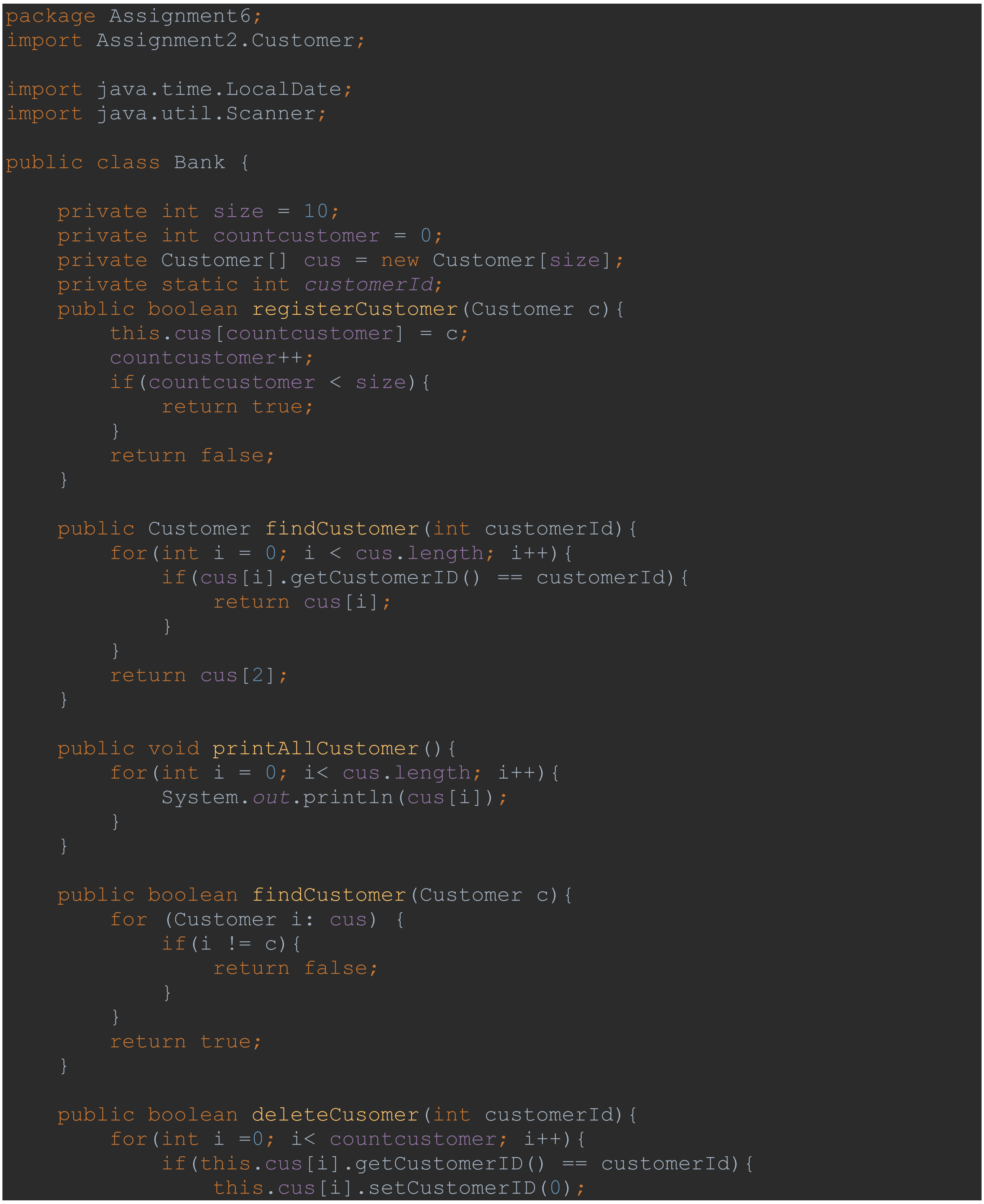
Java Day6

Amar Jyoti Patel

16890

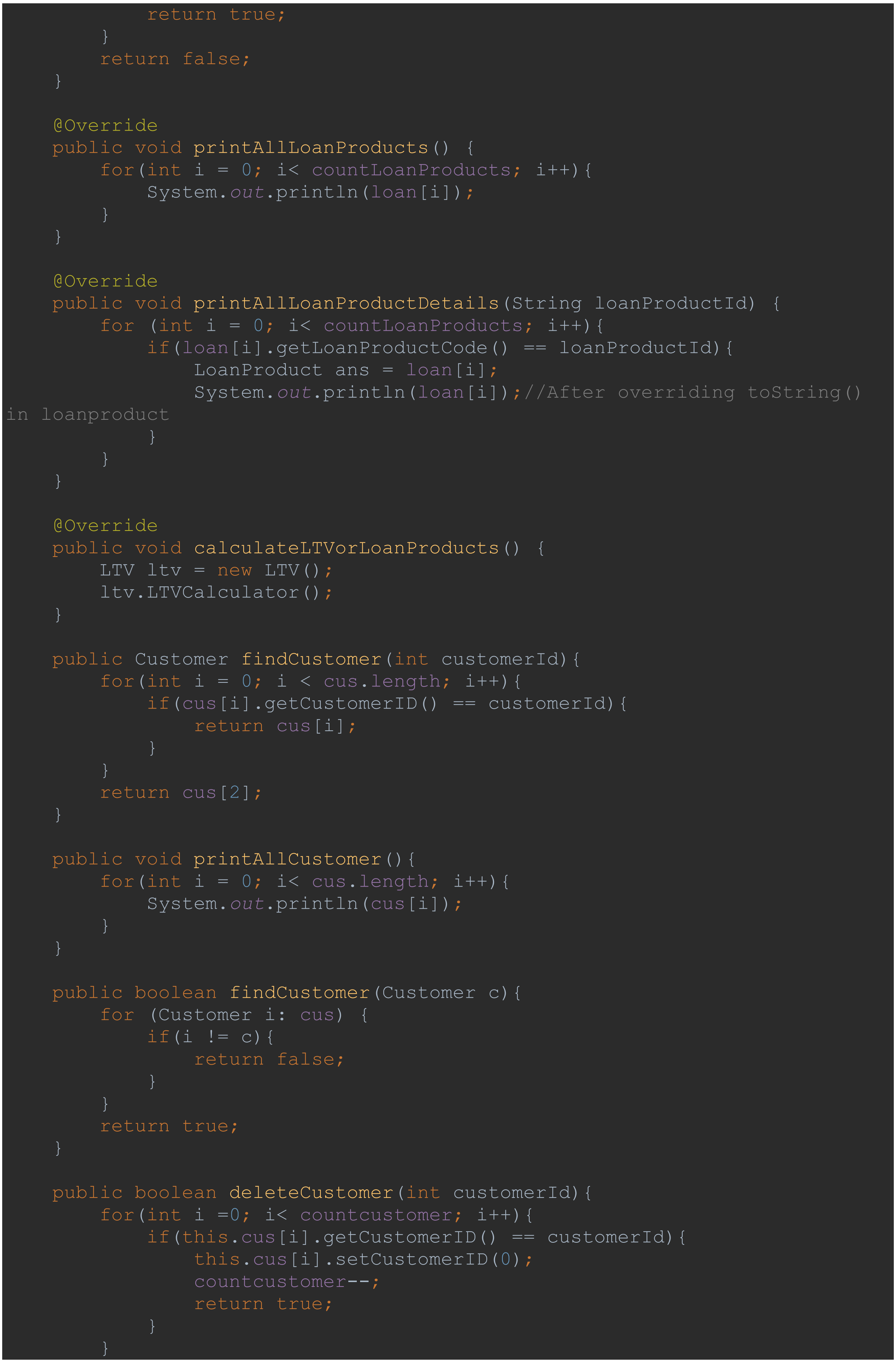
1.



```java
package Assignment6;
import Assignment2.Customer;

import java.time.LocalDate;
import java.util.Scanner;

public class Bank {

    private int size = 10;
    private int countcustomer = 0;
    private Customer[] cus = new Customer[size];
    private static int customerId;
    public boolean registerCustomer(Customer c){
        this.cus[countcustomer] = c;
        countcustomer++;
        if(countcustomer < size){
            return true;
        }
        return false;
    }

    public Customer findCustomer(int customerId){
        for(int i = 0; i < cus.length; i++){
            if(cus[i].getCustomerID() == customerId){
                return cus[i];
            }
        }
        return cus[2];
    }

    public void printAllCustomer(){
        for(int i = 0; i< cus.length; i++){
            System.out.println(cus[i]);
        }
    }

    public boolean findCustomer(Customer c){
        for (Customer i: cus) {
            if(i != c){
                return false;
            }
        }
        return true;
    }

    public boolean deleteCusomer(int customerId){
        for(int i =0; i< countcustomer; i++){
            if(this.cus[i].getCustomerID() == customerId){
                this.cus[i].setCustomerID(0);
                countcustomer--;
                return true;
            }
        }
        return false;
    }

    public int getSize(){
        return size;
    }

    public void setSize(int size){
        this.size = size;
    }

   public int getCustomerId(){
        return customerId;
   }

   public void setCustomerId(int customerId){
        this.customerId = customerId;
   }

   public int getCountcustomer(){
        return countcustomer;
   }

   public void setCus(Customer[] cus){
        this.cus = cus;
   }


}
```

