Address.java

```
Public class Address
{
// The street number and name
Private String street;
// The city in which the address is located
Private String city;
// The state in which the address is located
Private String state;
// The zip code associated with the city and street
Private String zip;
/**
Constructor
@param road Describes the street number and name.
@param town Describes the city.
@param st Describes the state.
@param zipCode Describes the zip code.
*/
Public Address(String road, String town, String st,
String zipCode)
{
Street = road;
City = town;
State = st;
Zip = zipCode;
}
Public String toString()
```

```
{
Return (street + “, “ + city +
“, “ + state + “ “ + zip);
}
}
```

Money.java

```
public class Money
{
 private long dollars;
 private long cents;
 public Money(double amount)
 {
 if (amount < 0)
 {
 System.out.println("Error: Negative amounts " +
 "of money are not allowed.");
 System.exit(0);
 }
 else
 {
 long allCents = Math.round(amount * 100);
 dollars = allCents / 100;
 cents = allCents % 100;
 }
 }
 public Money (Money money) {
            dollars= money.dollars;
            cents=money.cents;
```

```
 }
 public Money add(Money otherAmount)
 {
 Money sum = new Money(0);
 sum.cents = this.cents + otherAmount.cents;
 long carryDollars = sum.cents / 100;
 sum.cents = sum.cents % 100;
 sum.dollars = this.dollars +
 otherAmount.dollars +
carryDollars;
 return sum;
 }
 public Money subtract (Money amount)
 {
 Money difference = new Money(0);
 if (this.cents < amount.cents)
 {
 this.dollars = this.dollars - 1;
 this.cents = this.cents + 100;
 }
 difference.dollars = this.dollars - amount.dollars;
 difference.cents = this.cents - amount.cents;
 return difference;
 }
 public int compareTo(Money amount)
 {
 int value;
 if(this.dollars < amount.dollars)
 value = -1;
 else if (this.dollars > amount.dollars)
 value = 1;
 else if (this.cents < amount.cents)
```

```
 value = -1;
 else if (this.cents > amount.cents)
 value = 1;
 else
 value = 0;
 return value;
 }
 public boolean equals(Money money) {
          return (dollars==money.dollars && cents==money.cents);
 }
 public String toString() {
          String temp="$"+dollars;
          if(cents<10) {
                   temp=temp+".0"+cents;
          }
          else {
                   temp=temp+"."+cents;
          }
          return temp;
 }
}
```

MoneyDemo.java

```
/**
 This program demonstrates the Money class.
*/
```

```
public class MoneyDemo
{
 public static void main(String[] args)
 {
 // Named constants
 final int BEGINNING = 500; // Beginning balance
 final Money FIRST_AMOUNT = new Money(10.02);
 final Money SECOND_AMOUNT = new Money(10.02);
 final Money THIRD_AMOUNT = new Money(10.88);
 // Create an instance of the Money class with
 // the beginning balance.
 Money balance = new Money(BEGINNING);
 // Display the current balance.
 System.out.println("The current amount is " +
 balance.toString());
 // Add the second amount to the balance
 // and display the results.
 balance = balance.add(SECOND_AMOUNT);
 System.out.println("Adding " + SECOND_AMOUNT +
 " gives " + balance.toString());
 // Subtract the third amount from the balance
 // and display the results.
 balance = balance.subtract(THIRD_AMOUNT);
 System.out.println("Subtracting " + THIRD_AMOUNT +
 " gives " + balance.toString());
 // Determine if the second amount equals
 // the first amount and store the result.
 boolean equal = SECOND_AMOUNT.equals(FIRST_AMOUNT);
 // Display the result.
 if(equal)
 {
 // The first and second amounts are equal.
```

```
 System.out.println(SECOND_AMOUNT + " equals " +
 FIRST_AMOUNT);
 }
 else
 {
 // The first and second amounts are not equal.
 System.out.println(SECOND_AMOUNT +
 " does not equal " +
FIRST_AMOUNT);
 }
 // Determine if the third amount equals
 // the first amount and store the result.
 equal = THIRD_AMOUNT.equals(FIRST_AMOUNT);
 // Display the result.
 if(equal)
 {
 // The third and first amounts are equal.
 System.out.println(THIRD_AMOUNT + " equals " +
 FIRST_AMOUNT);
 }
 else
 {
 // The third and first amounts are not equal.
 System.out.println(THIRD_AMOUNT +
 " does not equal " +
FIRST_AMOUNT);
 }
 }
}
```

