MAKALAH, PERCOBAAN, LATIHAN, DAN TUGAS
MODUL PRAKTIKUM 7

Disusun sebagai salah satu tugas
mata kuliah PBO I

Patricia Joanne
140810160065

Dikumpulkan tanggal
24 Oktober 2017

PROGRAM STUDI S-1 TEKNIK INFORMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS PADJADJARAN
2017

Tugas Pendahuluan

1. Jelaskan kembali pengertian inner class menurut bahasa anda!
2. Buatlah review mengenai tipe Wrapper dan macam-macamnya, dan penggunaan konstruktor dari masing-masing tipe Wrapper.
3. Buatlah review mengenai:
   a. Definisi enum
   b. Merubah bentuk konstanta(terdapat lebih dari satu konstanta) menjadi bentuk enum
   c. Letak yang diperbolehkan dan tidak diperbolehkan untuk pendeklarasian enum

---

1. Inner class merupakan class yang tidak berada pada top level atau class yang dideklarasikan di dalam class lain(Outer class). Sama seperti halnya instance method dan variabel, inner class harus menggunakan instance object dari top level class untuk mengakses object method atau field. Karena inner class menggunakan instance dari top level class, maka membernya tidak dapat dideklarasikan dengan static.

2. Wrapper class merupakan tipe data bawaan Java yang berupa object. Setiap tipe data primitif mempunyai padanan di wrapper class ini. Walaupun wrapper class ini berupa class, tetapi variabel yang memegang objectnya bukanlah variabel reference. Sebab wrapper class ini memiliki sifat Immutable, yaitu apabila ada dua buah variabel yang memegang nilai yang sama, bila satu variabel diganti nilainya, maka variabel yang lain tidak ikut berubah nilainya.

Berikut tipe-tipe data primitif dan Class Wrappernya sekaligus Constructor-nya:

Tipe Data Primitif	Class Wrapper	Argument Constructor
Boolean	Boolean	boolean atau String
Byte	Byte	byte atau String
Char	Character	char

Double	Double	double atau String
Float	Float	float, double, atau String
Int	Integer	int atau String
Long	Long	long atau String
Short	Short	short atau String

3. a) Definisi enum

Enum adalah sebuah tipe data yang nilainya hanya terbatas dari pilihan nilai-nilai yang telah didefinisikan terlebih dahulu. Enumeration di Java ini baru diperkenalkan pada versi Java 5.0.

b) Merubah bentuk konstanta (terdapat lebih dari satu konstanta) menjadi bentuk enum

Contoh:

```
enum Transport{
     Mobil, Truck, Kapal, Kereta, Becak
}
```

Mobil, Kapal dan seterusnya adalah sebuah konstanta dari enumeration. Masing-masing dideklarasikan public jadi bisa diakses secara bebas oleh Transport. Lalu tipe data dari mobil, kapal dan seterusnya itu merupakan tipe nama enumeration yang dideklarasikan. Jadi mobil, kapal dan seterusnya merupakan tipe dari Transport. Oleh karena itu, konstanta enumeration disebut self-type, artinya merupakan tipe dirinya sendiri.

c) Letak yang diperbolehkan dan tidak diperbolehkan untuk pendeklarasian enum

- Letak yang diperbolehkan untuk pendeklarasian enum

  <u>Mendeklarasikan enum di luar class</u>

```
enum CoffeeSize {BIG, HUGE, OVERWHELMING}

public class Coffee {
    CoffeeSize size ;
}
```

```
class CoffeeTest1 {
    public static void main(String[] args) {
         Coffee drink = new Coffee();
         drink.size = CoffeeSize.BIG ;
    }
}
```

<u>Mendeklarasikan enum di dalam class</u>

```
public class Coffee2 {
    num CoffeeSize {BIG, HUGE, OVERWHELMING}
    CoffeeSize size ;
}
class CoffeeTest2{
    public static void main(String[] args) {
         Coffee2 drink = new Coffee2();
         drink.size = Coffee2.CoffeeSize.BIG;
    }
}
```

- Letak yang tidak diperbolehkan untuk pendeklarasian enum

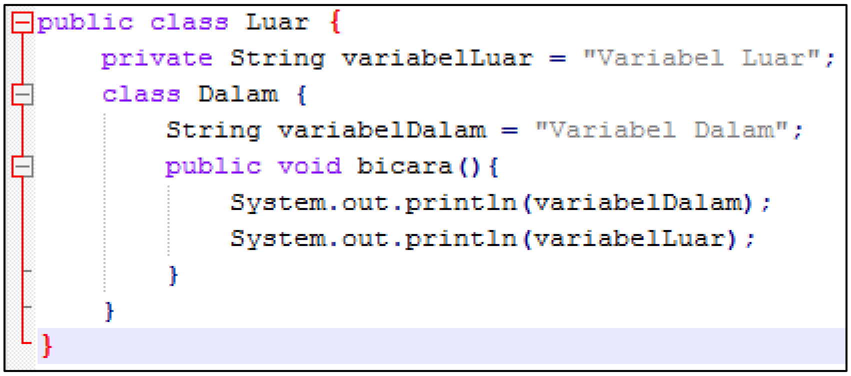
<u>Mendeklarasikan enum di dalam method</u>

```
public class CoffeeTest3 {
    public static void main(String[] args) {
         enum CoffeeSize {BIG, HUGE, OVERWHELMING}
         Coffee drink = new Coffee();
         drink.size = CoffeeSize.BIG;
    }
}
```

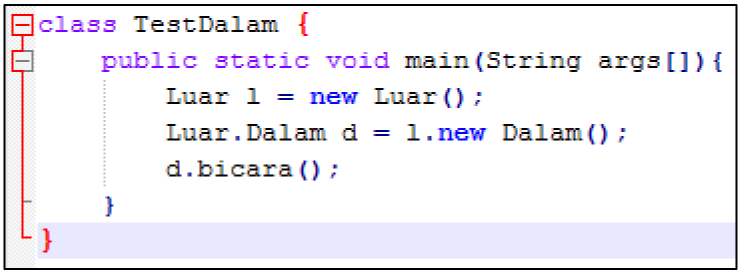
Praktikum 7

Nested Class, Type Wrapper, Type Enum

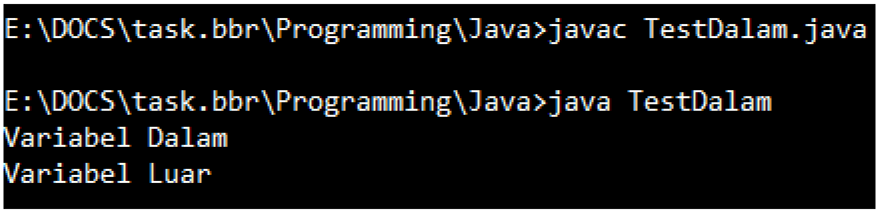
Percobaan 1: Non-Static Inner class yang dideklarasikan di dalam class



```java
public class Luar {
    private String variabelLuar = "Variabel Luar";
    class Dalam {
        String variabelDalam = "Variabel Dalam";
        public void bicara(){
            System.out.println(variabelDalam);
            System.out.println(variabelLuar);
        }
    }
}
```

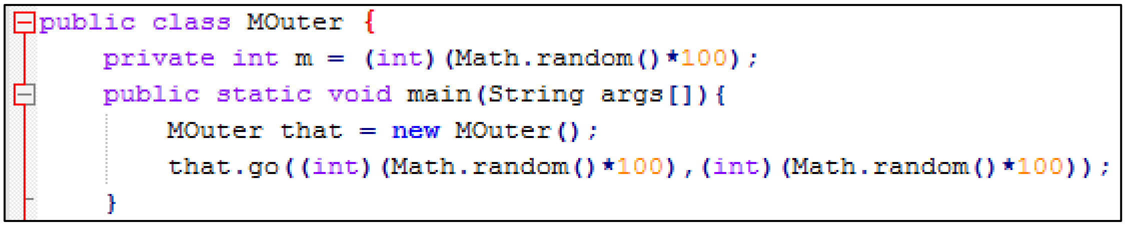


```java
class TestDalam {
    public static void main(String args[]){
        Luar l = new Luar();
        Luar.Dalam d = l.new Dalam();
        d.bicara();
    }
}
```

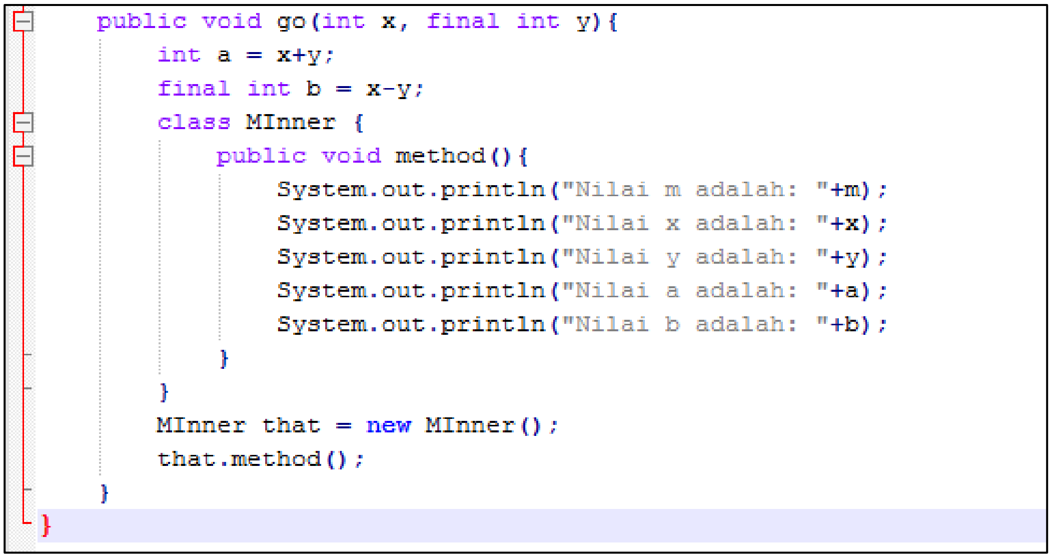


```
E:\DOCS\task.bbr\Programming\Java>javac TestDalam.java

E:\DOCS\task.bbr\Programming\Java>java TestDalam
Variabel Dalam
Variabel Luar
```

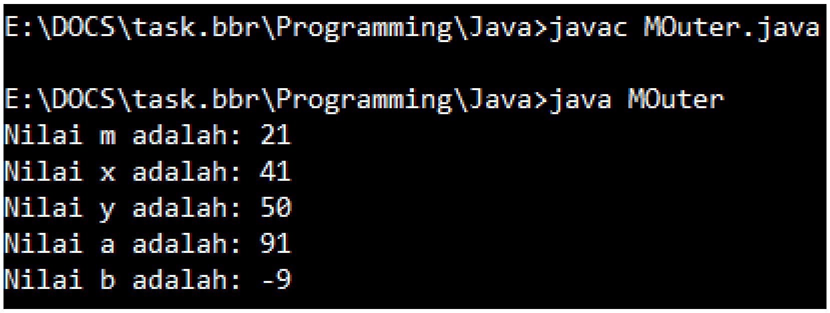
Percobaan 2: Inner class yang dideklarasikan di dalam method



```java
public class MOuter {
    private int m = (int)(Math.random()*100);
    public static void main(String args[]){
        MOuter that = new MOuter();
        that.go((int)(Math.random()*100),(int)(Math.random()*100));
    }
```

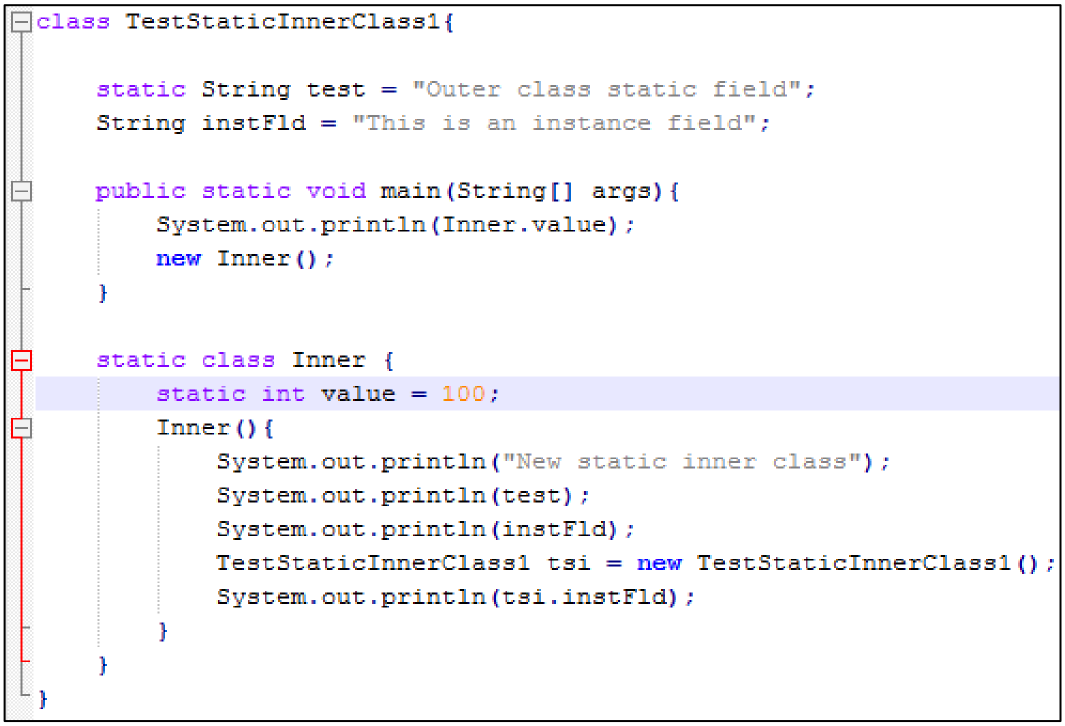


```
    public void go(int x, final int y){
        int a = x+y;
        final int b = x-y;
        class MInner {
            public void method(){
                System.out.println("Nilai m adalah: "+m);
                System.out.println("Nilai x adalah: "+x);
                System.out.println("Nilai y adalah: "+y);
                System.out.println("Nilai a adalah: "+a);
                System.out.println("Nilai b adalah: "+b);
            }
        }
        MInner that = new MInner();
        that.method();
    }
}
```

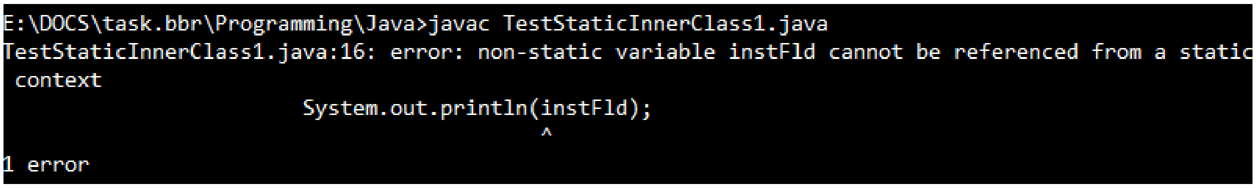


```
E:\DOCS\task.bbr\Programming\Java>javac MOuter.java

E:\DOCS\task.bbr\Programming\Java>java MOuter
Nilai m adalah: 21
Nilai x adalah: 41
Nilai y adalah: 50
Nilai a adalah: 91
Nilai b adalah: -9
```

Percobaan 3: Static Inner class. Perbaiki error yang terjadi!