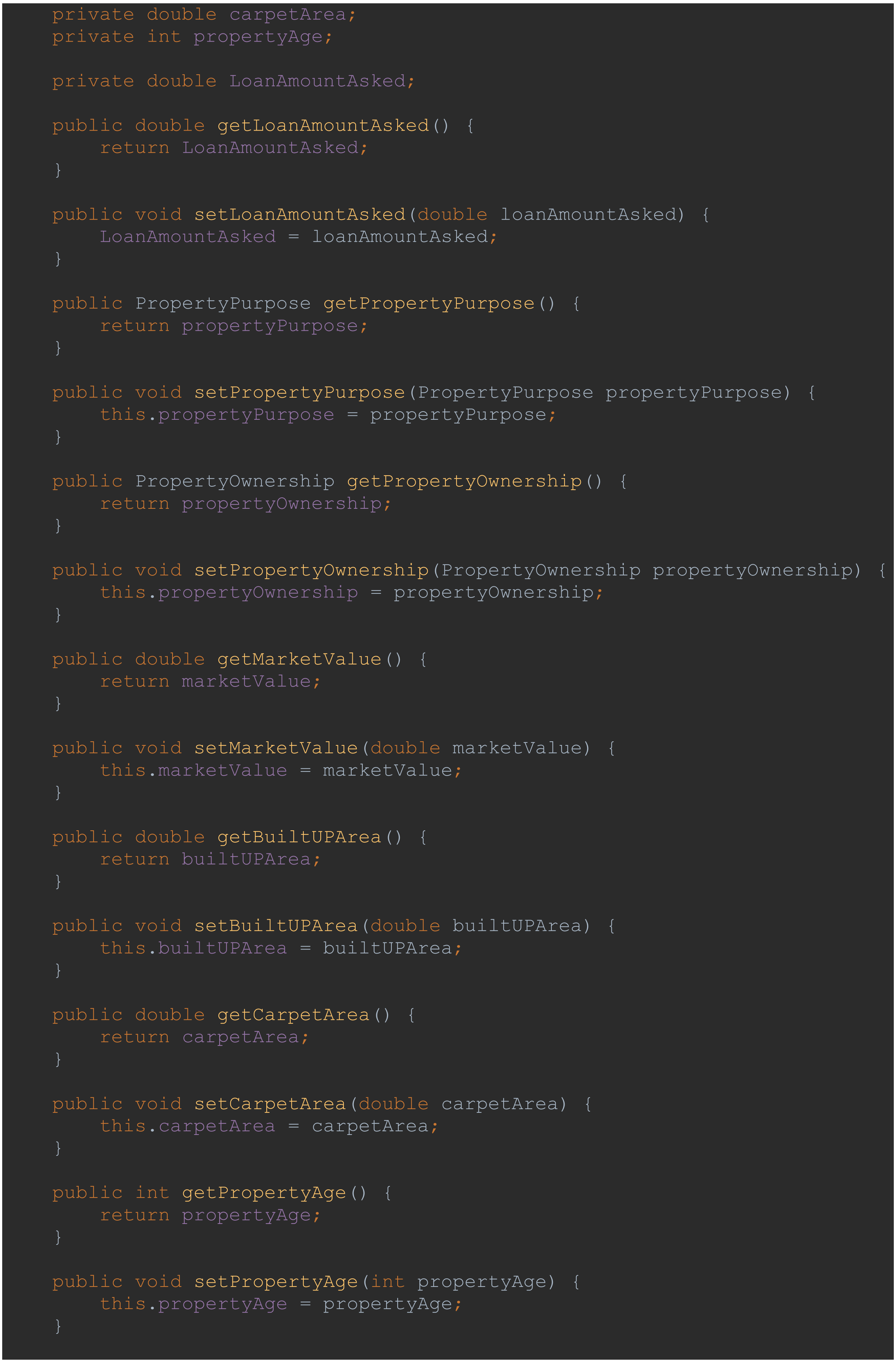
2.

```
package Assignment6;
import Assignment1.LTV;
import Assignment2.Customer;
import Assignment2.LoanProduct;

import java.time.LocalDate;
import java.util.Scanner;

public class Bank implements Maker,Operator{

    private int size = 10;
    private int countcustomer = 0;

    private int countLoanProducts = 0;

    private LoanProduct[] loan = new LoanProduct[size];
    private Customer[] cus = new Customer[size];
    private static int customerId;
    public boolean registerCustomer(Customer c){
        this.cus[countcustomer] = c;
        countcustomer++;
        if(countcustomer < size){
```

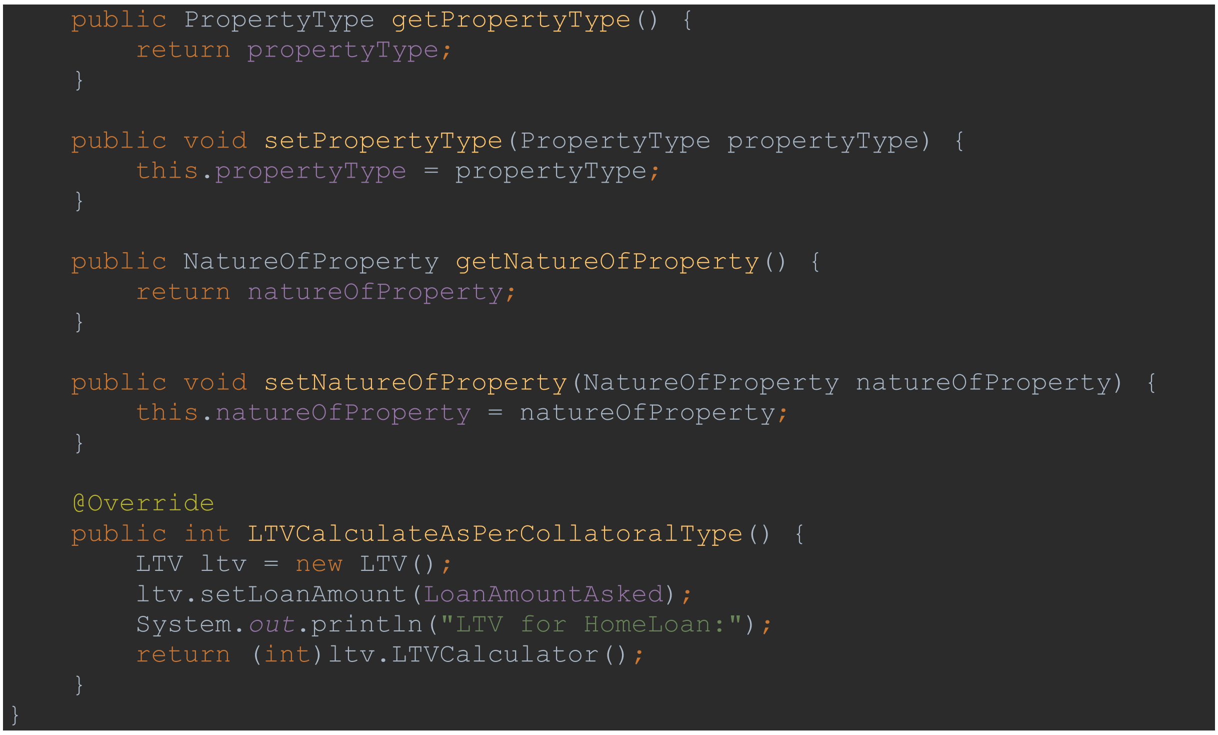


```
            return true;
        }
        return false;
    }

    @Override
    public void printAllLoanProducts() {
        for(int i = 0; i< countLoanProducts; i++){
            System.out.println(loan[i]);
        }
    }

    @Override
    public void printAllLoanProductDetails(String loanProductId) {
        for (int i = 0; i< countLoanProducts; i++){
            if(loan[i].getLoanProductCode() == loanProductId){
                LoanProduct ans = loan[i];
                System.out.println(loan[i]);//After overriding toString()
in loanproduct
            }
        }
    }

    @Override
    public void calculateLTVorLoanProducts() {
        LTV ltv = new LTV();
        ltv.LTVCalculator();
    }

    public Customer findCustomer(int customerId){
        for(int i = 0; i < cus.length; i++){
            if(cus[i].getCustomerID() == customerId){
                return cus[i];
            }
        }
        return cus[2];
    }

    public void printAllCustomer(){
        for(int i = 0; i< cus.length; i++){
            System.out.println(cus[i]);
        }
    }

    public boolean findCustomer(Customer c){
        for (Customer i: cus) {
            if(i != c){
                return false;
            }
        }
        return true;
