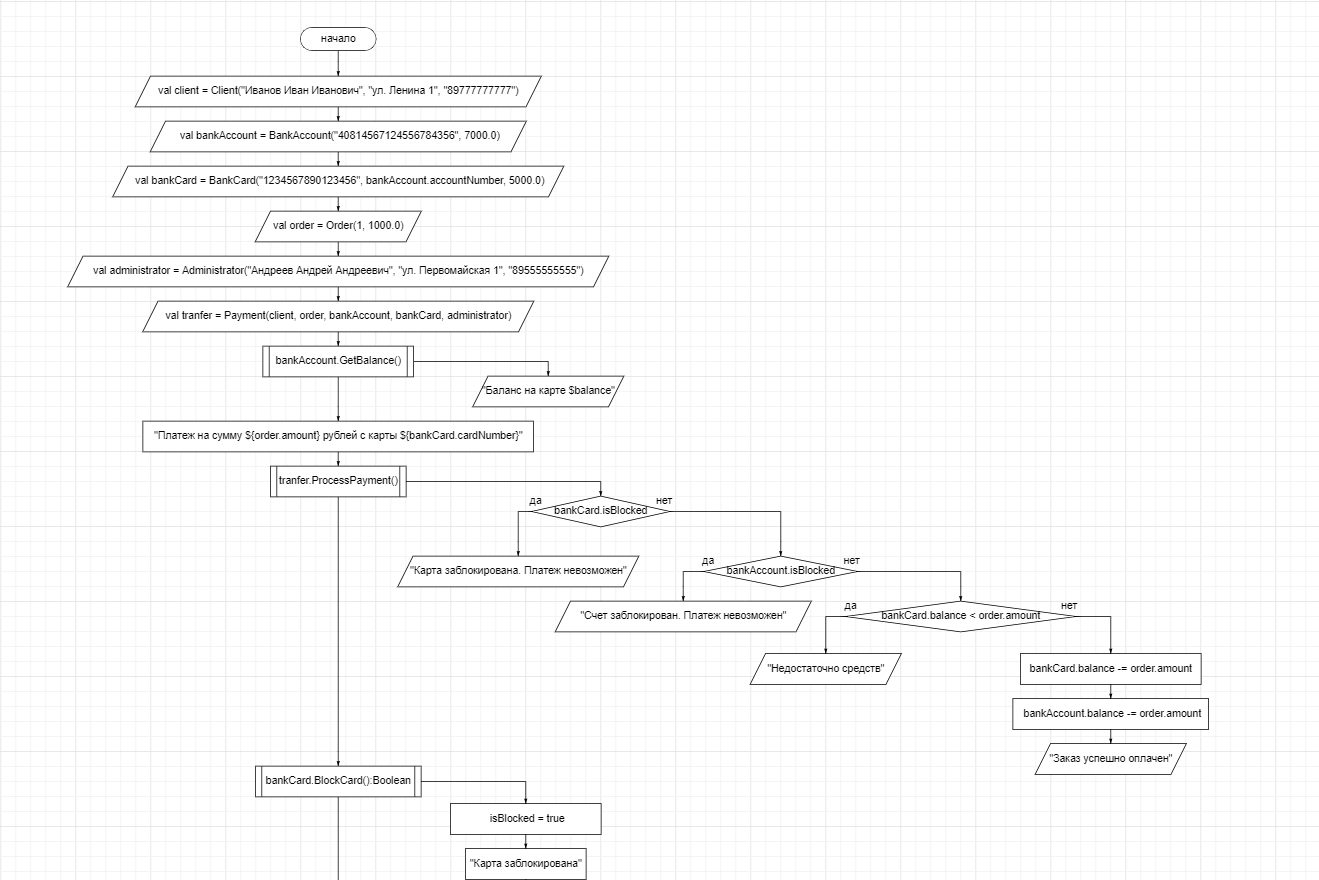
bankAccount.GetBalance()

