```
 1 package com.beginning_java;
 2
 3 import java.util.*;
 4 import java.util.concurrent.TimeUnit;
 5
 6 class User{
 7     String name;
 8     Account account;
 9     Card card;
10     User(String name,Account account,
   Card card){
11         this.name = name;
12         this.account = account;
13         this.card = card;
14     }
15 }
16 class Account{
17     int balance;
18     Account(int balance){
19         this.balance = balance;
20     }
21 }
22 class Card{
23     HashMap<Integer,Integer> idAndPin
    =new HashMap<>();
24     Card(int id,int pin){
25         idAndPin.put(id,pin);
26     }
27 }
```

```
28
29 class ATM{
30     Scanner sc = new Scanner(System.
   in);
31
32     final int STARTING_BALANCE =
   10000;
33     int machineBalance =
   STARTING_BALANCE;
34
35     User user;
36     ATM(User user){
37         this.user = user;
38     }
39
40     void startMachine(){
41         System.out.println("Welcome"
   );
42         try{
43             TimeUnit.MILLISECONDS.
   sleep(500);
44         }catch (Exception ignored){}
45         System.out.println("Insert
   your card and enter your PIN");
46         authenticatePin();
47     }
48     private void authenticatePin(){
49         int enteredPin = sc.nextInt
   ();
```

```
50          if(user.card.idAndPin.get(1
   )==enteredPin){
51              System.out.println("PIN
   validated");
52              try{
53                  TimeUnit.SECONDS.
   sleep(1);
54              }catch (Exception ignored
   ){}
55              selectTransaction();
56          }
57          else{
58              System.out.println("ERROR
    : Invalid PIN");
59              try{
60                  TimeUnit.SECONDS.
   sleep(3);
61              }catch (Exception ignored
   ){}
62              System.out.println("");
63              startMachine();
64          }
65      }
66      private void selectTransaction(){
67          System.out.println("Hello Mr
   . "+user.name);
68          try{
69              TimeUnit.SECONDS.sleep(1
   );
```

```
70          }catch (Exception ignored){}
71          System.out.println("Select
   transaction");
72          System.out.println("Cash
   Withdraw(Enter 1) Cash Deposit(Enter
   2) View Balance(Enter 3)");
73          int selectedTransaction = sc.
   nextInt();
74          if(selectedTransaction==1){
75              withdraw();
76          }
77          else if(selectedTransaction==
   2){
78              deposit();
79          }
80          else if(selectedTransaction==
   3){
81              viewBalance();
82          }
83          else{
84              System.out.println("
   Invalid Option, kindly re-select the
   transaction");
85              selectTransaction();
86          }
87      }
88      private void withdraw(){
89          System.out.println("Enter the
    amount to be withdrawn");
```

```
 90          int withdrawlAmount = sc.
    nextInt();
 91          if(withdrawlAmount % 100 !=
    0){
 92              System.out.println("
    ERROR : Only 100 and 500
    denominations available in machine"
    );
 93              try{
 94                  TimeUnit.SECONDS.
    sleep(2);
 95              }catch (Exception
    ignored){}
 96              System.out.println("");
 97              startMachine();
 98          }
 99          else{
100              if(user.account.balance<
    withdrawlAmount){
101                  System.out.println("
    ERROR : Insufficient Balance in
    Account");
102                  try{
103                      TimeUnit.SECONDS
    .sleep(2);
104                  }catch (Exception
    ignored){}
105                  System.out.println(
    "");
```

```
106                     startMachine();
107                 }
108                 else if(machineBalance<
    withdrawlAmount){
109                     System.out.println("
    ERROR : Insufficient Balance in
    Machine");
110                     machineBalance +=
    STARTING_BALANCE;
111                     try{
112                         TimeUnit.SECONDS
    .sleep(2);
113                     }catch (Exception
    ignored){}
114                     System.out.println(
    "");
115                     startMachine();
116                 }
117                 else{
118                     user.account.balance
     -= withdrawlAmount;
119                     machineBalance -=
    withdrawlAmount;
120                     try{
121                         TimeUnit.SECONDS
    .sleep(1);
122                     }catch (Exception
    ignored){}
123                     System.out.println("
```

```
123 Please collect your cash");
124                 try{
125                     TimeUnit.SECONDS
    .sleep(1);
126                 }catch (Exception
    ignored){}
127                 System.out.println("
    Transaction is Successful");
128                 try{
129                     TimeUnit.SECONDS
    .sleep(3);
130                 }catch (Exception
    ignored){}
131                 System.out.println(
    "");
132                 startMachine();
133             }
134         }
135
136     }
137     private void deposit(){
138         System.out.println("Enter
    the amount to be deposited");
139         int depositAmount = sc.
    nextInt();
140         System.out.println("Put the
     "+depositAmount+" cash in the slot"
    );
141         try{
```

```
142             TimeUnit.SECONDS.sleep(2
    );
143         }catch (Exception ignored){}
144         user.account.balance +=
    depositAmount;
145         System.out.println("
    Transaction is Successful");
146         try{
147             TimeUnit.SECONDS.sleep(3
    );
148         }catch (Exception ignored){}
149         System.out.println("");
150         startMachine();
151     }
152     private void viewBalance(){
153         System.out.println("Current
    Account Balance : "+ user.account.
    balance);
154         try{
155             TimeUnit.SECONDS.sleep(1
    );
156         }catch (Exception ignored){}
157         System.out.println("
    Transaction is Successful");
158         try{
159             TimeUnit.SECONDS.sleep(3
    );
160         }catch (Exception ignored){}
161         System.out.println("");
```

```
162             startMachine();
163         }
164 }
165
166 public class Main {
167
168     public static void main(String
    [] args) {
169
170         Account accountA = new
    Account(5000);
171         Card cardA = new Card(1,1234
    );
172         User userA = new User("
    Deepak",accountA,cardA);
173
174         ATM atm = new ATM(userA);
175         atm.startMachine();
176
177     }
178 }
179
```