    }

    public boolean deleteCustomer(int customerId){
        for(int i =0; i< countcustomer; i++){
            if(this.cus[i].getCustomerID() == customerId){
                this.cus[i].setCustomerID(0);
                countcustomer--;
                return true;
            }
        }
```

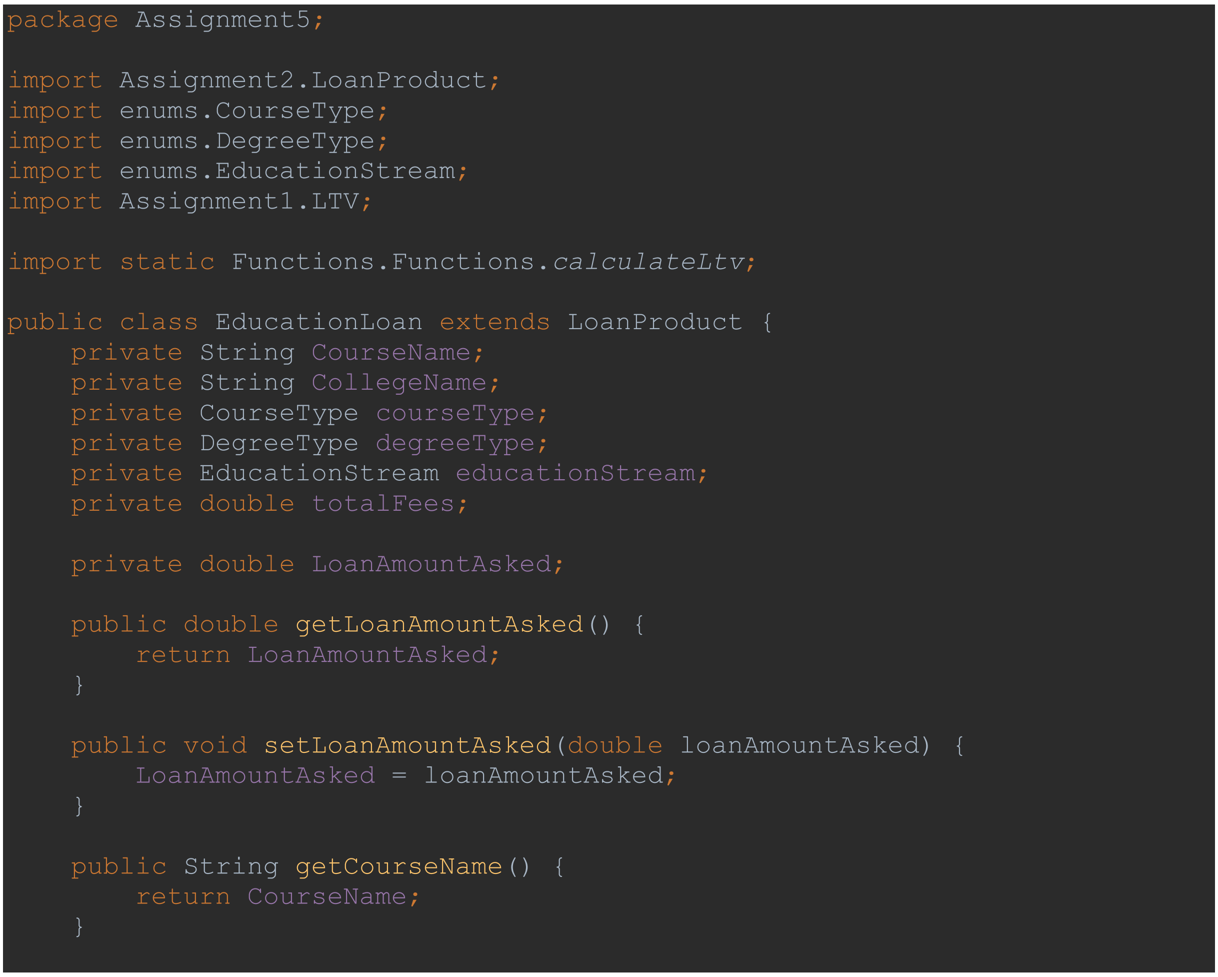
```java
        return false;
    }

    public boolean addNewLoanProduct(){
        return true;
    }

    @Override
    public boolean removeLoanProduct(String loanProductCode) {
        for(int i = 0; i < countLoanProducts; i++){
            if(loan[i].getLoanProductCode() == loanProductCode){
                loan[i].setLoanProductCode(null);
            }
        }
        countLoanProducts--;
        return true;
    }

    public int getSize(){
        return size;
    }

    public void setSize(int size){
        this.size = size;
    }

   public int getCustomerId(){
        return customerId;
   }

   public void setCustomerId(int customerId){
        this.customerId = customerId;
   }

   public int getCountcustomer(){
        return countcustomer;
   }

   public void setCus(Customer[] cus){
        this.cus = cus;
   }


}

package Assignment5;

import Assignment1.LTV;
import Assignment2.LoanProduct;
import enums.*;
public class HomeLoan extends LoanProduct {

    private PropertyType propertyType;
    private NatureOfProperty natureOfProperty;

    private PropertyPurpose propertyPurpose;

    private PropertyOwnership propertyOwnership;

    private double marketValue;
    private double builtUPArea;
```



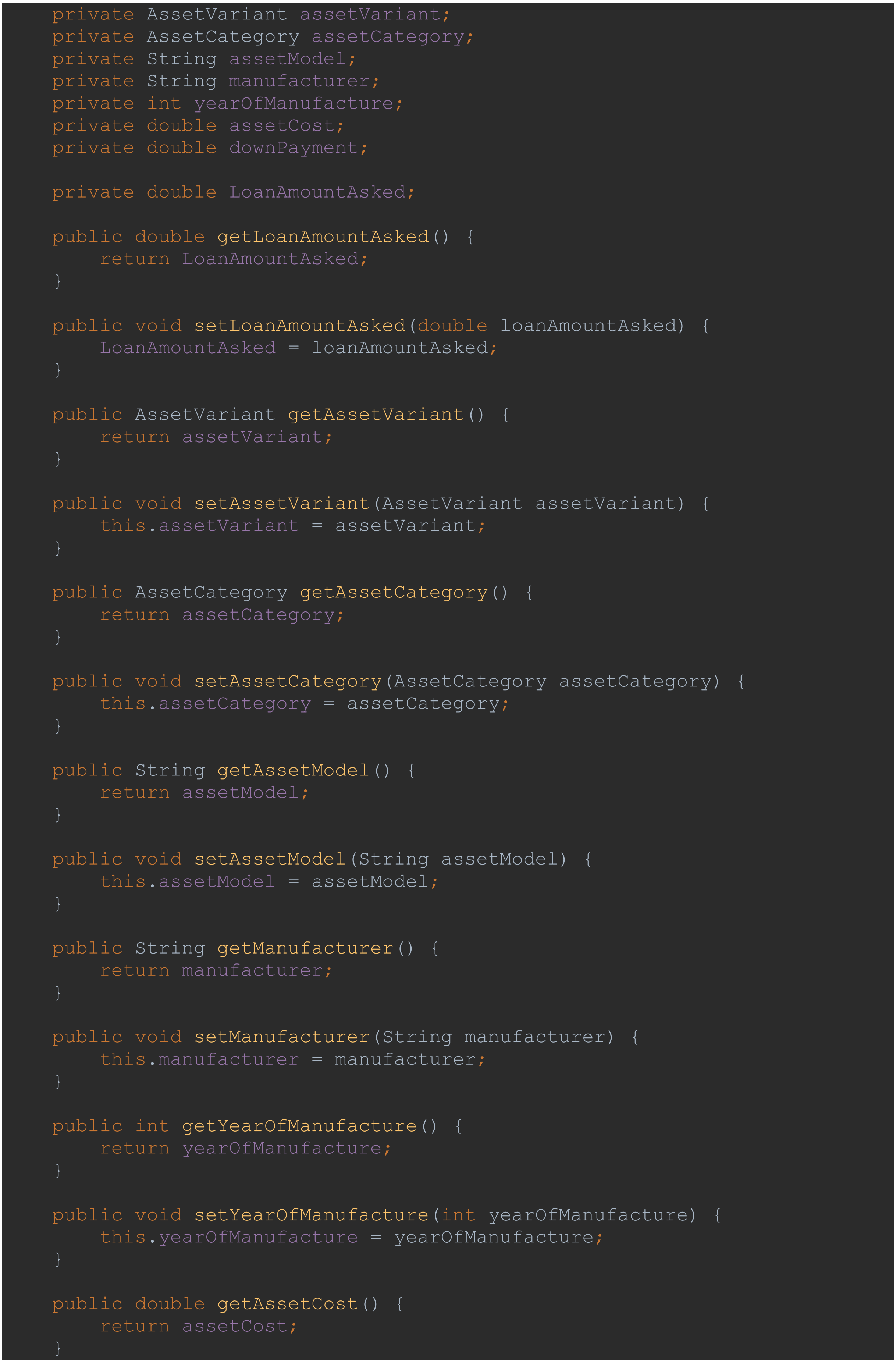
```
    private double carpetArea;
    private int propertyAge;

    private double LoanAmountAsked;

    public double getLoanAmountAsked() {
        return LoanAmountAsked;
    }

    public void setLoanAmountAsked(double loanAmountAsked) {
        LoanAmountAsked = loanAmountAsked;
    }