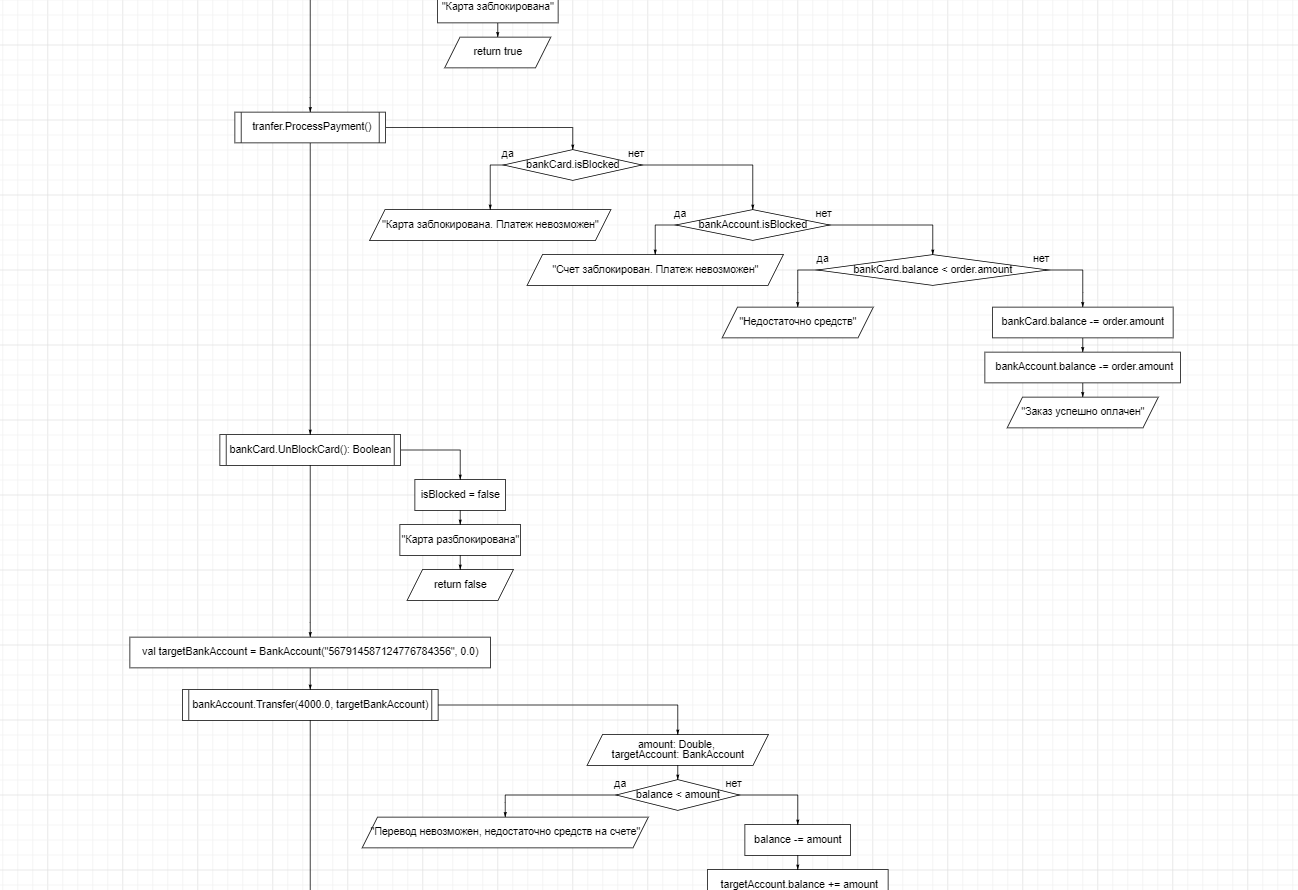
targetBankAccount.GetBalance()