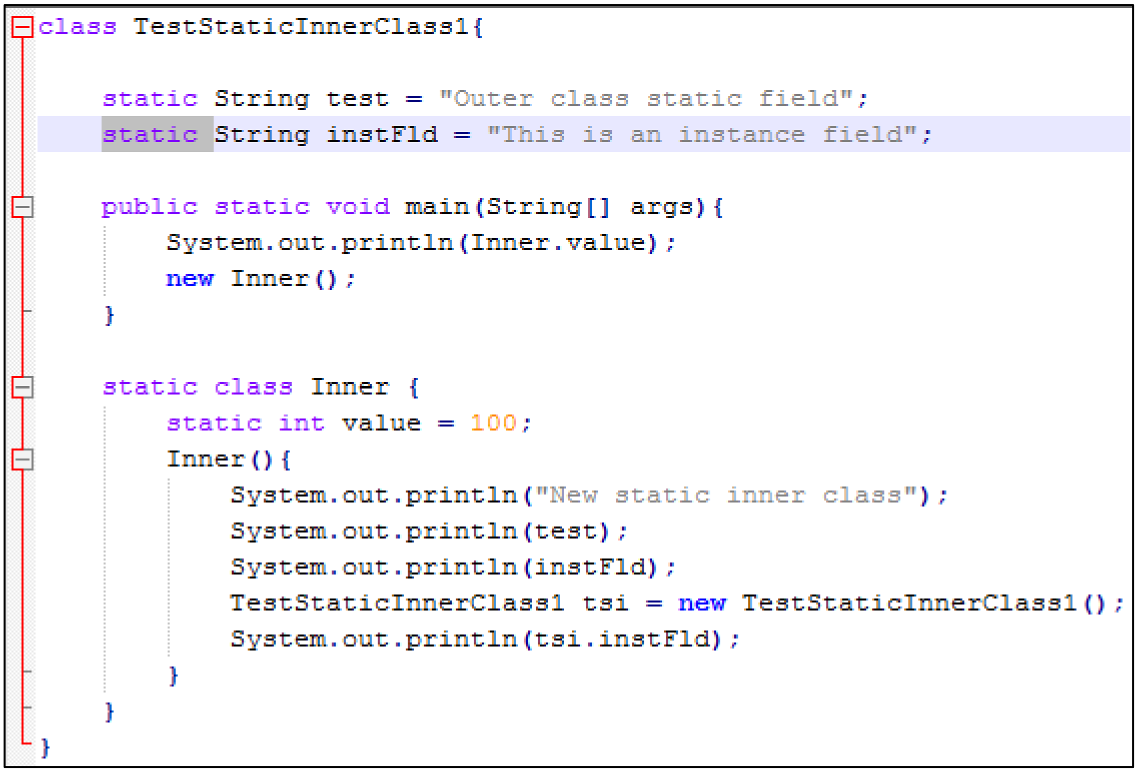


```
class TestStaticInnerClass1{

    static String test = "Outer class static field";
    String instFld = "This is an instance field";

    public static void main(String[] args){
        System.out.println(Inner.value);
        new Inner();
    }

    static class Inner {
        static int value = 100;
        Inner(){
            System.out.println("New static inner class");
            System.out.println(test);
            System.out.println(instFld);
            TestStaticInnerClass1 tsi = new TestStaticInnerClass1();
            System.out.println(tsi.instFld);
        }
    }
}
```

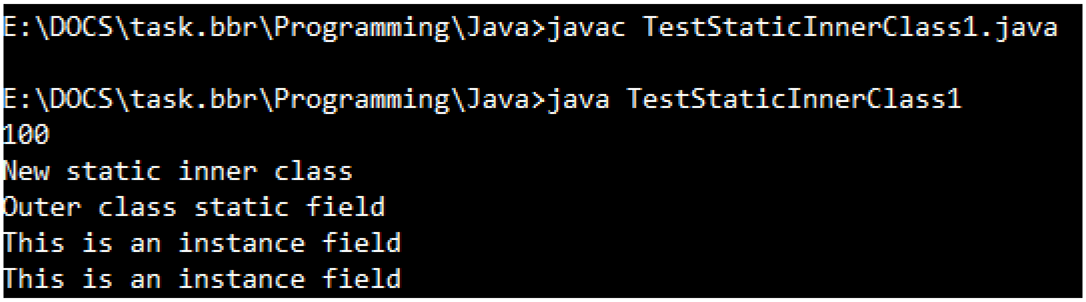


```
E:\DOCS\task.bbr\Programming\Java>javac TestStaticInnerClass1.java
TestStaticInnerClass1.java:16: error: non-static variable instFld cannot be referenced from a static
 context
                        System.out.println(instFld);
                                           ^
1 error
```

Setelah diperbaiki:

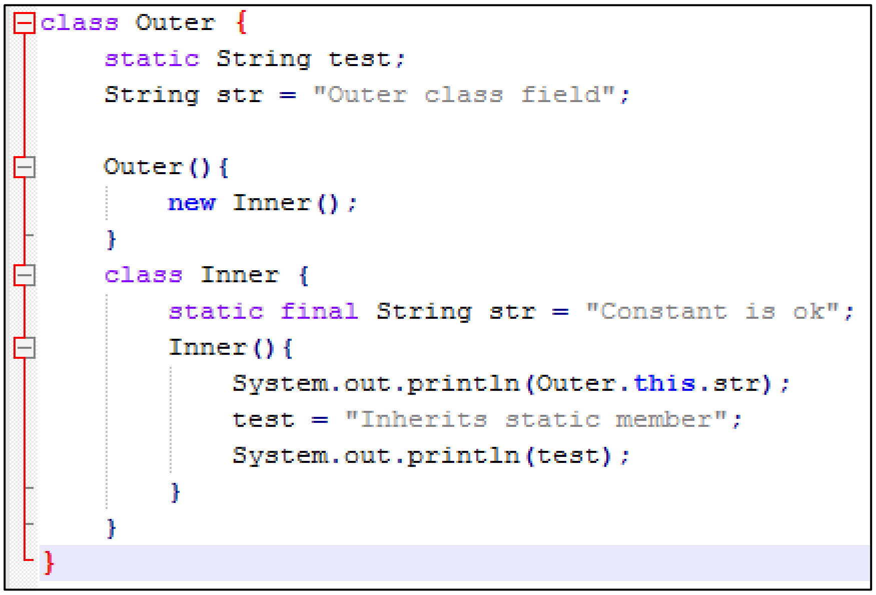


```java
class TestStaticInnerClass1{

    static String test = "Outer class static field";
    static String instFld = "This is an instance field";

    public static void main(String[] args){
        System.out.println(Inner.value);
        new Inner();
    }

    static class Inner {
        static int value = 100;
        Inner(){
            System.out.println("New static inner class");
            System.out.println(test);
            System.out.println(instFld);
            TestStaticInnerClass1 tsi = new TestStaticInnerClass1();
            System.out.println(tsi.instFld);
        }
    }
}
```

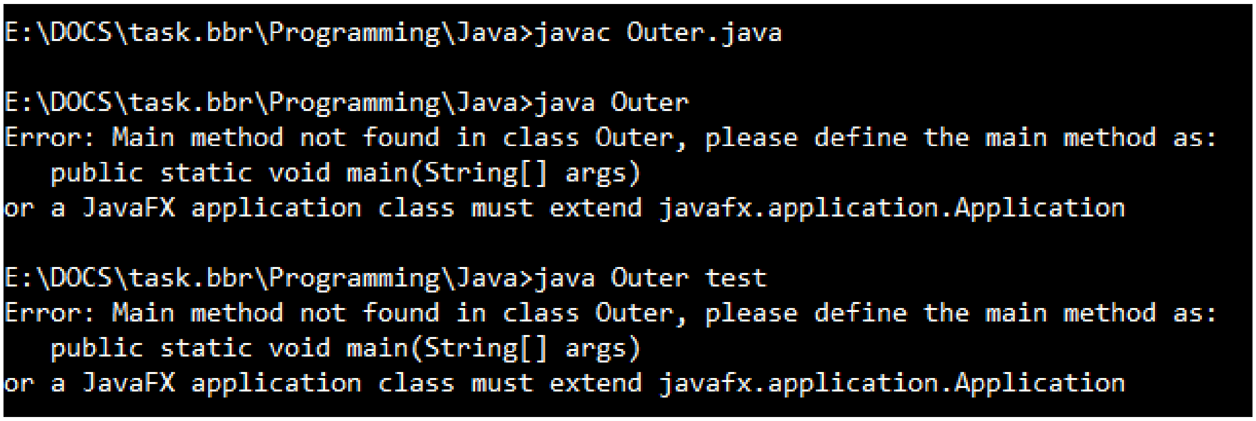


```
E:\DOCS\task.bbr\Programming\Java>javac TestStaticInnerClass1.java

E:\DOCS\task.bbr\Programming\Java>java TestStaticInnerClass1
100
New static inner class
Outer class static field
This is an instance field
This is an instance field
```

Percobaan 4: Non Static Inner class yang dideklarasikan dalam class

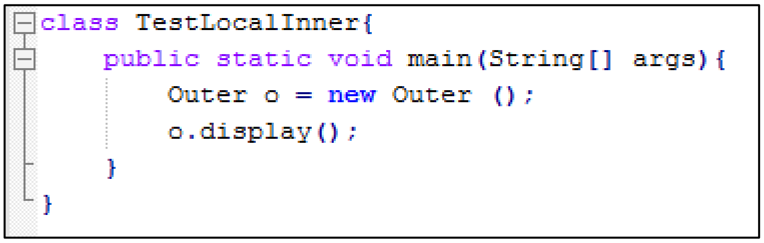


```
class Outer {
    static String test;
    String str = "Outer class field";

    Outer(){
        new Inner();
    }
    class Inner {
        static final String str = "Constant is ok";
        Inner(){
            System.out.println(Outer.this.str);
            test = "Inherits static member";
            System.out.println(test);
        }
    }
}
```

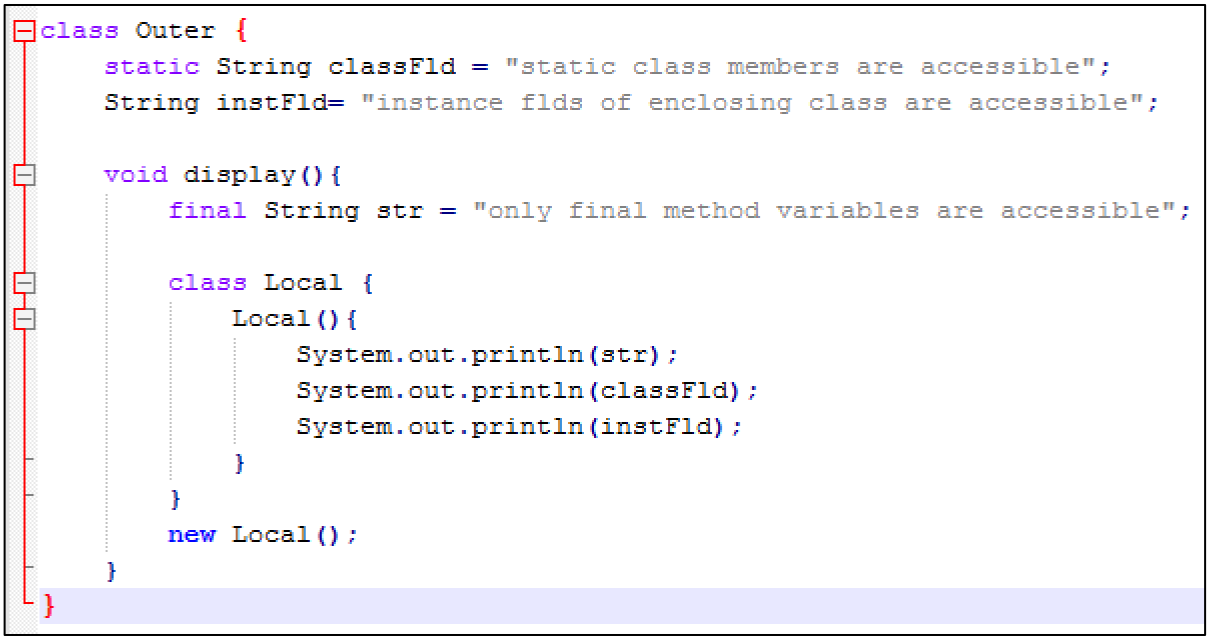


```
E:\DOCS\task.bbr\Programming\Java>javac Outer.java

E:\DOCS\task.bbr\Programming\Java>java Outer
Error: Main method not found in class Outer, please define the main method as:
   public static void main(String[] args)
or a JavaFX application class must extend javafx.application.Application

E:\DOCS\task.bbr\Programming\Java>java Outer test
Error: Main method not found in class Outer, please define the main method as:
   public static void main(String[] args)
or a JavaFX application class must extend javafx.application.Application
```

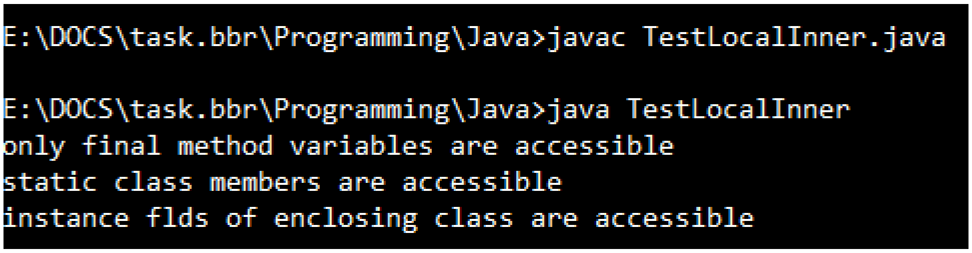
Percobaan 5: Local Inner class



```
class TestLocalInner{
    public static void main(String[] args){
        Outer o = new Outer ();
        o.display();
    }
}
```

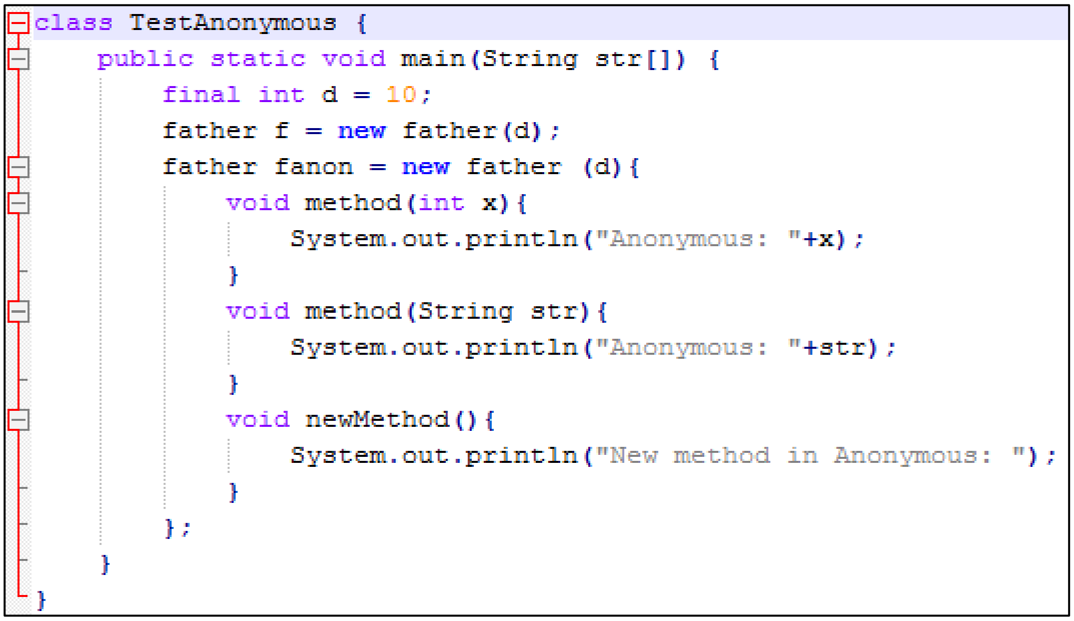


```java
class Outer {
    static String classFld = "static class members are accessible";
    String instFld= "instance flds of enclosing class are accessible";

    void display(){
        final String str = "only final method variables are accessible";

        class Local {
            Local(){
                System.out.println(str);
                System.out.println(classFld);
                System.out.println(instFld);
            }
        }
        new Local();
    }
}
```