    public PropertyPurpose getPropertyPurpose() {
        return propertyPurpose;
    }

    public void setPropertyPurpose(PropertyPurpose propertyPurpose) {
        this.propertyPurpose = propertyPurpose;
    }

    public PropertyOwnership getPropertyOwnership() {
        return propertyOwnership;
    }

    public void setPropertyOwnership(PropertyOwnership propertyOwnership) {
        this.propertyOwnership = propertyOwnership;
    }

    public double getMarketValue() {
        return marketValue;
    }

    public void setMarketValue(double marketValue) {
        this.marketValue = marketValue;
    }

    public double getBuiltUPArea() {
        return builtUPArea;
    }

    public void setBuiltUPArea(double builtUPArea) {
        this.builtUPArea = builtUPArea;
    }

    public double getCarpetArea() {
        return carpetArea;
    }

    public void setCarpetArea(double carpetArea) {
        this.carpetArea = carpetArea;
    }

    public int getPropertyAge() {
        return propertyAge;
    }

    public void setPropertyAge(int propertyAge) {
        this.propertyAge = propertyAge;
    }
```



```java
    public PropertyType getPropertyType() {
        return propertyType;
    }

    public void setPropertyType(PropertyType propertyType) {
        this.propertyType = propertyType;
    }

    public NatureOfProperty getNatureOfProperty() {
        return natureOfProperty;
    }

    public void setNatureOfProperty(NatureOfProperty natureOfProperty) {
        this.natureOfProperty = natureOfProperty;
    }

    @Override
    public int LTVCalculateAsPerCollatoralType() {
        LTV ltv = new LTV();
        ltv.setLoanAmount(LoanAmountAsked);
        System.out.println("LTV for HomeLoan:");
        return (int)ltv.LTVCalculator();
    }
}
```



```java
package Assignment5;

import Assignment2.LoanProduct;
import enums.CourseType;
import enums.DegreeType;
import enums.EducationStream;