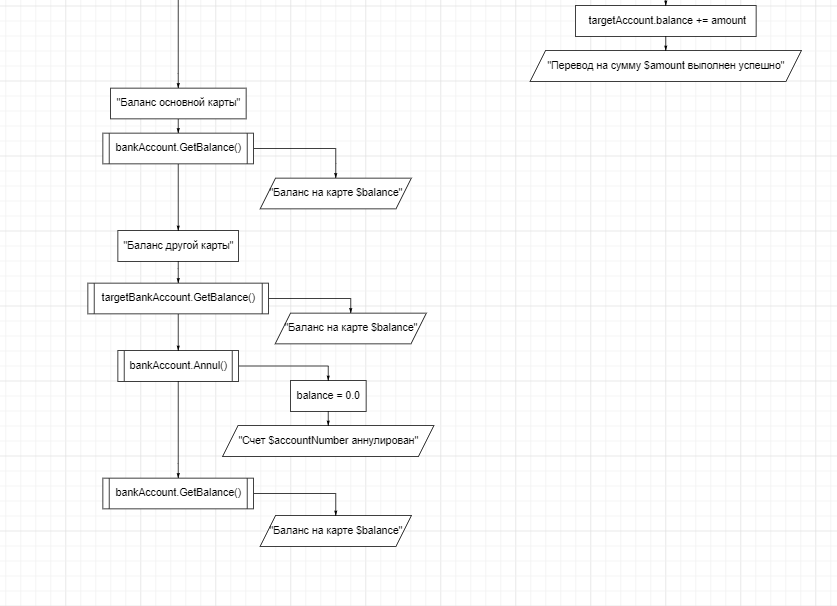
bankAccount.Annul()

bankAccount.GetBalance()