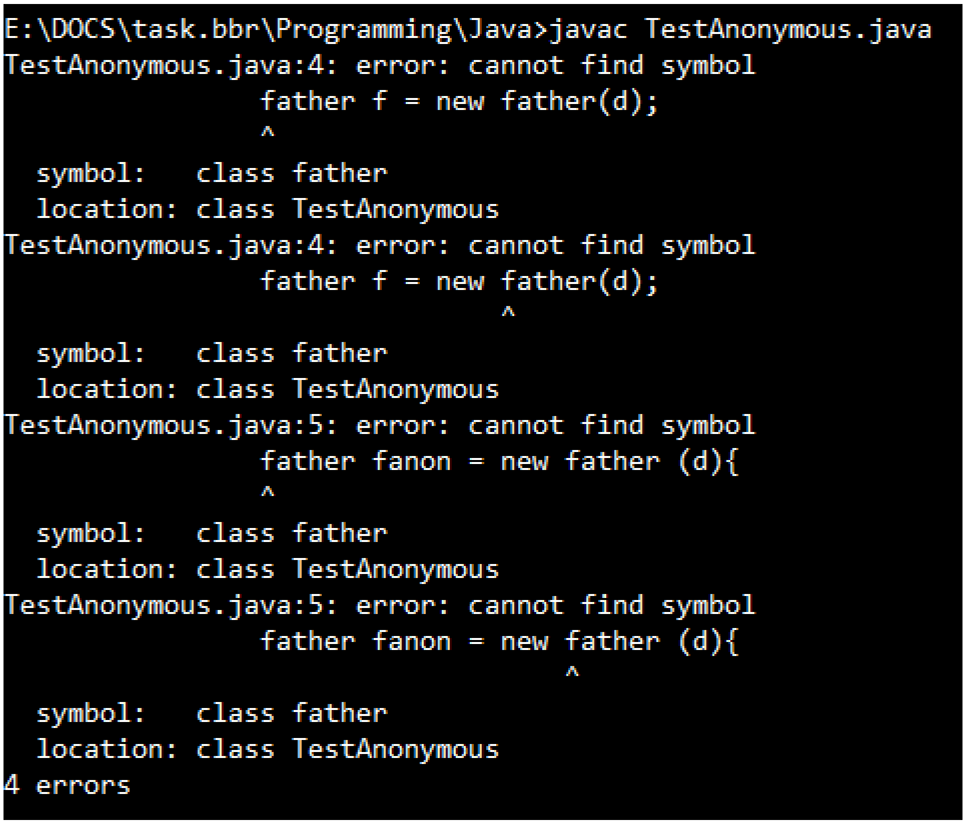


```
E:\DOCS\task.bbr\Programming\Java>javac TestLocalInner.java

E:\DOCS\task.bbr\Programming\Java>java TestLocalInner
only final method variables are accessible
static class members are accessible
instance flds of enclosing class are accessible
```

Percobaan 6: Anonymous class

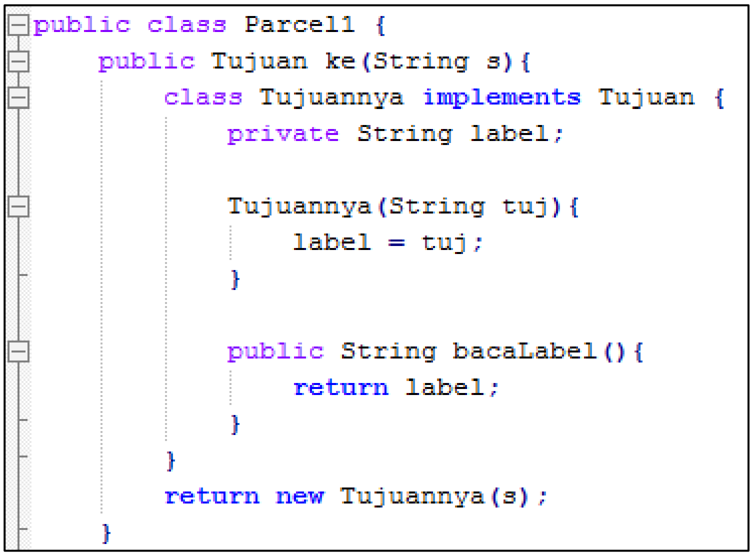


```java
class TestAnonymous {
    public static void main(String str[]) {
        final int d = 10;
        father f = new father(d);
        father fanon = new father (d){
            void method(int x){
                System.out.println("Anonymous: "+x);
            }
            void method(String str){
                System.out.println("Anonymous: "+str);
            }
            void newMethod(){
                System.out.println("New method in Anonymous: ");
            }
        };
    }
}
```

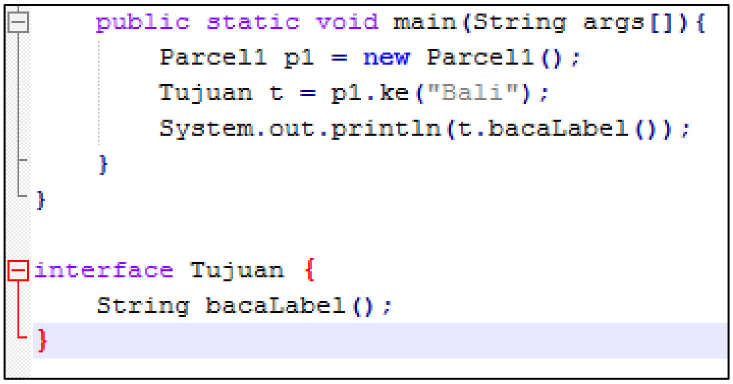


```
E:\DOCS\task.bbr\Programming\Java>javac TestAnonymous.java
TestAnonymous.java:4: error: cannot find symbol
                father f = new father(d);
                ^
  symbol:   class father
  location: class TestAnonymous
TestAnonymous.java:4: error: cannot find symbol
                father f = new father(d);
                               ^
  symbol:   class father
  location: class TestAnonymous
TestAnonymous.java:5: error: cannot find symbol
                father fanon = new father (d){
                ^
  symbol:   class father
  location: class TestAnonymous
TestAnonymous.java:5: error: cannot find symbol
                father fanon = new father (d){
                                   ^
  symbol:   class father
  location: class TestAnonymous
4 errors
```

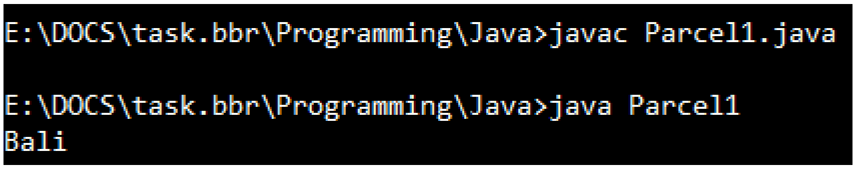
Percobaan 7: Inner class yang dideklarasikan Inner class yang dideklarasikan di dalam method



```
public class Parcel1 {
    public Tujuan ke(String s){
        class Tujuannya implements Tujuan {
            private String label;

            Tujuannya(String tuj){
                label = tuj;
            }

            public String bacaLabel(){
                return label;
            }
        }
        return new Tujuannya(s);
    }
```

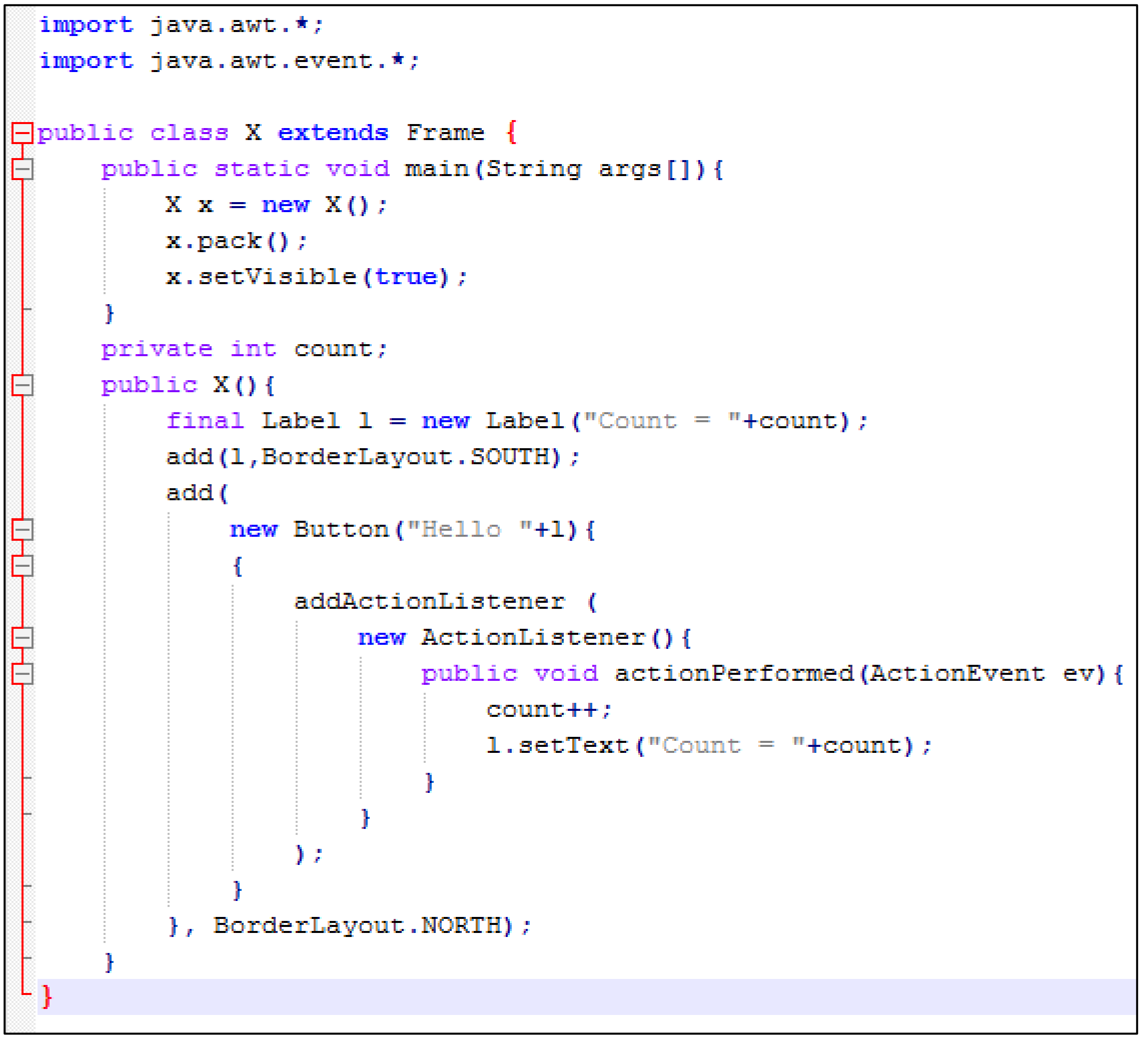


```
    public static void main(String args[]){
        Parcel1 p1 = new Parcel1();
        Tujuan t = p1.ke("Bali");
        System.out.println(t.bacaLabel());
    }
}

interface Tujuan {
    String bacaLabel();
}
```

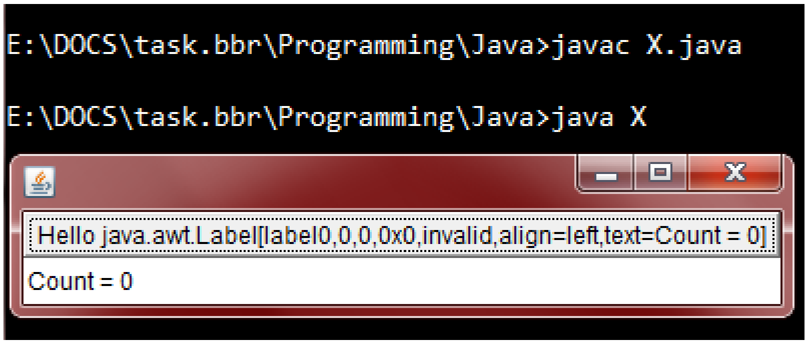


```
E:\DOCS\task.bbr\Programming\Java>javac Parcel1.java

E:\DOCS\task.bbr\Programming\Java>java Parcel1
Bali
```

Percobaan 8: Anonymous Inner Class



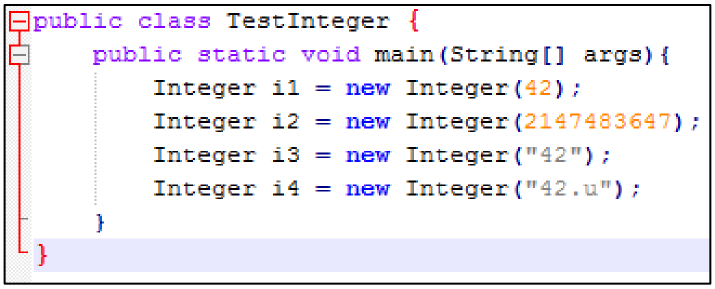
```
import java.awt.*;
import java.awt.event.*;

public class X extends Frame {
    public static void main(String args[]){
        X x = new X();
        x.pack();
        x.setVisible(true);
    }
    private int count;
    public X(){
        final Label l = new Label("Count = "+count);
        add(l,BorderLayout.SOUTH);
        add(
            new Button("Hello "+l){
            {
                addActionListener (
                    new ActionListener(){
                        public void actionPerformed(ActionEvent ev){
                            count++;
                            l.setText("Count = "+count);
                        }
                    }
                );
            }
        }, BorderLayout.NORTH);
    }
}
```



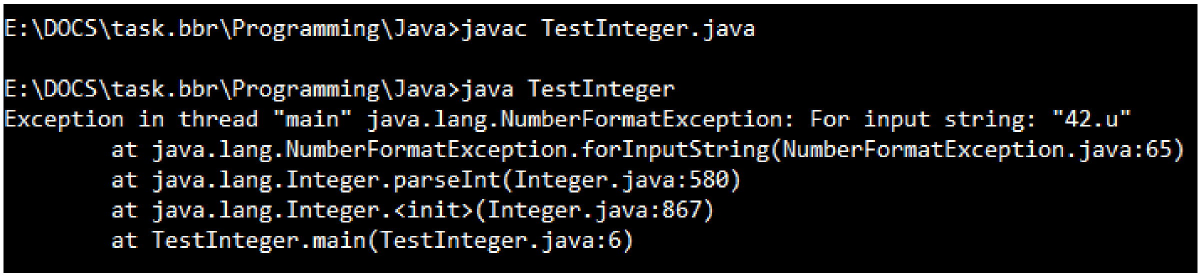
```
E:\DOCS\task.bbr\Programming\Java>javac X.java

E:\DOCS\task.bbr\Programming\Java>java X
```

```
Hello java.awt.Label[label0,0,0,0x0,invalid,align=left,text=Count = 0]
Count = 0
```

Percobaan 9: Memahami cara membuat objek Integer. Jika terjadi error atau exception, jelaskan penyebabnya!