import Assignment1.LTV;

import static Functions.Functions.calculateLtv;

public class EducationLoan extends LoanProduct {
    private String CourseName;
    private String CollegeName;
    private CourseType courseType;
    private DegreeType degreeType;
    private EducationStream educationStream;
    private double totalFees;

    private double LoanAmountAsked;

    public double getLoanAmountAsked() {
        return LoanAmountAsked;
    }

    public void setLoanAmountAsked(double loanAmountAsked) {
        LoanAmountAsked = loanAmountAsked;
    }

    public String getCourseName() {
        return CourseName;
    }
    public void setCourseName(String courseName) {
        CourseName = courseName;
    }

    public String getCollegeName() {
        return CollegeName;
    }

    public void setCollegeName(String collegeName) {
        CollegeName = collegeName;
    }

    public CourseType getCourseType() {
        return courseType;
    }

    public void setCourseType(CourseType courseType) {
        this.courseType = courseType;
    }

    public DegreeType getDegreeType() {
        return degreeType;
    }

    public void setDegreeType(DegreeType degreeType) {
        this.degreeType = degreeType;
    }

    public EducationStream getEducationStream() {
        return educationStream;
    }

    public void setEducationStream(EducationStream educationStream) {
        this.educationStream = educationStream;
    }

    public double getTotalFees() {
        return totalFees;
    }

    public void setTotalFees(double totalFees) {
        this.totalFees = totalFees;
    }

    @Override
    public int LTVCalculateAsPerCollatoralType() {
        LTV ltv = new LTV();
        ltv.setLoanAmount(LoanAmountAsked);
        System.out.println("LTv for EducationLoan:");
        return (int)ltv.LTVCalculator();
    }
}
package Assignment5;

import Assignment1.LTV;
import Assignment2.LoanProduct;