**Блок-схема**

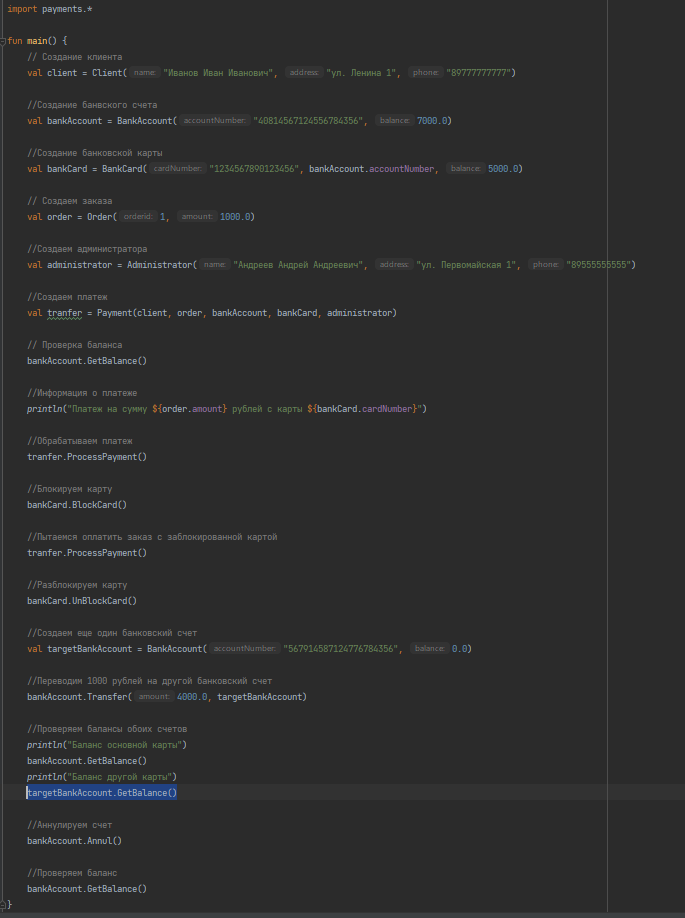




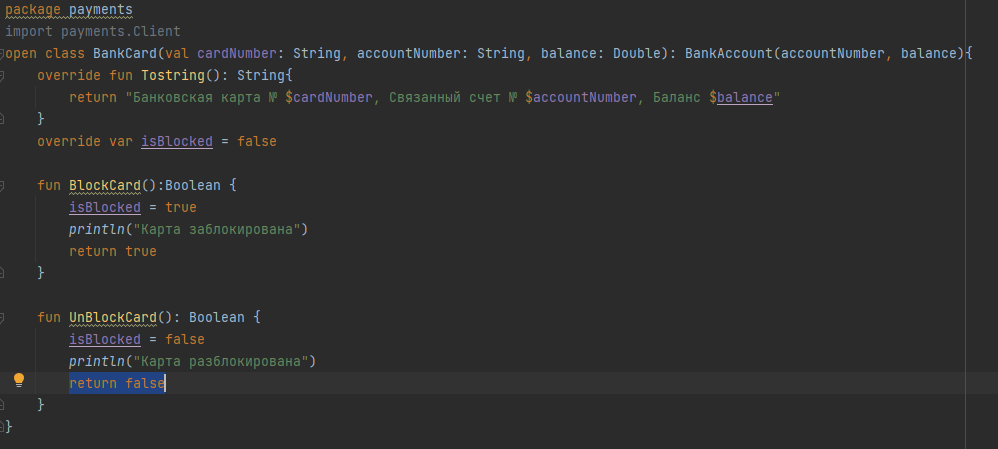


**Листинг программы**

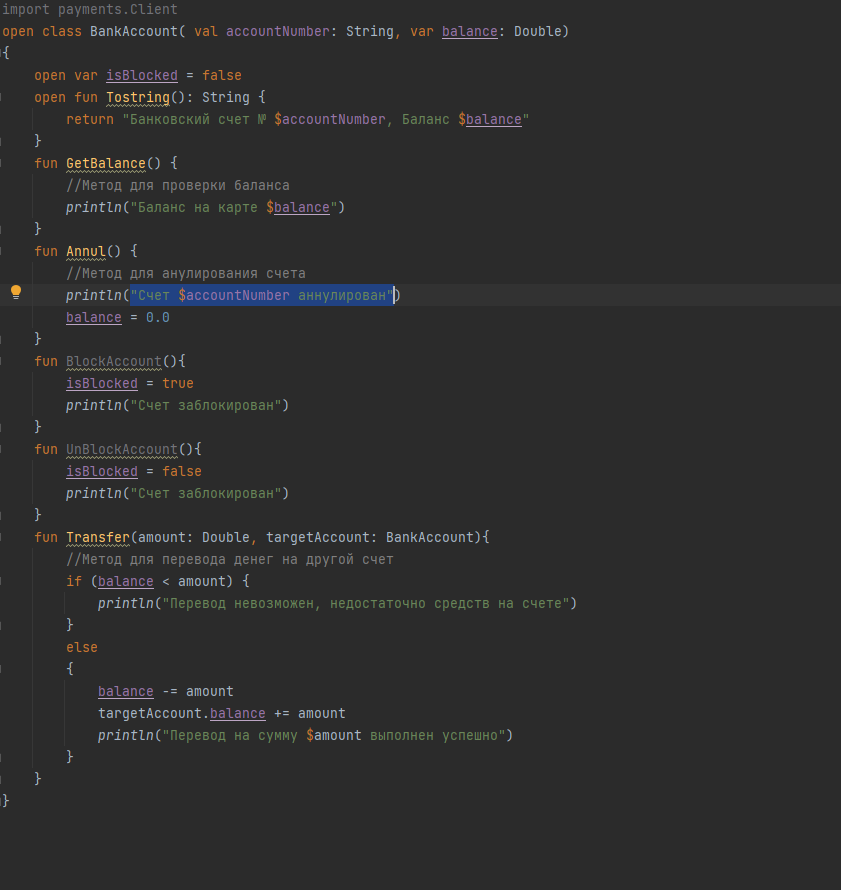
**Основная программа**



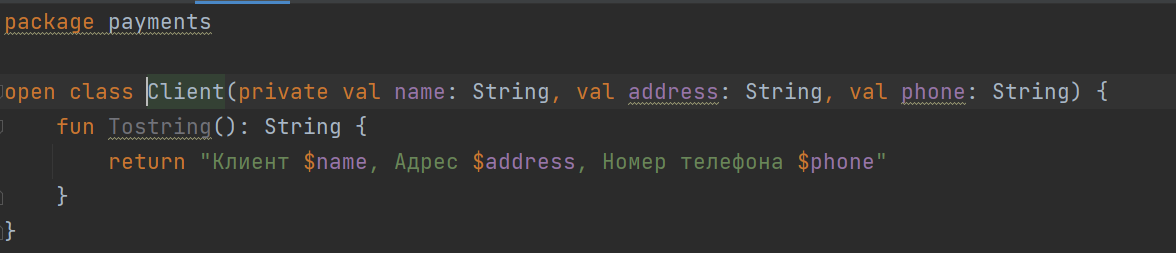
**Класс BankCard**



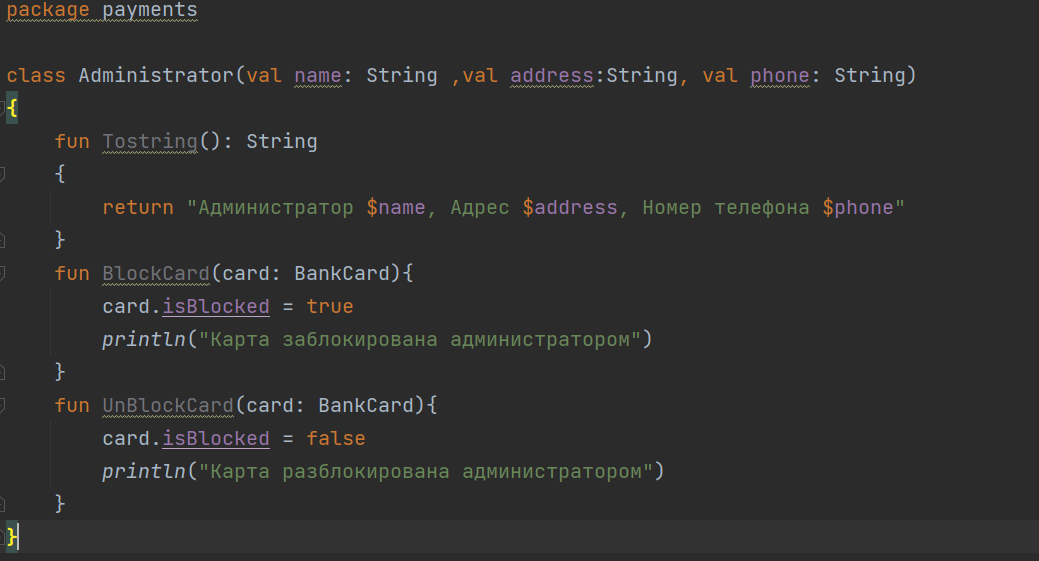
**Класс BankAccount**



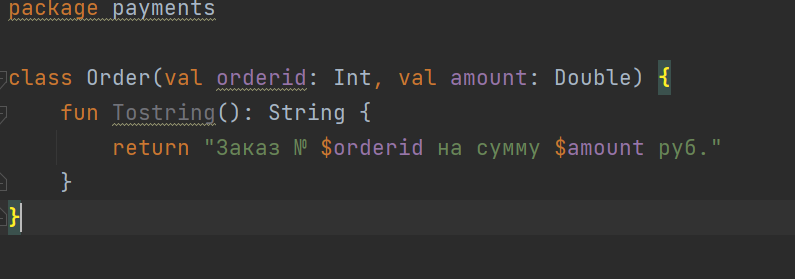
**Класс Client**



**Класс Administrator**

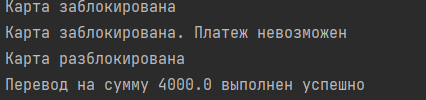


**Класс Order**

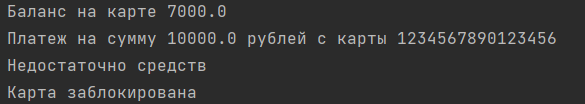


**Тестовые ситуации**

Проверка на обработку платежа, если карта заблокирована



Проверка на обработку платежа, если недостаточно средств



**Вывод**

Закрепил навыки использования классов, создания конструкторов и наследования в ЯП Kotlin.