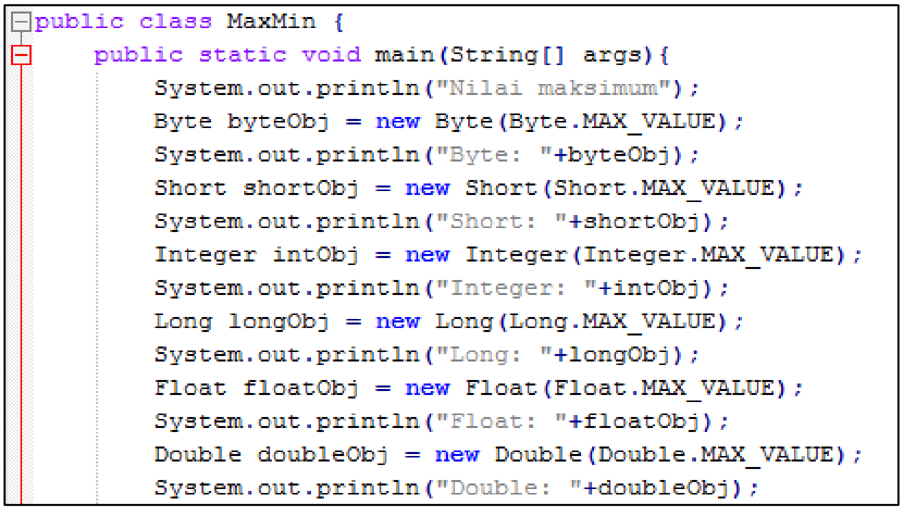


```java
public class TestInteger {
    public static void main(String[] args){
        Integer i1 = new Integer(42);
        Integer i2 = new Integer(2147483647);
        Integer i3 = new Integer("42");
        Integer i4 = new Integer("42.u");
    }
}
```

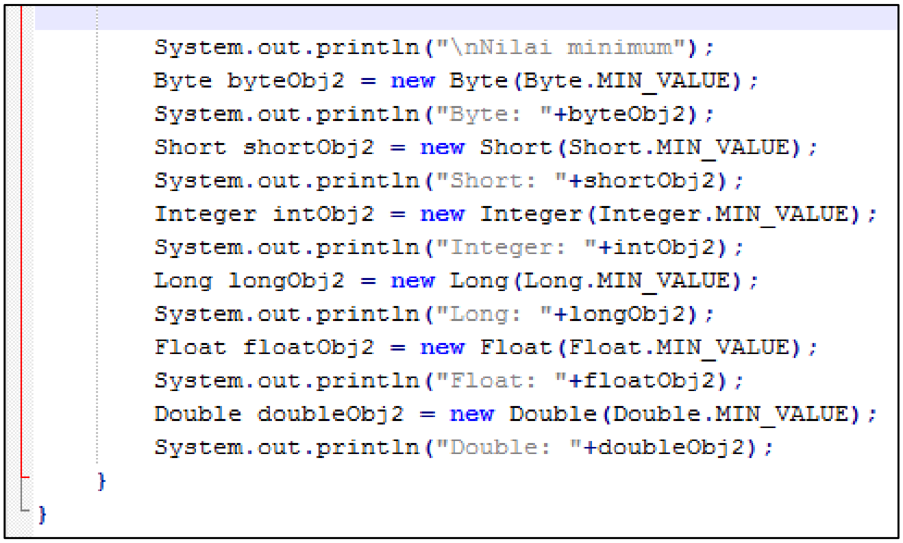


```
E:\DOCS\task.bbr\Programming\Java>javac TestInteger.java

E:\DOCS\task.bbr\Programming\Java>java TestInteger
Exception in thread "main" java.lang.NumberFormatException: For input string: "42.u"
        at java.lang.NumberFormatException.forInputString(NumberFormatException.java:65)
        at java.lang.Integer.parseInt(Integer.java:580)
        at java.lang.Integer.<init>(Integer.java:867)
        at TestInteger.main(TestInteger.java:6)
```

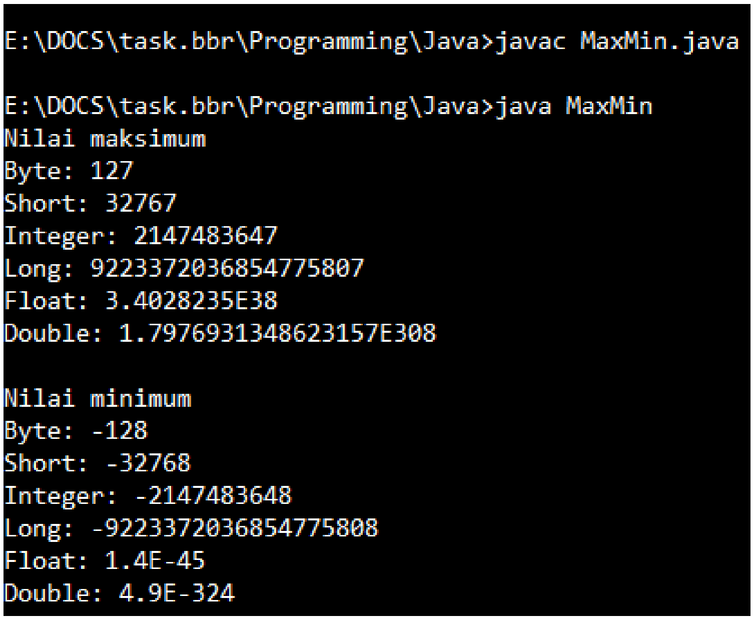
Percobaan 10: Mengetahui nilai maksimum dan minimum untuk tipe Wrapper



```java
public class MaxMin {
    public static void main(String[] args){
        System.out.println("Nilai maksimum");
        Byte byteObj = new Byte(Byte.MAX_VALUE);
        System.out.println("Byte: "+byteObj);
        Short shortObj = new Short(Short.MAX_VALUE);
        System.out.println("Short: "+shortObj);
        Integer intObj = new Integer(Integer.MAX_VALUE);
        System.out.println("Integer: "+intObj);
        Long longObj = new Long(Long.MAX_VALUE);
        System.out.println("Long: "+longObj);
        Float floatObj = new Float(Float.MAX_VALUE);
        System.out.println("Float: "+floatObj);
        Double doubleObj = new Double(Double.MAX_VALUE);
        System.out.println("Double: "+doubleObj);
```

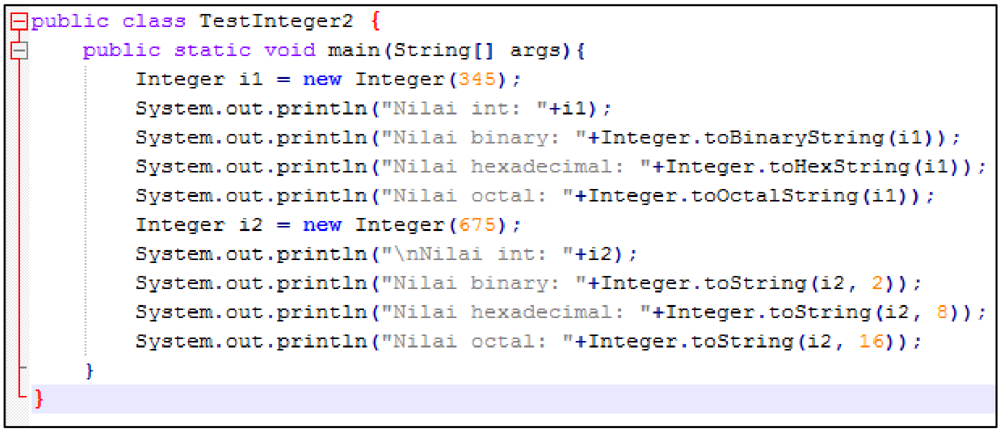


```java
        System.out.println("\nNilai minimum");
        Byte byteObj2 = new Byte(Byte.MIN_VALUE);
        System.out.println("Byte: "+byteObj2);
        Short shortObj2 = new Short(Short.MIN_VALUE);
        System.out.println("Short: "+shortObj2);
        Integer intObj2 = new Integer(Integer.MIN_VALUE);
        System.out.println("Integer: "+intObj2);
        Long longObj2 = new Long(Long.MIN_VALUE);
        System.out.println("Long: "+longObj2);
        Float floatObj2 = new Float(Float.MIN_VALUE);
        System.out.println("Float: "+floatObj2);
        Double doubleObj2 = new Double(Double.MIN_VALUE);
        System.out.println("Double: "+doubleObj2);
    }
}
```

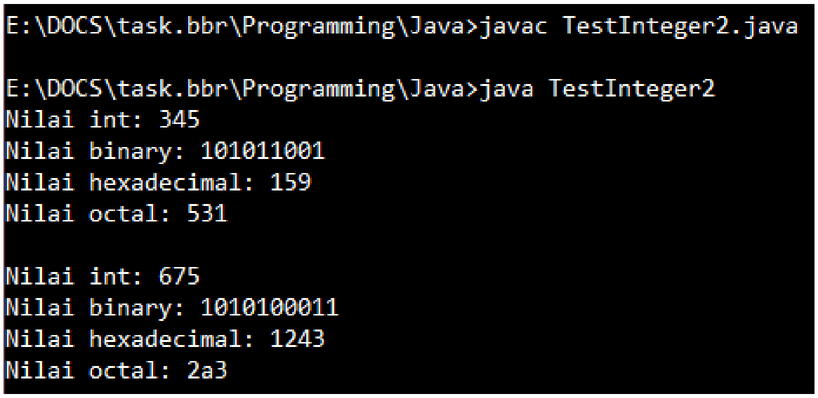


```
E:\DOCS\task.bbr\Programming\Java>javac MaxMin.java

E:\DOCS\task.bbr\Programming\Java>java MaxMin
Nilai maksimum
Byte: 127
Short: 32767
Integer: 2147483647
Long: 9223372036854775807
Float: 3.4028235E38
Double: 1.7976931348623157E308

Nilai minimum
Byte: -128
Short: -32768
Integer: -2147483648
Long: -9223372036854775808
Float: 1.4E-45
Double: 4.9E-324
```

Percobaan 11: Menampilkan bilangan integer menjadi biner, octal dan heksadesimal

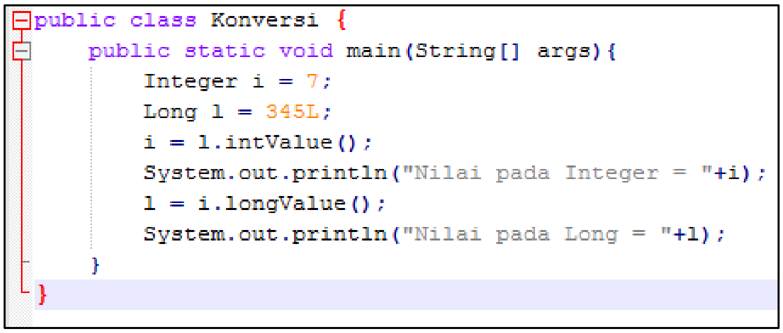


```java
public class TestInteger2 {
    public static void main(String[] args){
        Integer i1 = new Integer(345);
        System.out.println("Nilai int: "+i1);
        System.out.println("Nilai binary: "+Integer.toBinaryString(i1));
        System.out.println("Nilai hexadecimal: "+Integer.toHexString(i1));
        System.out.println("Nilai octal: "+Integer.toOctalString(i1));
        Integer i2 = new Integer(675);
        System.out.println("\nNilai int: "+i2);
        System.out.println("Nilai binary: "+Integer.toString(i2, 2));
        System.out.println("Nilai hexadecimal: "+Integer.toString(i2, 8));
        System.out.println("Nilai octal: "+Integer.toString(i2, 16));
    }
}
```

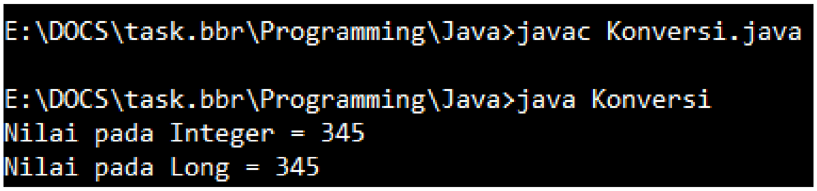


```
E:\DOCS\task.bbr\Programming\Java>javac TestInteger2.java

E:\DOCS\task.bbr\Programming\Java>java TestInteger2
Nilai int: 345
Nilai binary: 101011001
Nilai hexadecimal: 159
Nilai octal: 531

Nilai int: 675
Nilai binary: 1010100011
Nilai hexadecimal: 1243
Nilai octal: 2a3
```

Percobaan 12: Memahami cara konversi antar tipe Wrapper, misal dari objek Integer diassignkan ke objek Long dan sebaliknya

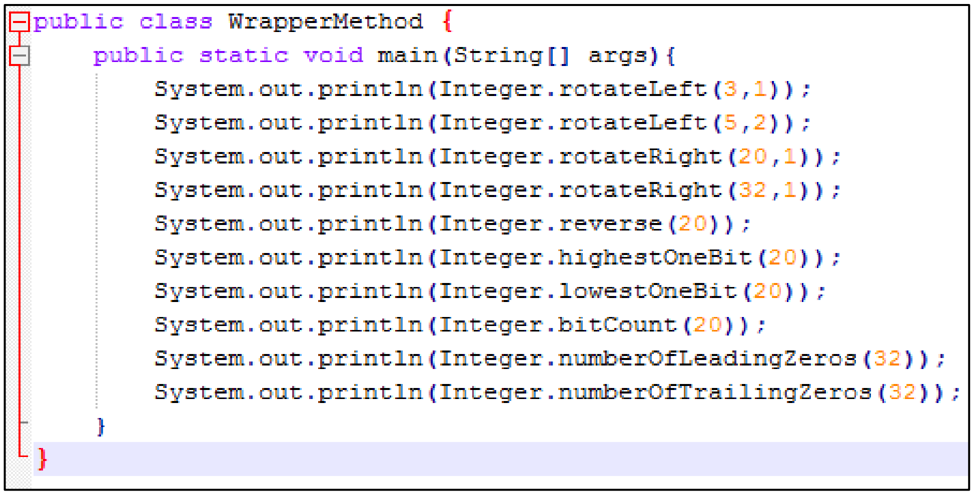


```
public class Konversi {
    public static void main(String[] args){
        Integer i = 7;
        Long l = 345L;
        i = l.intValue();
        System.out.println("Nilai pada Integer = "+i);
        l = i.longValue();
        System.out.println("Nilai pada Long = "+l);
    }
}
```

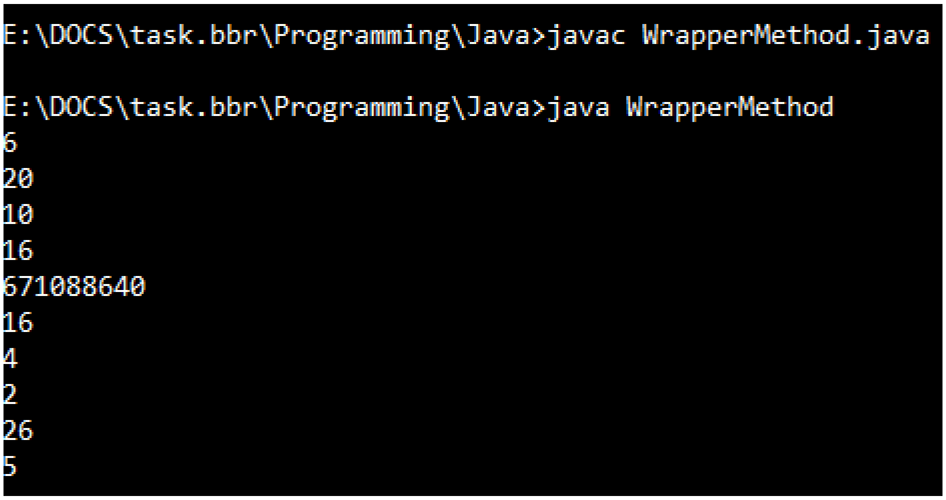


```
E:\DOCS\task.bbr\Programming\Java>javac Konversi.java

E:\DOCS\task.bbr\Programming\Java>java Konversi
Nilai pada Integer = 345
Nilai pada Long = 345
```

Percobaan 13: Jelaskan kegunaan dari method-method di bawah ini!