import enums.AssetCategory;
import enums.AssetVariant;

public class ConsumerVechileLoan extends LoanProduct {
```



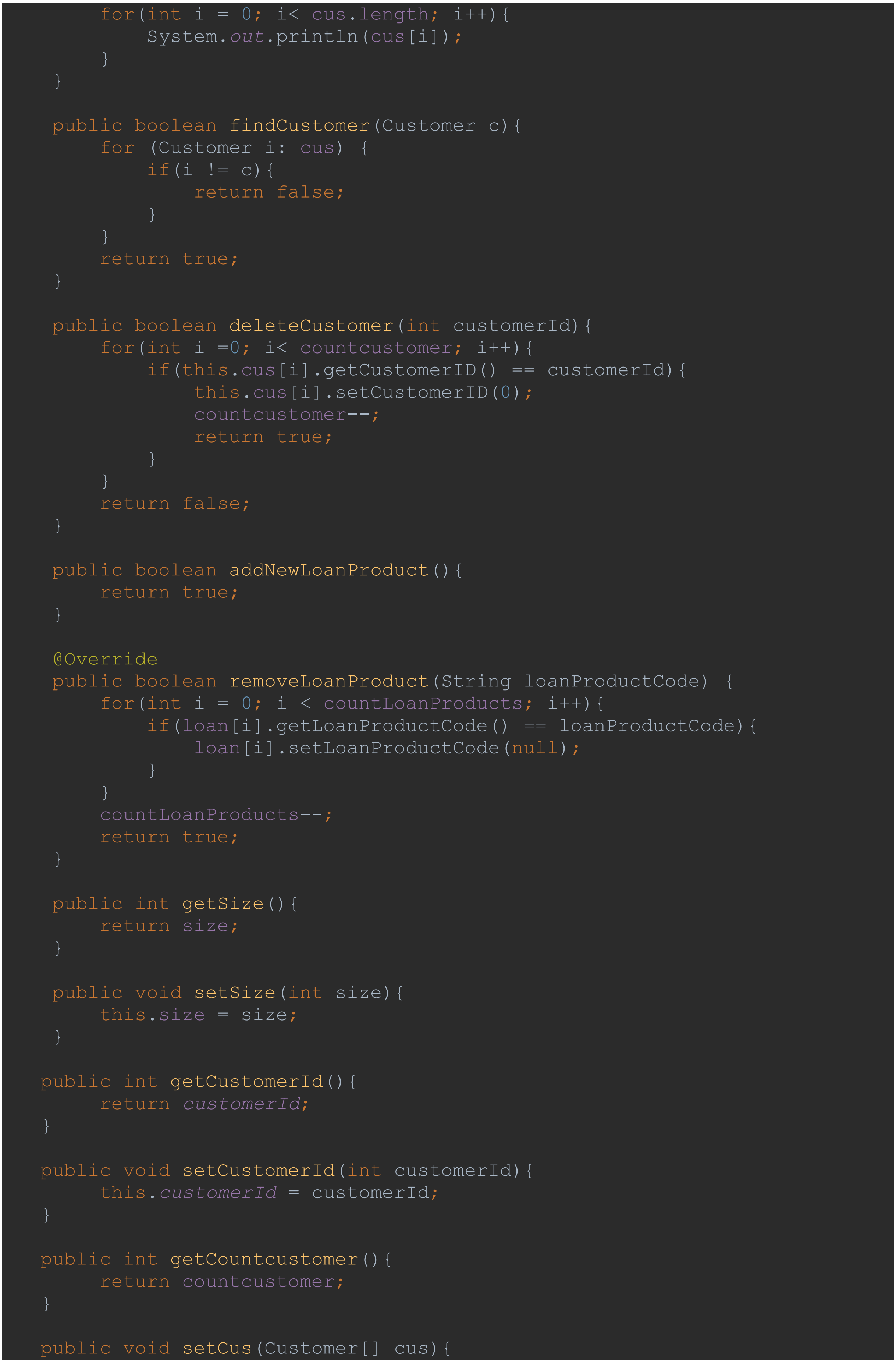
```java
    private AssetVariant assetVariant;
    private AssetCategory assetCategory;
    private String assetModel;
    private String manufacturer;
    private int yearOfManufacture;
    private double assetCost;
    private double downPayment;

    private double LoanAmountAsked;

    public double getLoanAmountAsked() {
        return LoanAmountAsked;
    }

    public void setLoanAmountAsked(double loanAmountAsked) {
        LoanAmountAsked = loanAmountAsked;
    }

    public AssetVariant getAssetVariant() {
        return assetVariant;
    }

    public void setAssetVariant(AssetVariant assetVariant) {
        this.assetVariant = assetVariant;
    }

    public AssetCategory getAssetCategory() {
        return assetCategory;
    }

    public void setAssetCategory(AssetCategory assetCategory) {
        this.assetCategory = assetCategory;
    }

    public String getAssetModel() {
        return assetModel;
    }

    public void setAssetModel(String assetModel) {
        this.assetModel = assetModel;
    }

    public String getManufacturer() {
        return manufacturer;
    }

    public void setManufacturer(String manufacturer) {
        this.manufacturer = manufacturer;
    }

    public int getYearOfManufacture() {
        return yearOfManufacture;
    }

    public void setYearOfManufacture(int yearOfManufacture) {
        this.yearOfManufacture = yearOfManufacture;
    }

    public double getAssetCost() {
        return assetCost;
    }
    public void setAssetCost(double assetCost) {
        this.assetCost = assetCost;
    }

    public double getDownPayment() {
        return downPayment;
    }

    public void setDownPayment(double downPayment) {
        this.downPayment = downPayment;
    }

    @Override
