

```

1 // fo1/7/bank/SavingsAccount.java
2 //
3 //
4 // Copyright 2003 Bill Campbell and Ethan Bolker
5
6 /**
7  * A SavingsAccount is a BankAccount that bears interest.
8  * A fee is charged for too many transactions in a month.
9  *
10 * @see BankAccount
11 *
12 * @version 7
13 */
14
15 public class SavingsAccount extends BankAccount
16 {
17     private int transactionsThisMonth;
18
19     /**
20      * Override getTransactionFee() to return a non-zero fee
21      * after the appropriate number of free monthly transactions.
22      *
23      * @return the fee for current transaction.
24      */
25     protected int getTransactionFee()
26     {
27         if (transactionsThisMonth >
28             getIssuingBank().getMaxFreeTransactions()) {
29             return getIssuingBank().getTransactionFee();
30         }
31         else {
32             return 0;
33         }
34     }
35
36     /**
37      * Increment count of transactions, for this account for
38      * this Month and in total and for the issuing Bank, by one.
39      *
40      * @exception InsufficientFundsException when appropriate.
41      */
42     public void countTransaction()
43     {
44         throws InsufficientFundsException
45     }
46     {
47         transactionsThisMonth++;
48         super.countTransaction();
49     }
50
51     /**
52      * Constructor, accepting an initial balance.
53      * @param initialBalance the opening balance.
54      *
55      * @param issuingBank the bank that issued this account.
56

```

```

57     * @exception InsufficientFundsException when appropriate.
58     */
59
60     public SavingsAccount( int initialBalance, Bank issuingBank )
61     {
62         throws InsufficientFundsException
63         super( initialBalance, issuingBank);
64         transactionsThisMonth = 1;
65     }
66
67     /**
68      * A SavingsAccount earns interest each month.
69      *
70      * @exception InsufficientFundsException when appropriate.
71      */
72
73     public void newMonth()
74     {
75         throws InsufficientFundsException
76         {
77             double monthlyRate = getIssuingBank().getInterestRate()/12;
78             incrementBalance( (int)(monthlyRate * getBalance()));
79             transactionsThisMonth = 0;
80         }
81     }

```