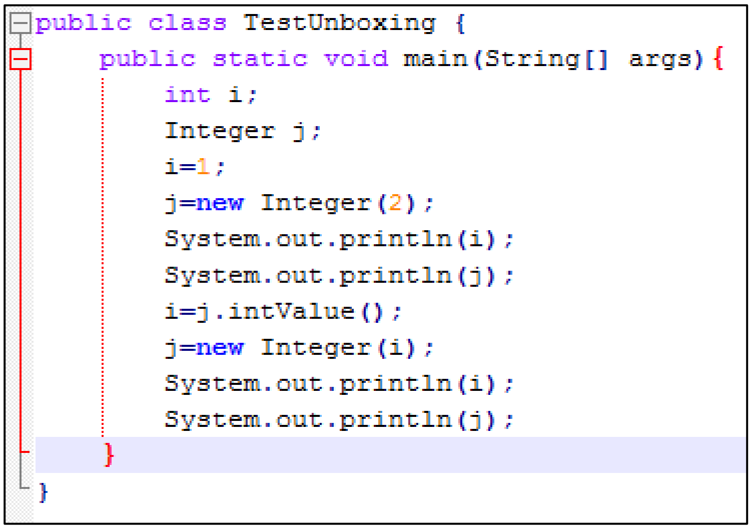


```
public class WrapperMethod {
    public static void main(String[] args){
        System.out.println(Integer.rotateLeft(3,1));
        System.out.println(Integer.rotateLeft(5,2));
        System.out.println(Integer.rotateRight(20,1));
        System.out.println(Integer.rotateRight(32,1));
        System.out.println(Integer.reverse(20));
        System.out.println(Integer.highestOneBit(20));
        System.out.println(Integer.lowestOneBit(20));
        System.out.println(Integer.bitCount(20));
        System.out.println(Integer.numberOfLeadingZeros(32));
        System.out.println(Integer.numberOfTrailingZeros(32));
    }
}
```

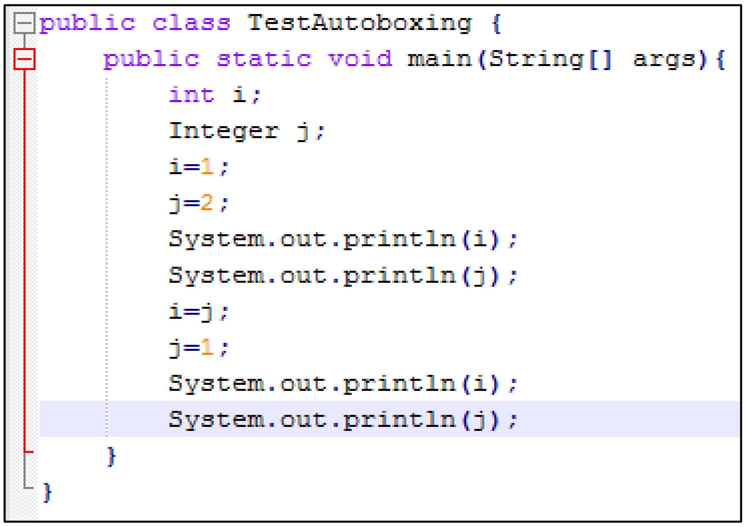


```
E:\DOCS\task.bbr\Programming\Java>javac WrapperMethod.java

E:\DOCS\task.bbr\Programming\Java>java WrapperMethod
6
20
10
16
671088640
16
4
2
26
5
```

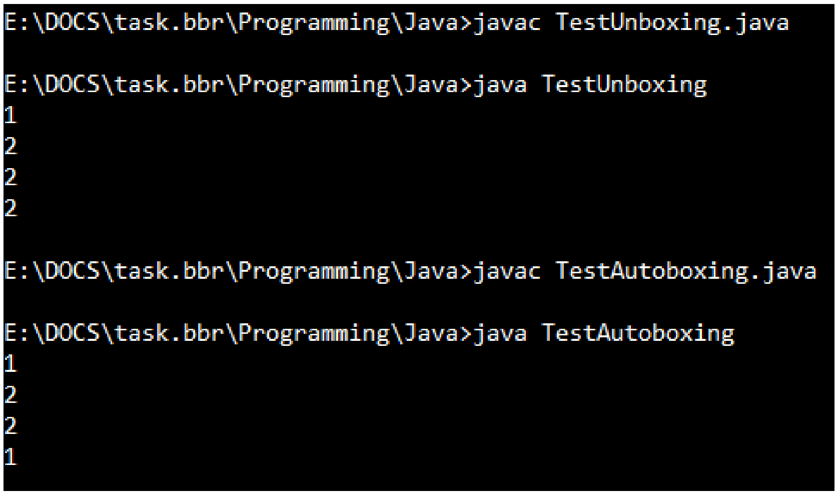
Percobaan 14: Memahami mengenai Autoboxing dan Auto-unboxing



```
public class TestUnboxing {
    public static void main(String[] args){
        int i;
        Integer j;
        i=1;
        j=new Integer(2);
        System.out.println(i);
        System.out.println(j);
        i=j.intValue();
        j=new Integer(i);
        System.out.println(i);
        System.out.println(j);
    }
}
```

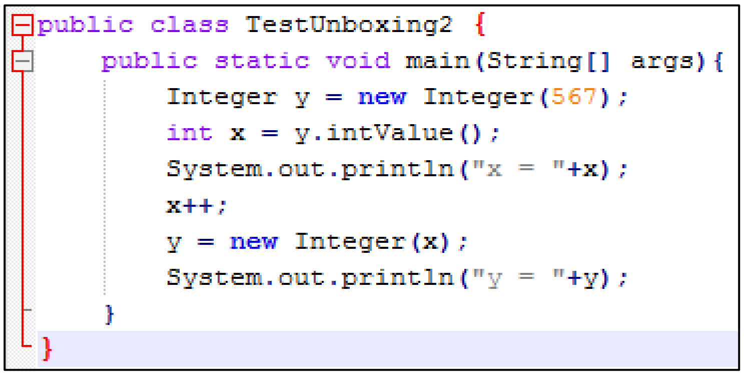


```
public class TestAutoboxing {
    public static void main(String[] args){
        int i;
        Integer j;
        i=1;
        j=2;
        System.out.println(i);
        System.out.println(j);
        i=j;
        j=1;
        System.out.println(i);
        System.out.println(j);
    }
}
```

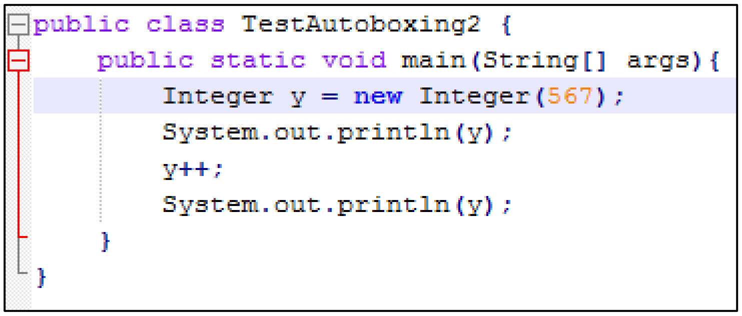


```
E:\DOCS\task.bbr\Programming\Java>javac TestUnboxing.java

E:\DOCS\task.bbr\Programming\Java>java TestUnboxing
1
2
2
2

E:\DOCS\task.bbr\Programming\Java>javac TestAutoboxing.java

E:\DOCS\task.bbr\Programming\Java>java TestAutoboxing
1
2
2
1
```

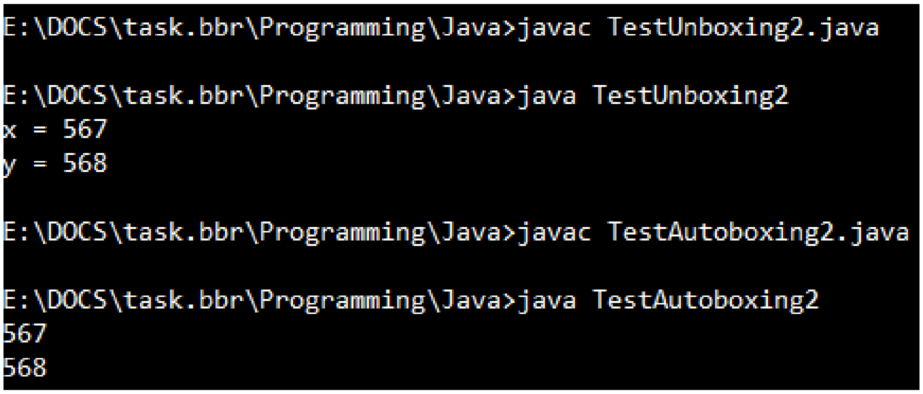
Percobaan 15: Memahami mengenai Autoboxing dan Auto-unboxing (2)



```
public class TestUnboxing2 {
    public static void main(String[] args){
        Integer y = new Integer(567);
        int x = y.intValue();
        System.out.println("x = "+x);
        x++;
        y = new Integer(x);
        System.out.println("y = "+y);
    }
}
```

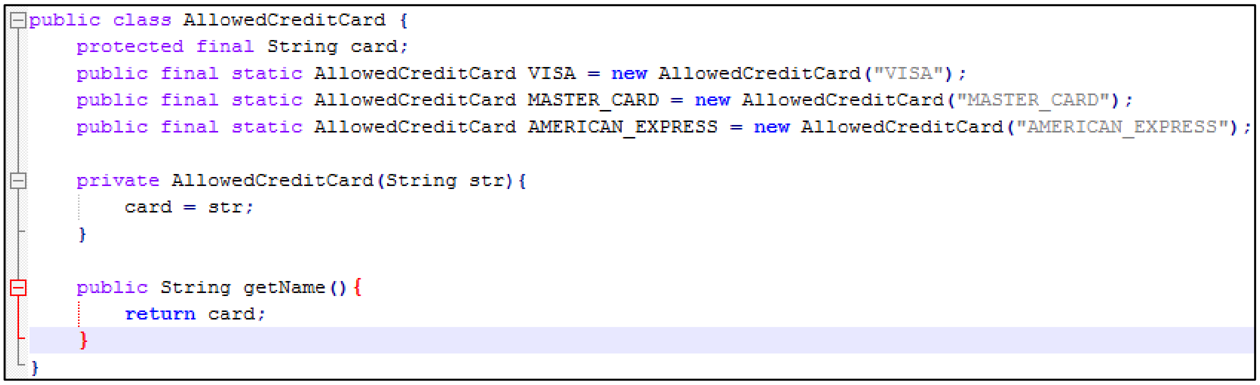


```
public class TestAutoboxing2 {
    public static void main(String[] args){
        Integer y = new Integer(567);
        System.out.println(y);
        y++;
        System.out.println(y);
    }
}
```

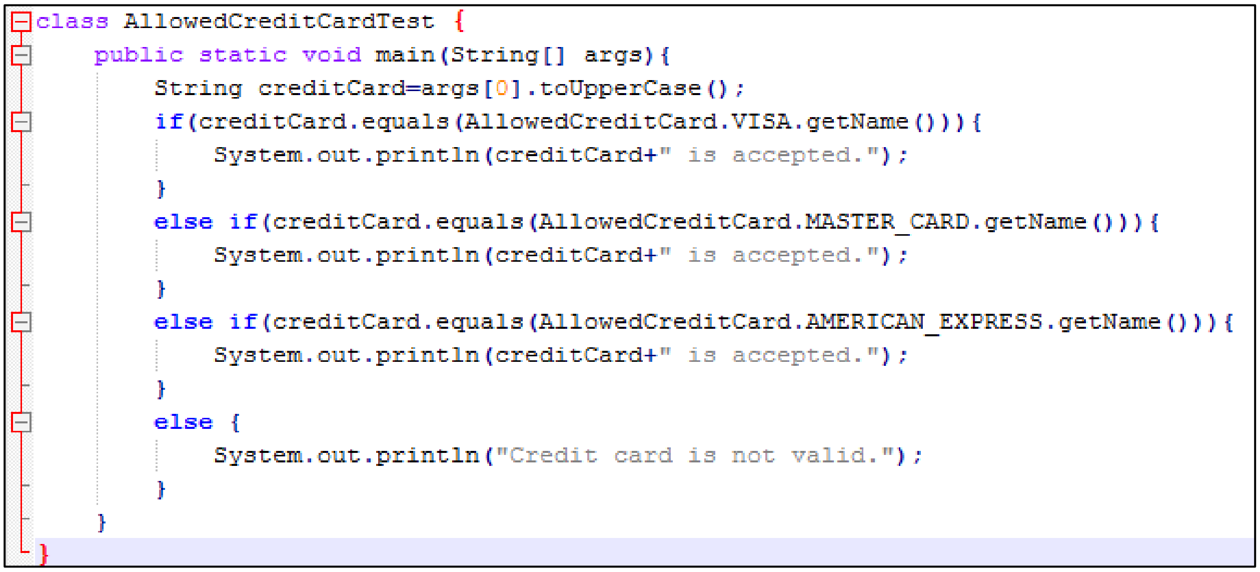


```
E:\DOCS\task.bbr\Programming\Java>javac TestUnboxing2.java

E:\DOCS\task.bbr\Programming\Java>java TestUnboxing2
x = 567
y = 568

E:\DOCS\task.bbr\Programming\Java>javac TestAutoboxing2.java

E:\DOCS\task.bbr\Programming\Java>java TestAutoboxing2
567
568
```

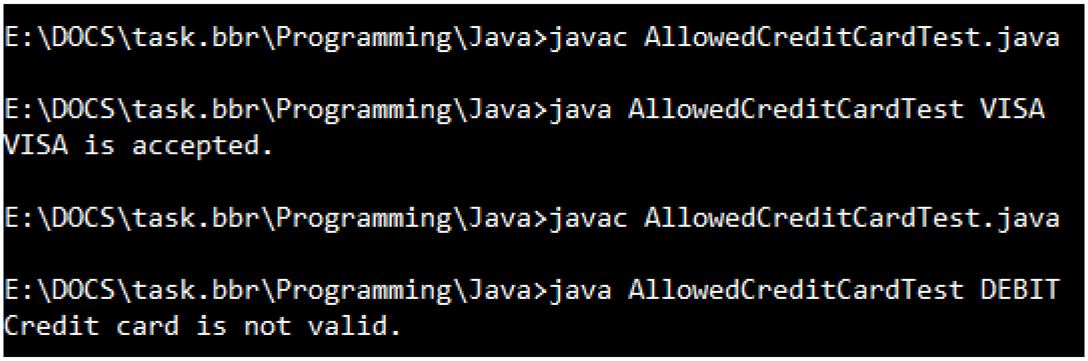
Percobaan 16: Class AllowedCreditCard diubah menjadi bentuk enum



```
public class AllowedCreditCard {
    protected final String card;
    public final static AllowedCreditCard VISA = new AllowedCreditCard("VISA");
    public final static AllowedCreditCard MASTER_CARD = new AllowedCreditCard("MASTER_CARD");
    public final static AllowedCreditCard AMERICAN_EXPRESS = new AllowedCreditCard("AMERICAN_EXPRESS");

    private AllowedCreditCard(String str){
        card = str;
    }

    public String getName(){
        return card;
    }
}
```

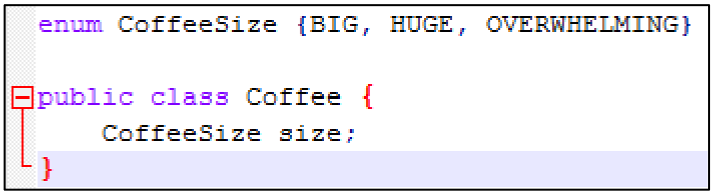


```
class AllowedCreditCardTest {
    public static void main(String[] args){
        String creditCard=args[0].toUpperCase();
        if(creditCard.equals(AllowedCreditCard.VISA.getName())){
            System.out.println(creditCard+" is accepted.");
        }
        else if(creditCard.equals(AllowedCreditCard.MASTER_CARD.getName())){
            System.out.println(creditCard+" is accepted.");
        }
        else if(creditCard.equals(AllowedCreditCard.AMERICAN_EXPRESS.getName())){
            System.out.println(creditCard+" is accepted.");
        }
        else {
            System.out.println("Credit card is not valid.");
        }
    }
}
```