    public int LTVCalculateAsPerCollatoralType() {
        LTV ltv = new LTV();
        ltv.setLoanAmount(LoanAmountAsked);
        System.out.println("LTV for consumerVechileLoan :");
        return (int)ltv.LTVCalculator();
    }
}
```

3.

```
package Assignment6;

import Assignment2.Customer;

public interface Maker {

     boolean registerCustomer(Customer c);
     boolean deleteCustomer(int customerId);

     boolean addNewLoanProduct();
     boolean removeLoanProduct(String loanProductCode);
}
```

```
package Assignment6;

import Assignment2.Customer;

public interface Operator {
     void printAllLoanProducts();
     void printAllLoanProductDetails(String loanProductId);
     void calculateLTVorLoanProducts();
     Customer findCustomer(int customerId);
     boolean findCustomer(Customer c);
     void printAllCustomer();
}
```

```
package Assignment6;
import Assignment1.LTV;
import Assignment2.Customer;
import Assignment2.LoanProduct;

import java.time.LocalDate;
import java.util.Scanner;

public class Bank implements Maker,Operator{

    private int size = 10;
    private int countcustomer = 0;

    private int countLoanProducts = 0;

    private LoanProduct[] loan = new LoanProduct[size];
    private Customer[] cus = new Customer[size];
    private static int customerId;
    public boolean registerCustomer(Customer c){
        this.cus[countcustomer] = c;
        countcustomer++;
        if(countcustomer < size){
            return true;
        }
        return false;
    }

    @Override
    public void printAllLoanProducts() {
        for(int i = 0; i< countLoanProducts; i++){
            System.out.println(loan[i]);
        }
    }

    @Override
    public void printAllLoanProductDetails(String loanProductId) {
        for (int i = 0; i< countLoanProducts; i++){
            if(loan[i].getLoanProductCode() == loanProductId){
                LoanProduct ans = loan[i];