Person.java

```
public class Person
{
// The person's last name
private String lastName;
// The person's first name
private String firstName;
// The person's address
private Address home;
/**
Constructor
@param last The person's last name.
@param first The person's first name.
@param residence The person's address.
*/
public Person(String last, String first,
Address residence)
{
lastName = last;
firstName = first;
home = residence;
}

public String toString()
{
return(firstName + " " + lastName +
", " + home.toString());
}
}
```

Creditcard.java

```
public class CreditCard {
private Person owner;
private Money balance;
private Money creditLimit;
public CreditCard(Person owner, Money creditLimit) {
	super();
	this.owner = owner;
	this.creditLimit =new Money(creditLimit);
	this.balance=new Money(0);
}
public Money getBalance() {
	return new Money( balance);
}
public void setBalance(Money balance) {
	this.balance = balance;
}
public Money getCreditLimit() {
	return new Money(creditLimit);
}
public void setCreditLimit(Money creditLimit) {
	this.creditLimit = creditLimit;
}
public String getOwner() {
	return owner.toString();
```

```
}
public void setOwner(Person owner) {
	this.owner = owner;
}
public void charge(Money anount) {
	Money temp= new Money(balance.add(anount));
	if(temp.compareTo(creditLimit)==1) {
		System.out.println("Vuot qua limit");
	}
	else {
		balance=temp;
	}
}
public void payment(Money anount) {
	balance=balance.subtract(anount);
}
public Person getPersonals() {
	return this.owner;

}

}
```

CreditCardDemo.java

```
public class CreditCardDemo
{
```

```
public static void main(String[] args)
{
// Named constants
final Money CREDIT_LIMIT = new Money(1000);
final Money FIRST_AMOUNT = new Money(200);
final Money SECOND_AMOUNT = new Money(10.02);
final Money THIRD_AMOUNT = new Money(25);
final Money FOURTH_AMOUNT = new Money(990);
// Create an instance of the Person class.
Person owner = new Person("Christie", "Diane",
new Address("237J Harvey Hall",
"Menomonie", "WI", "54751"));
// Create an instance of the CreditCard class.
CreditCard visa = new CreditCard(owner,
CREDIT_LIMIT);
// Display the credit card information.
System.out.println(visa.getPersonals());
System.out.println("Balance: " + visa.getBalance());
System.out.println("Credit Limit: " +
visa.getCreditLimit());
System.out.println(); // To print a new line
// Attempt to charge the first amount and
// display the results.
System.out.println("Attempting to charge " +
FIRST_AMOUNT);
visa.charge(FIRST_AMOUNT);
System.out.println("Balance: " + visa.getBalance());
System.out.println(); // To print a new line
// Attempt to charge the second amount and
// display the results.
System.out.println("Attempting to charge " +
SECOND_AMOUNT);
```

```
visa.charge(SECOND_AMOUNT);
System.out.println("Balance: " + visa.getBalance());
System.out.println(); // To print a new line
//Attempt to pay using the third amount and
//display the results.
System.out.println("Attempting to pay " +
THIRD_AMOUNT);
visa.payment(THIRD_AMOUNT);
System.out.println("Balance: " + visa.getBalance());
System.out.println(); // To print a new line
//Attempt to charge using the fourth amount and
//display the results.
System.out.println("Attempting to charge " +
FOURTH_AMOUNT);
visa.charge(FOURTH_AMOUNT);
System.out.println("Balance: " + visa.getBalance());
}
}
```