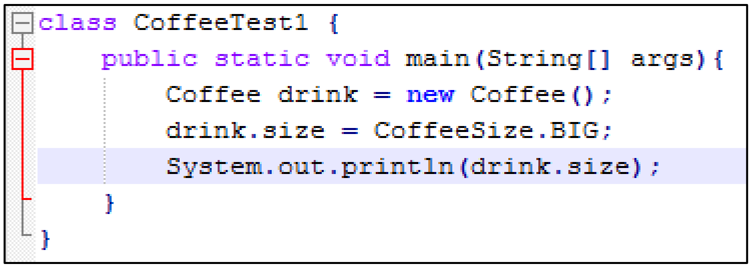


```
E:\DOCS\task.bbr\Programming\Java>javac AllowedCreditCardTest.java

E:\DOCS\task.bbr\Programming\Java>java AllowedCreditCardTest VISA
VISA is accepted.

E:\DOCS\task.bbr\Programming\Java>javac AllowedCreditCardTest.java

E:\DOCS\task.bbr\Programming\Java>java AllowedCreditCardTest DEBIT
Credit card is not valid.
```

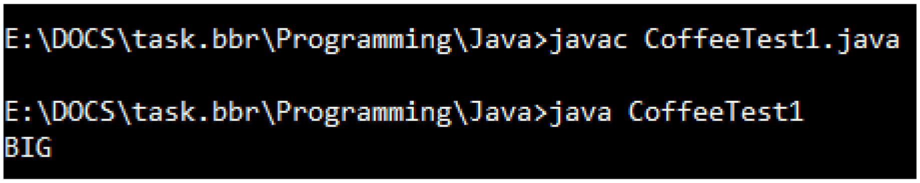
Percobaan 17: Mendeklarasikan enum di luar class



```
enum CoffeeSize {BIG, HUGE, OVERWHELMING}

public class Coffee {
    CoffeeSize size;
}
```

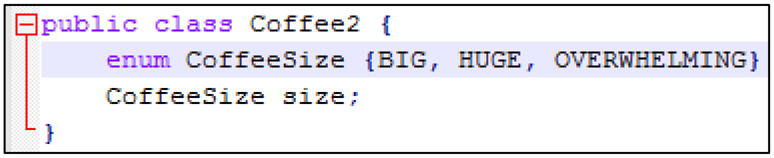


```
class CoffeeTest1 {
    public static void main(String[] args){
        Coffee drink = new Coffee();
        drink.size = CoffeeSize.BIG;
        System.out.println(drink.size);
    }
}
```

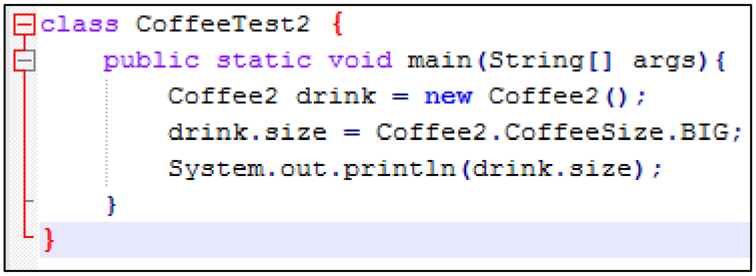


```
E:\DOCS\task.bbr\Programming\Java>javac CoffeeTest1.java

E:\DOCS\task.bbr\Programming\Java>java CoffeeTest1
BIG
```

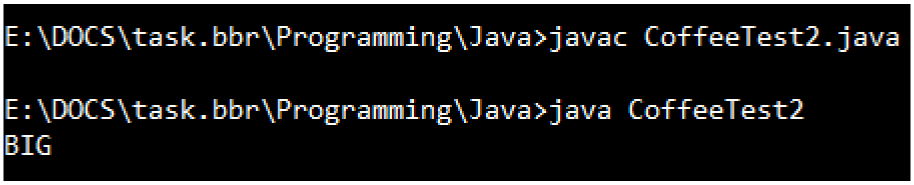
Percobaan 18: Mendeklarasikan enum di dalam class



```
public class Coffee2 {
    enum CoffeeSize {BIG, HUGE, OVERWHELMING}
    CoffeeSize size;
}
```

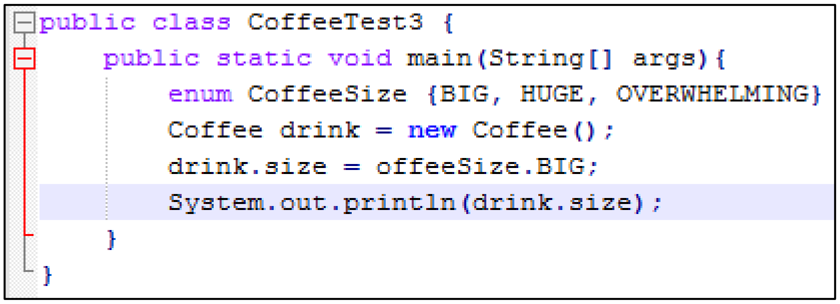


```
class CoffeeTest2 {
    public static void main(String[] args){
        Coffee2 drink = new Coffee2();
        drink.size = Coffee2.CoffeeSize.BIG;
        System.out.println(drink.size);
    }
}
```

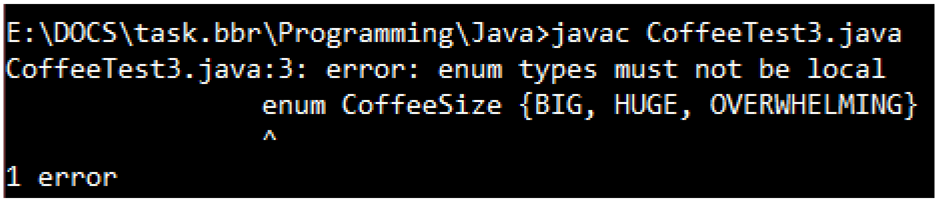


```
E:\DOCS\task.bbr\Programming\Java>javac CoffeeTest2.java

E:\DOCS\task.bbr\Programming\Java>java CoffeeTest2
BIG
```

Percobaan 19: Tidak bisa mendeklarasikan enum di dalam method

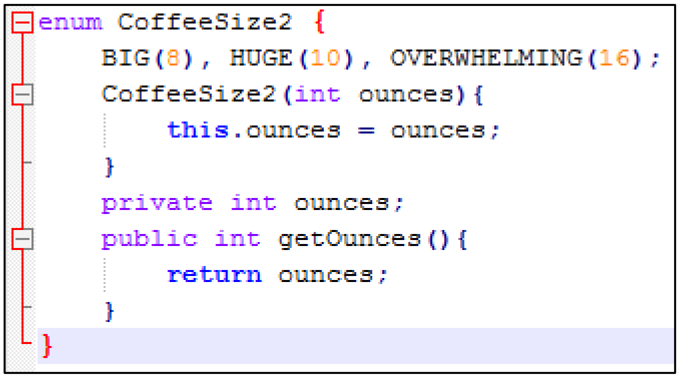


```
public class CoffeeTest3 {
    public static void main(String[] args){
        enum CoffeeSize {BIG, HUGE, OVERWHELMING}
        Coffee drink = new Coffee();
        drink.size = offeeSize.BIG;
        System.out.println(drink.size);
    }
}
```

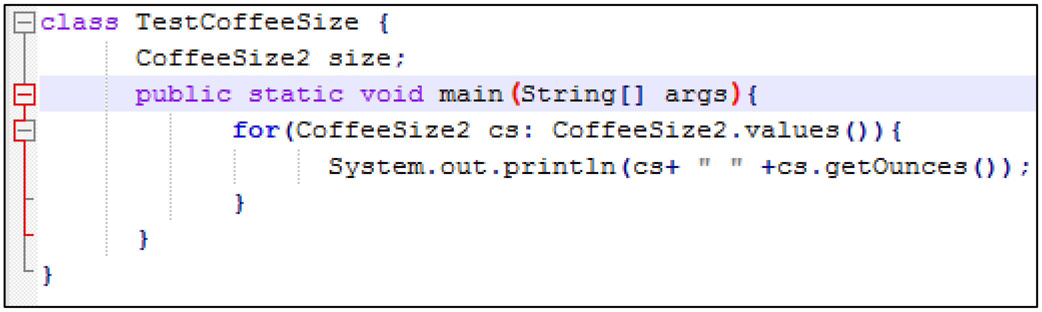


```
E:\DOCS\task.bbr\Programming\Java>javac CoffeeTest3.java
CoffeeTest3.java:3: error: enum types must not be local
                enum CoffeeSize {BIG, HUGE, OVERWHELMING}
                ^
1 error
```

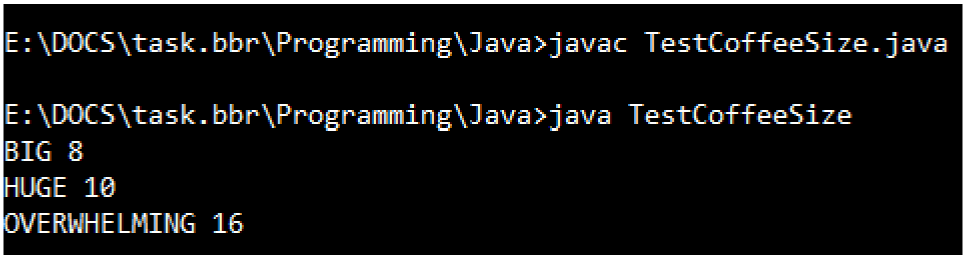
Percobaan 20: Menentukan sendiri nilai konstanta dari enum



```
enum CoffeeSize2 {
    BIG(8), HUGE(10), OVERWHELMING(16);
    CoffeeSize2(int ounces){
        this.ounces = ounces;
    }
    private int ounces;
    public int getOunces(){
        return ounces;
    }
}
```

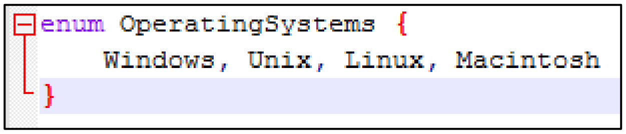


```
class TestCoffeeSize {
      CoffeeSize2 size;
      public static void main(String[] args){
            for(CoffeeSize2 cs: CoffeeSize2.values()){
                  System.out.println(cs+ " " +cs.getOunces());
            }
      }
}
```

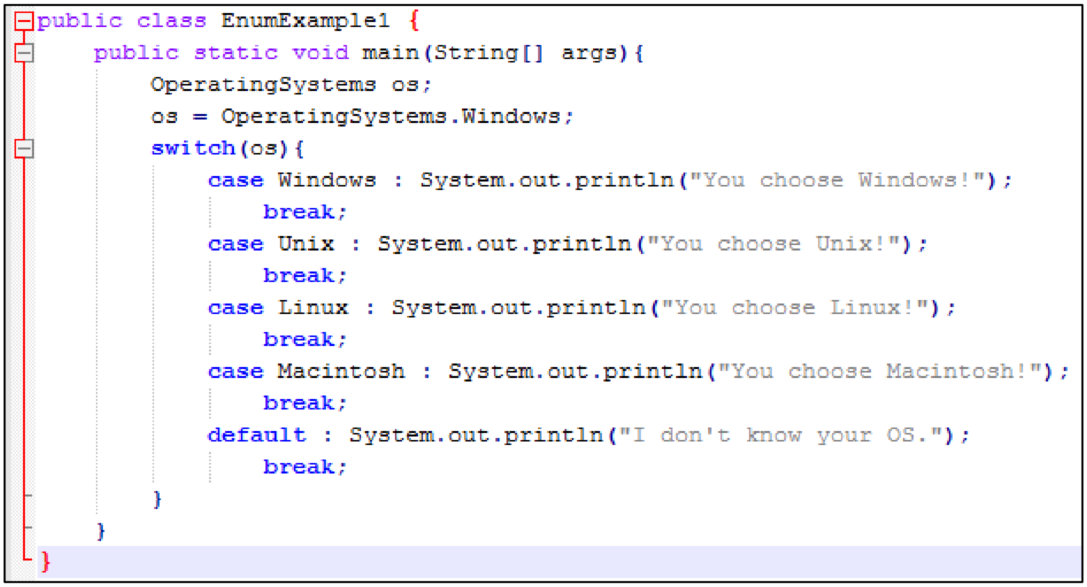


```
E:\DOCS\task.bbr\Programming\Java>javac TestCoffeeSize.java

E:\DOCS\task.bbr\Programming\Java>java TestCoffeeSize
BIG 8
HUGE 10
OVERWHELMING 16
```

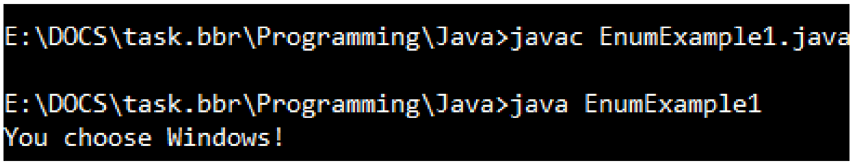
Percobaan 21: Enum dengan statement switch



```
enum OperatingSystems {
    Windows, Unix, Linux, Macintosh
}
```

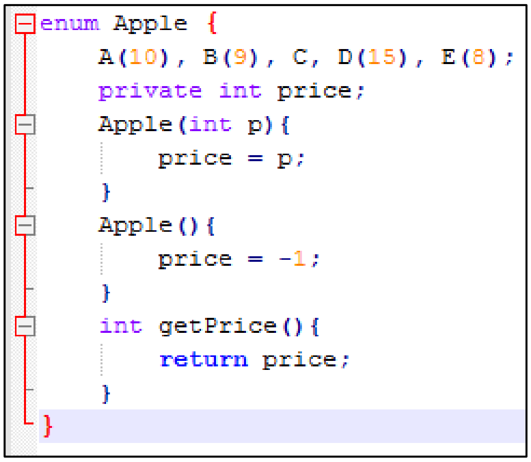


```
public class EnumExample1 {
    public static void main(String[] args){
        OperatingSystems os;
        os = OperatingSystems.Windows;
        switch(os){
            case Windows : System.out.println("You choose Windows!");
                break;
            case Unix : System.out.println("You choose Unix!");
                break;
            case Linux : System.out.println("You choose Linux!");
                break;
            case Macintosh : System.out.println("You choose Macintosh!");
                break;
            default : System.out.println("I don't know your OS.");
                break;
        }
    }
}
```

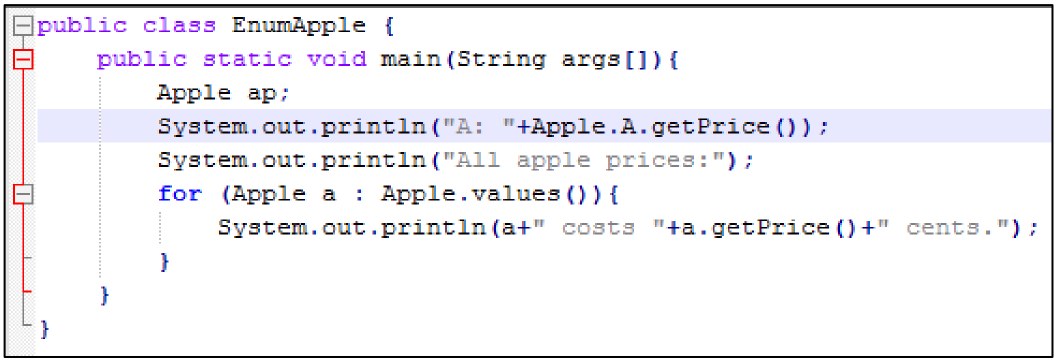


```
E:\DOCS\task.bbr\Programming\Java>javac EnumExample1.java

E:\DOCS\task.bbr\Programming\Java>java EnumExample1
You choose Windows!
```

