

Computer Programming 과제 #07

201402414 장수훈

소스코드 및 알고리즘

설명

```
class BankAccount {
    String accountNumber; //계좌번호
    String owner; // 예금주
    int balance; // 잔액표시

    public BankAccount() {
        accountNumber = "";
        owner = "";
        balance = 0;
    }

    void deposit (int amount) {
        balance = balance + amount;
    }
    void withdraw(int amount) {
        balance = balance - amount;
    }
    public String toString() {
        return "잔액 : "+balance;
    }
    public void sendAccount(int amount, BankAccount otherAccount) {
        otherAccount.balance += amount;
        balance -= amount;
    }
}
```

```
import java.util.Scanner;
public class BankAccountTest {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Scanner input = new Scanner(System.in);
        int a;

        BankAccount t1 = new BankAccount();
        BankAccount t2 = new BankAccount();
        System.out.println("계좌1 정보 입력");
        System.out.print("계좌 번호 :");
        t1.accountNumber = input.nextLine();
        System.out.print("예금주 :");
        t1.owner = input.nextLine();
        System.out.print("초기 잔액 : ");
        t1.balance = input.nextInt();

        System.out.println("");

        System.out.println("계좌2 정보 입력");
        System.out.print("계좌 번호 :");
        t2.accountNumber = input.nextLine();
        t2.accountNumber = input.nextLine();
        System.out.print("예금주 :");
        t2.owner = input.nextLine();
        System.out.print("초기 잔액 : ");
        t2.balance = input.nextInt();

        System.out.println("");

        System.out.print("계좌1에서 계좌2로 송금할 금액 :");
        a = input.nextInt();
        t1.sendAccount(a, t2);

        if(t1.balance+1 < a) {
            System.out.println("계좌 잔액이 부족합니다.");
        }
        else {
            System.out.println("=====");
            System.out.println("");

            System.out.println("계좌1");
            System.out.println("계좌번호 : " + t1.accountNumber);
            System.out.println("예금주 : " + t1.owner);
            System.out.println(t1.toString());

            System.out.println("");

            System.out.println("계좌2");
            System.out.println("계좌번호 : " + t2.accountNumber);
            System.out.println("예금주 : " + t2.owner);
            System.out.println(t2.toString());
        }
    }
}
```

BankAccount 클래스에서 계좌번호랑 예금주 잔액표시변수들을 선언하고 생성자를 통해 초기값을 준다. 저금기능 메소드는 잔액에 amount 값을 더해주고 출금기능에서는 잔액에서 amount 값을 빼줌으로 인해서 저금 인출 메소드를 만든다.

잔액 표시 메소드도 만든위 계좌이체 기능 메소드를 만든다. 계좌이체는 객체1인 t1에서 객체2인 t2로 송금하는 기능으로 객체1에서 송금 금액을 빼주고 (balance -= amount;) 객체2에는 송금받은 금액만큼 더해준다. otherAccount.balance += amount;

메인 함수로 넘어와서는 객체를 생성하고 계좌1과 계좌2의 기본 정보들을 입력받아 저장한 후 송금할 금액을 받아 BankAccount클래스를 통해서 저금, 출금, 계좌이체 메소드를 통하여 출력이 된다.

여기서 계좌 이체 송금액이 계좌 잔액보다 큰 경우 계좌이체 실패메시지를 출력해야 함으로 if문을 통하여 계좌의 잔액이 계좌이체할 금액보다 작은 경우 메시지를 의운다. 금액보다 클경우는 계좌이체를 계속한다.

결과 값	결과 값 분석
<div>계좌1 정보 입력 계좌 번호 :110 예금주 :장수훈 초기 잔액 : 100000</div> <div>계좌2 정보 입력 계좌 번호 :111 예금주 :박지원 초기 잔액 : 0</div> <div>계좌1에서 계좌2로 송금할 금액 : 20000 =====</div> <div>계좌1 계좌번호 : 110 예금주 : 장수훈 잔액 : 80000</div> <div>계좌2 계좌번호 : 111 예금주 : 박지원 잔액 : 20000</div>	<p>계좌1과 계좌2에 각각 정보들을 입력한 후 송금 금액을 정한다. 송금 금액을 적어 객채를 통해 송금금액만큼 계좌1에서 빼고 계좌2에 더해준다. 계좌1에 10만원 있던걸 0원이었던 계좌2에 2만원을 송금하여 계좌1에는 8만원이 남고 계좌2은 2만원이 된다.</p>