                System.out.println(loan[i]);//After overriding toString()
in loanproduct
            }
        }
    }

    @Override
    public void calculateLTVorLoanProducts() {
        LTV ltv = new LTV();
        ltv.LTVCalculator();
    }

    public Customer findCustomer(int customerId){
        for(int i = 0; i < cus.length; i++){
            if(cus[i].getCustomerID() == customerId){
                return cus[i];
            }
        }
        return cus[2];
    }

    public void printAllCustomer(){
```



```
        for(int i = 0; i< cus.length; i++){
            System.out.println(cus[i]);
        }
    }

    public boolean findCustomer(Customer c){
        for (Customer i: cus)  {
            if(i != c){
                return false;
            }
        }
        return true;
    }

    public boolean deleteCustomer(int customerId){
        for(int i =0; i< countcustomer; i++){
            if(this.cus[i].getCustomerID() == customerId){
                this.cus[i].setCustomerID(0);
                countcustomer--;
                return true;
            }
        }
        return false;
    }

    public boolean addNewLoanProduct(){
        return true;
    }

    @Override
    public boolean removeLoanProduct(String loanProductCode)  {
        for(int i = 0; i < countLoanProducts; i++){
            if(loan[i].getLoanProductCode() == loanProductCode){
                loan[i].setLoanProductCode(null);
            }
        }
        countLoanProducts--;
        return true;
    }

    public int getSize(){
        return size;
    }

    public void setSize(int size){
        this.size = size;
    }

   public int getCustomerId(){
        return customerId;
   }

   public void setCustomerId(int customerId){
        this.customerId = customerId;
   }

   public int getCountcustomer(){
        return countcustomer;
   }

   public void setCus(Customer[] cus){
        this.cus = cus;
    }


}
```