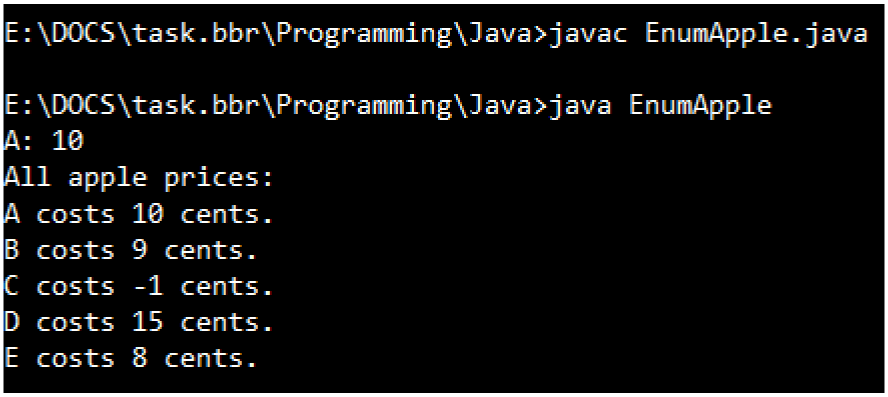
Percobaan 22: Fungsi pada Enum



```
enum Apple {
    A(10), B(9), C, D(15), E(8);
    private int price;
    Apple(int p){
        price = p;
    }
    Apple(){
        price = -1;
    }
    int getPrice(){
        return price;
    }
}
```

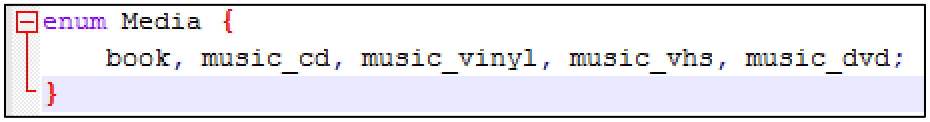


```
public class EnumApple {
    public static void main(String args[]){
        Apple ap;
        System.out.println("A: "+Apple.A.getPrice());
        System.out.println("All apple prices:");
        for (Apple a : Apple.values()){
            System.out.println(a+" costs "+a.getPrice()+" cents.");
        }
    }
}
```

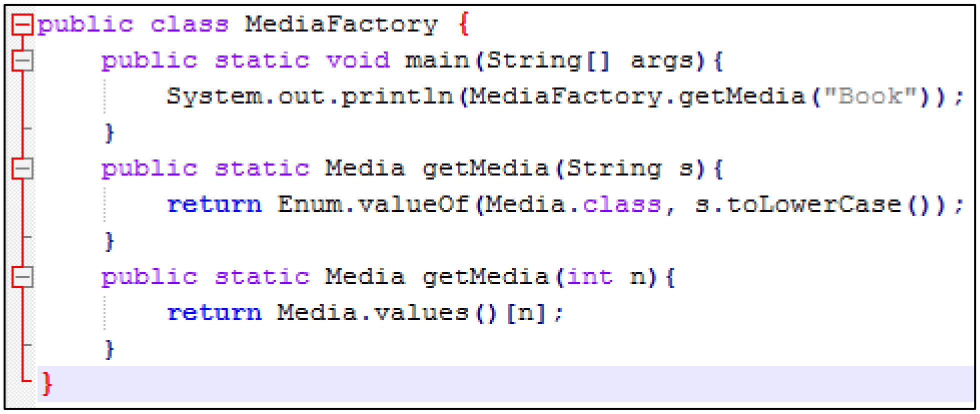


```
E:\DOCS\task.bbr\Programming\Java>javac EnumApple.java

E:\DOCS\task.bbr\Programming\Java>java EnumApple
A: 10
All apple prices:
A costs 10 cents.
B costs 9 cents.
C costs -1 cents.
D costs 15 cents.
E costs 8 cents.
```

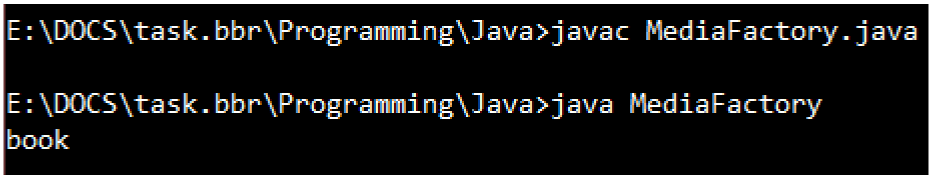
Percobaan 23: Melakukan enumerasi pada Enum



```
enum Media {
    book, music_cd, music_vinyl, music_vhs, music_dvd;
}
```

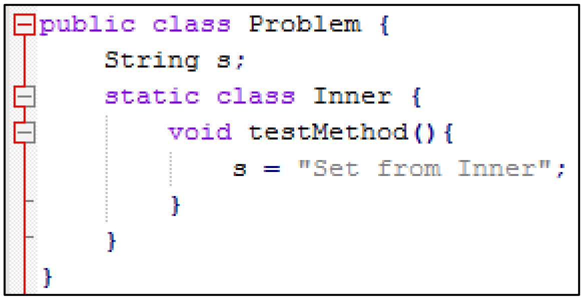


```
public class MediaFactory {
    public static void main(String[] args){
        System.out.println(MediaFactory.getMedia("Book"));
    }
    public static Media getMedia(String s){
        return Enum.valueOf(Media.class, s.toLowerCase());
    }
    public static Media getMedia(int n){
        return Media.values()[n];
    }
}
```

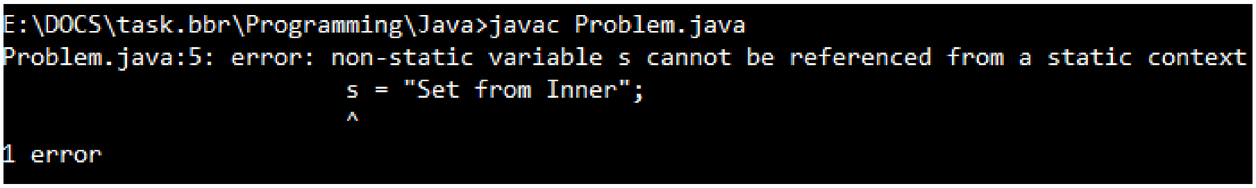


```
E:\DOCS\task.bbr\Programming\Java>javac MediaFactory.java

E:\DOCS\task.bbr\Programming\Java>java MediaFactory
book
```

Latihan 1: Program berikut tidak dapat dikompile. Perbaiaki agar dapat dikompile

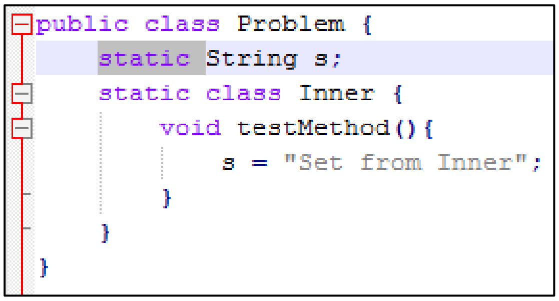


```
public class Problem {
    String s;
    static class Inner {
        void testMethod(){
            s = "Set from Inner";
        }
    }
}
```

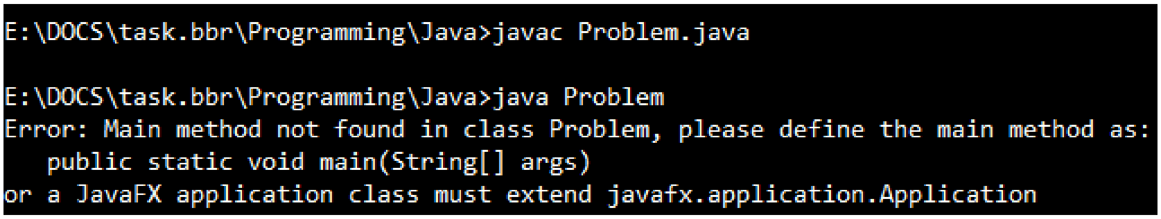


```
E:\DOCS\task.bbr\Programming\Java>javac Problem.java
Problem.java:5: error: non-static variable s cannot be referenced from a static context
                        s = "Set from Inner";
                        ^
1 error
```

Setelah diperbaiki:



```java
public class Problem {
    static String s;
    static class Inner {
        void testMethod(){
            s = "Set from Inner";
        }
    }
}
```

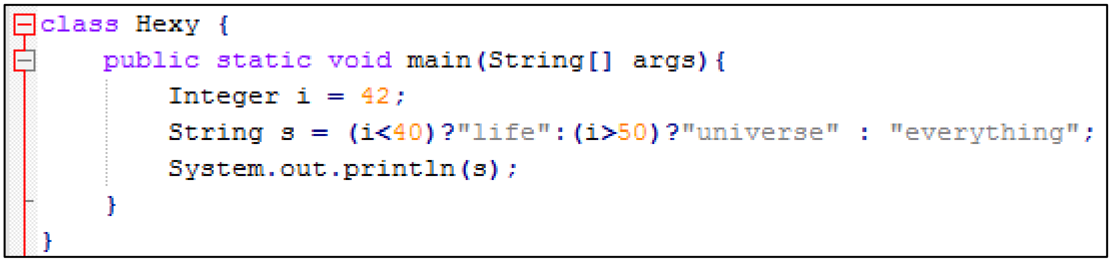


```
E:\DOCS\task.bbr\Programming\Java>javac Problem.java

E:\DOCS\task.bbr\Programming\Java>java Problem
Error: Main method not found in class Problem, please define the main method as:
   public static void main(String[] args)
or a JavaFX application class must extend javafx.application.Application
```

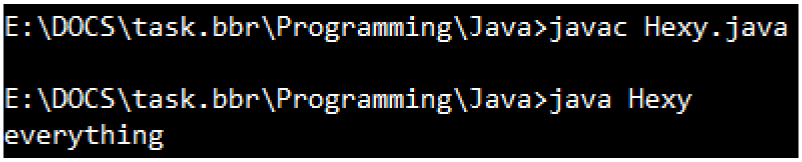
Latihan 2: Gunakan Java API documentation untuk Box class (di javax.swing package) untuk menjawab pertanyaan berikut.

1. What static nested class does Box define? `Box.Filler`
2. What inner class does Box define? `Box.AccessibleBox`
3. What is the superclass of Box's inner class? `[java.awt.]Container.AccessibleAWTContainer`
4. Which of Box's nested classes can you use from any class? `Box.Filler`
5. How do you create an instance of Box's Filler class? `new Box.Filler(minDimension, prefDimension, maxDimension)`

Latihan 3: Terdapat program seperti di bawah ini!

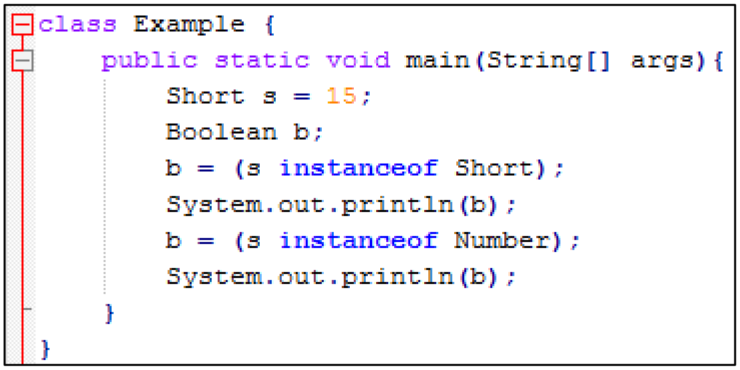


```java
class Hexy {
    public static void main(String[] args){
        Integer i = 42;
        String s = (i<40)?"life":(i>50)?"universe" : "everything";
        System.out.println(s);
    }
}
```

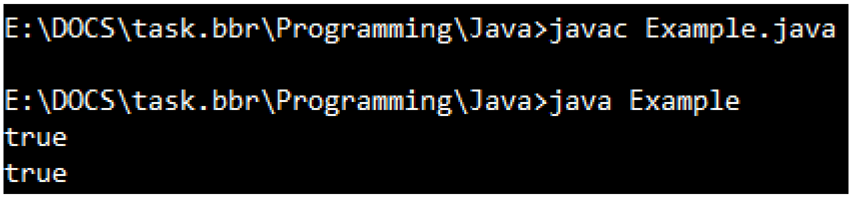


```
E:\DOCS\task.bbr\Programming\Java>javac Hexy.java

E:\DOCS\task.bbr\Programming\Java>java Hexy
everything
```

Latihan 4

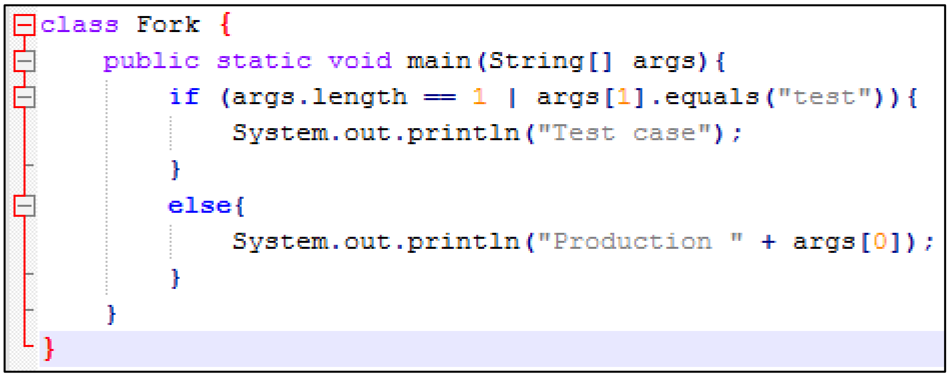


```
class Example {
    public static void main(String[] args){
        Short s = 15;
        Boolean b;
        b = (s instanceof Short);
        System.out.println(b);
        b = (s instanceof Number);
        System.out.println(b);
    }
}
```

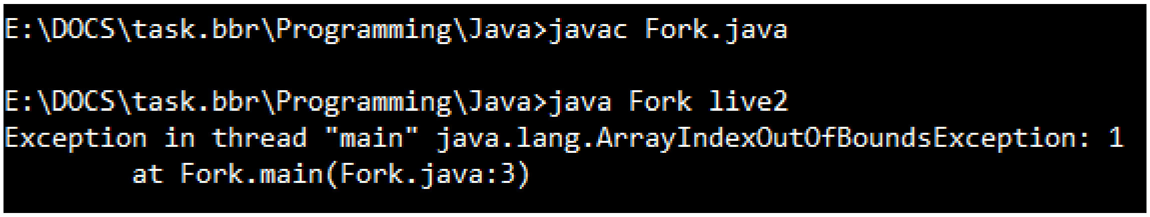


```
E:\DOCS\task.bbr\Programming\Java>javac Example.java

E:\DOCS\task.bbr\Programming\Java>java Example
true
true
```

Latihan 5

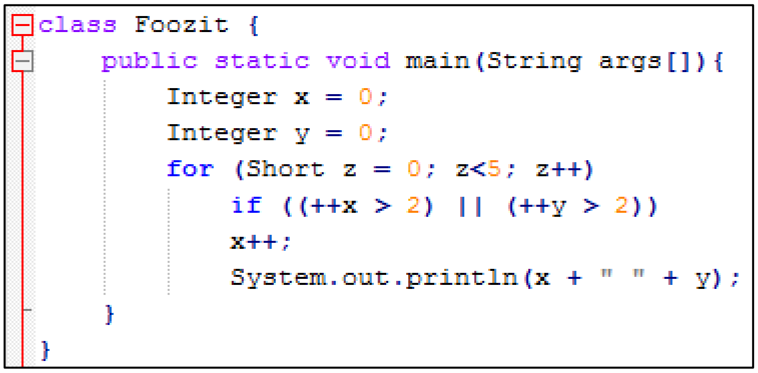


```
class Fork {
    public static void main(String[] args){
        if (args.length == 1 | args[1].equals("test")){
            System.out.println("Test case");
        }
        else{
            System.out.println("Production " + args[0]);
        }
    }
}
```

