// Copyright 2003 Bill Campbell and Ethan Bolker

/**
 * A FeeAccount is a BankAccount with one new feature:
 * the user is charged for each transaction.
 *
 * @version 5
 */

public class FeeAccount extends BankAccount
{
private static int transactionFee = 1;

/**
 * Constructor, accepting an initial balance and issuing Bank.
 *
 * @param initialBalance the opening balance.
 * @param issuingBank the bank that issued this account.
 */
public FeeAccount( int initialBalance, Bank issuingBank )
{
super( initialBalance, issuingBank);
}

/**
 * The way a transaction is counted for a Feeaccount: it levies
 * a transaction fee as well as counting the transaction.
 */
public void countTransaction()
{
incrementBalance( - transactionFee );
super.countTransaction();
}

/**
 * Action to take when a new month starts.
 */
public void newMonth()
{
}

/**
 * A FeeAccount extends a BankAccount.
 */
public class FeeAccount extends BankAccount
{
private static int transactionFee = 1;

/**
 * Constructor, accepting an initial balance and issuing Bank.
 *
 * @param initialBalance the opening balance.
 * @param issuingBank the bank that issued this account.
 */
public FeeAccount( int initialBalance, Bank issuingBank )
{
super( initialBalance, issuingBank);
}

/**
 * The way a transaction is counted for a Feeaccount: it levies
 * a transaction fee as well as counting the transaction.
 */
public void countTransaction()
{
incrementBalance( - transactionFee );
super.countTransaction();
}

/**
 * Action to take when a new month starts.
 */
public void newMonth()
{
}

/**
 * A FeeAccount extends a BankAccount.
 */
public class FeeAccount extends BankAccount
{