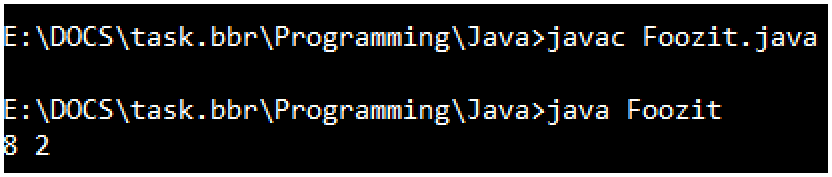


```
E:\DOCS\task.bbr\Programming\Java>javac Fork.java

E:\DOCS\task.bbr\Programming\Java>java Fork live2
Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: 1
        at Fork.main(Fork.java:3)
```

Latihan 6

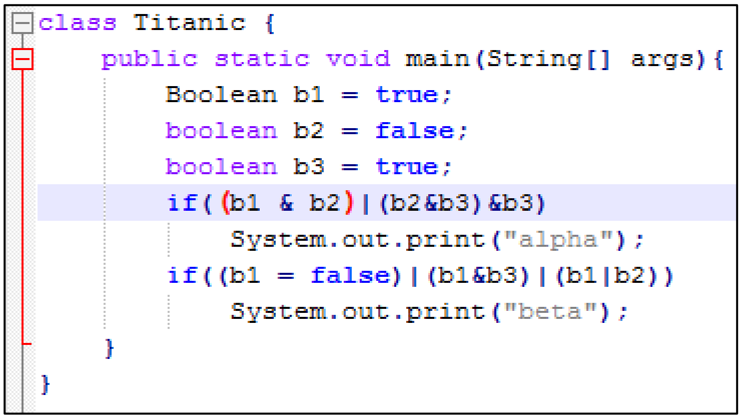


```
class Foozit {
    public static void main(String args[]){
        Integer x = 0;
        Integer y = 0;
        for (Short z = 0; z<5; z++)
            if ((++x > 2) || (++y > 2))
            x++;
            System.out.println(x + " " + y);
    }
}
```

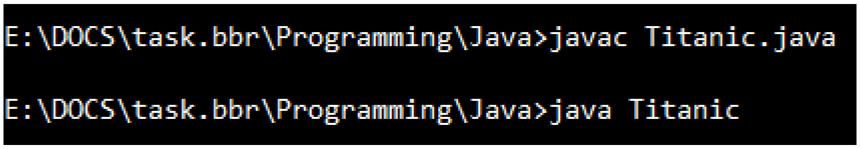


```
E:\DOCS\task.bbr\Programming\Java>javac Foozit.java

E:\DOCS\task.bbr\Programming\Java>java Foozit
8 2
```

Latihan 7

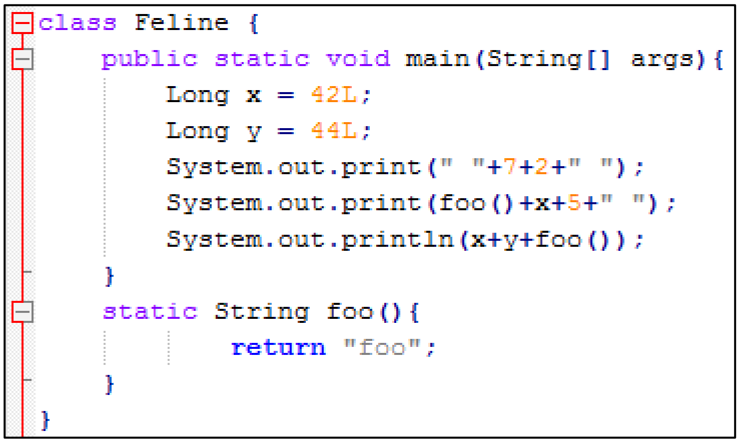


```
class Titanic {
    public static void main(String[] args){
        Boolean b1 = true;
        boolean b2 = false;
        boolean b3 = true;
        if((b1  & b2)|(b2&b3)&b3)
            System.out.print("alpha");
        if((b1 = false)|(b1&b3)|(b1|b2))
            System.out.print("beta");
    }
}
```

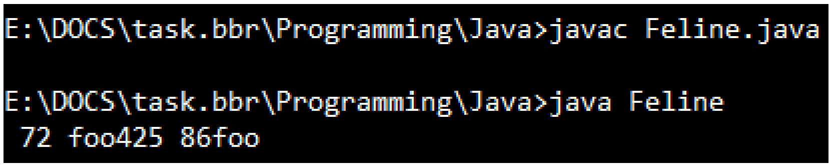


```
E:\DOCS\task.bbr\Programming\Java>javac Titanic.java

E:\DOCS\task.bbr\Programming\Java>java Titanic
```

Latihan 8

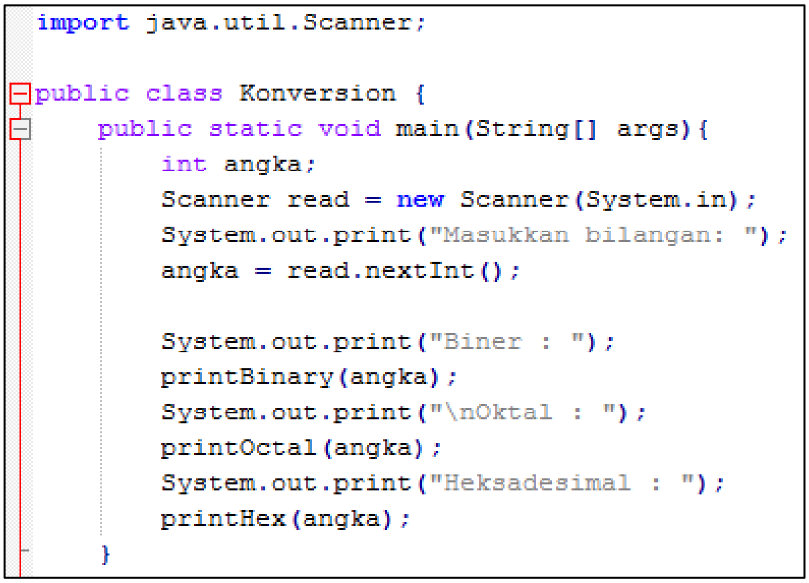


```
class Feline {
    public static void main(String[] args){
        Long x = 42L;
        Long y = 44L;
        System.out.print(" "+7+2+" ");
        System.out.print(foo()+x+5+" ");
        System.out.println(x+y+foo());
    }
    static String foo(){
            return "foo";
    }
}
```

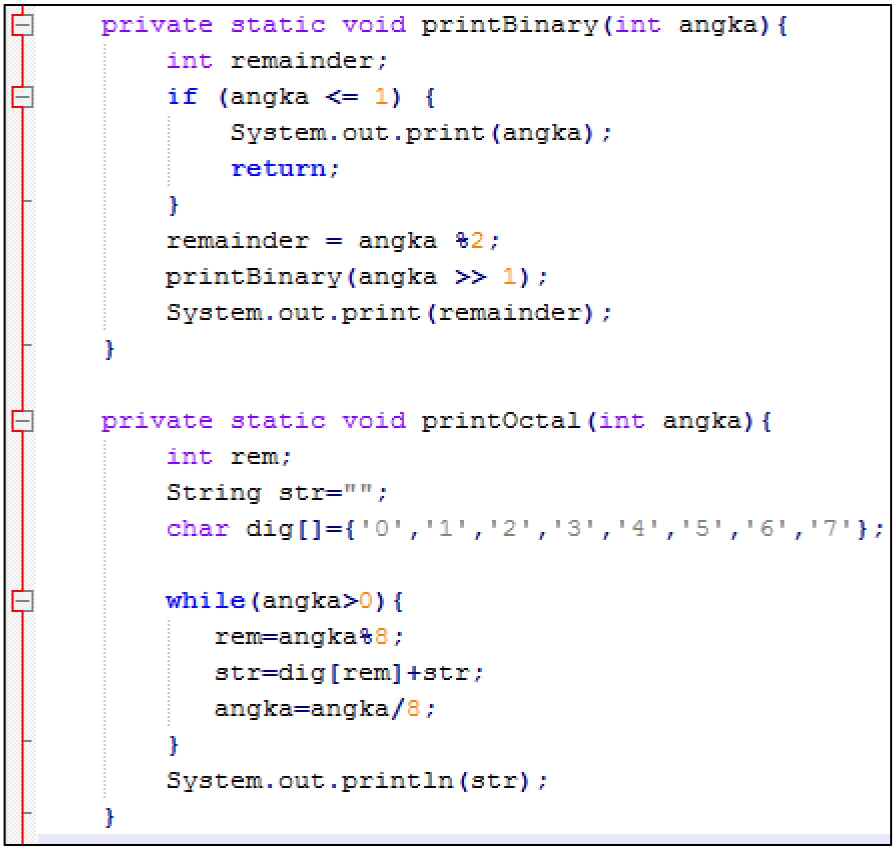


```
E:\DOCS\task.bbr\Programming\Java>javac Feline.java

E:\DOCS\task.bbr\Programming\Java>java Feline
 72 foo425 86foo
```

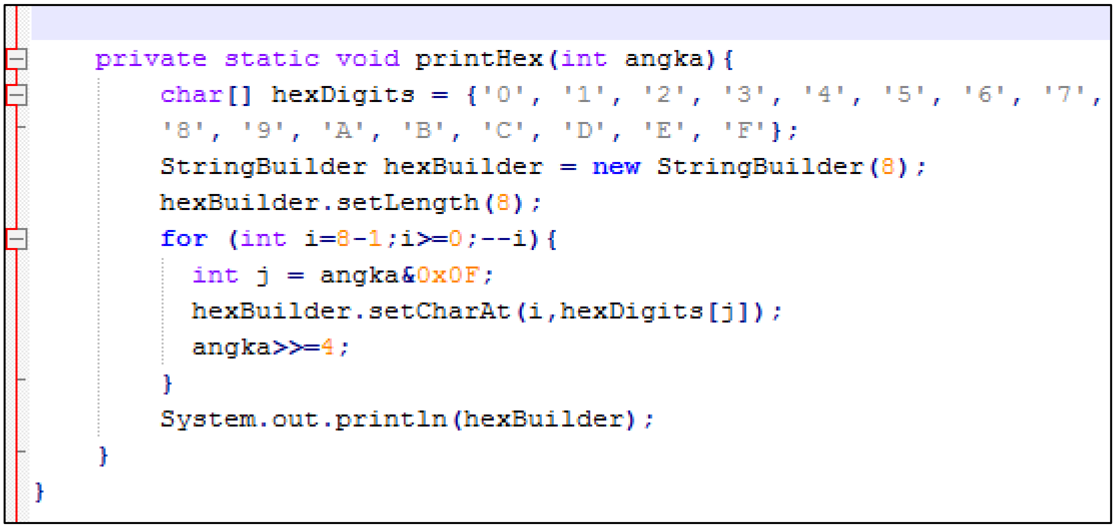
Latihan 9: Buatlah sebuah aplikasi untuk mengubah sebuah bilangan desimal menjadi bilangan biner, octal dan heksa berdasarkan inputan user!



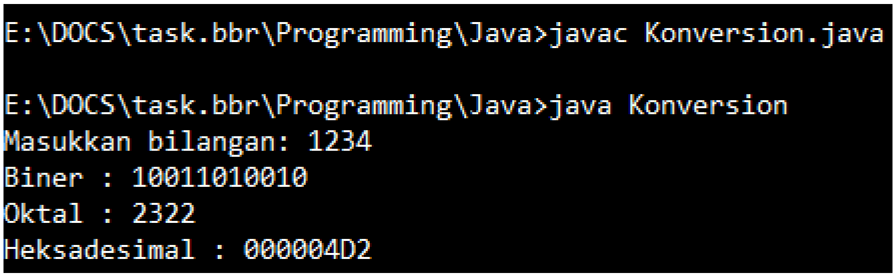
```
import java.util.Scanner;

public class Konversion {
    public static void main(String[] args){
        int angka;
        Scanner read = new Scanner(System.in);
        System.out.print("Masukkan bilangan: ");
        angka = read.nextInt();

        System.out.print("Biner : ");
        printBinary(angka);
        System.out.print("\nOktal : ");
        printOctal(angka);
        System.out.print("Heksadesimal : ");
        printHex(angka);
    }
```



```java
    private static void printBinary(int angka){
        int remainder;
        if (angka <= 1) {
            System.out.print(angka);
            return;
        }
        remainder = angka %2;
        printBinary(angka >> 1);
        System.out.print(remainder);
    }

    private static void printOctal(int angka){
        int rem;
        String str="";
        char dig[]={'0','1','2','3','4','5','6','7'};

        while(angka>0){
           rem=angka%8;
           str=dig[rem]+str;
           angka=angka/8;
        }
        System.out.println(str);
    }
```

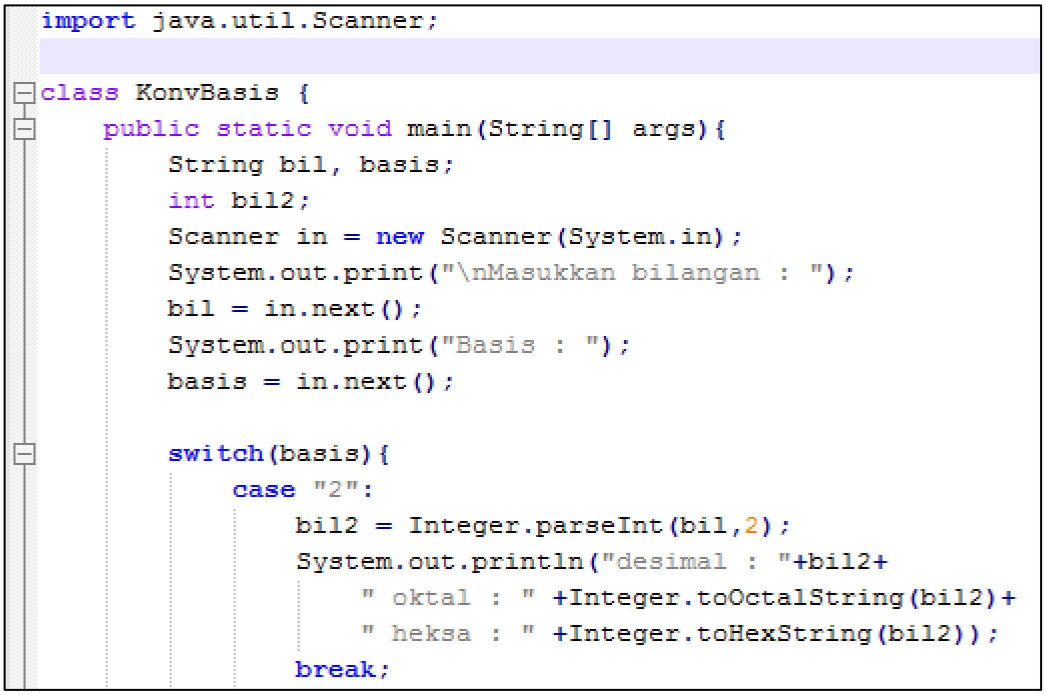


```java
    private static void printHex(int angka){
        char[] hexDigits = {'0', '1', '2', '3', '4', '5', '6', '7',
        '8', '9', 'A', 'B', 'C', 'D', 'E', 'F'};
        StringBuilder hexBuilder = new StringBuilder(8);
        hexBuilder.setLength(8);
        for (int i=8-1;i>=0;--i){
          int j = angka&0x0F;
          hexBuilder.setCharAt(i,hexDigits[j]);
          angka>>=4;
        }
        System.out.println(hexBuilder);
    }
}
```

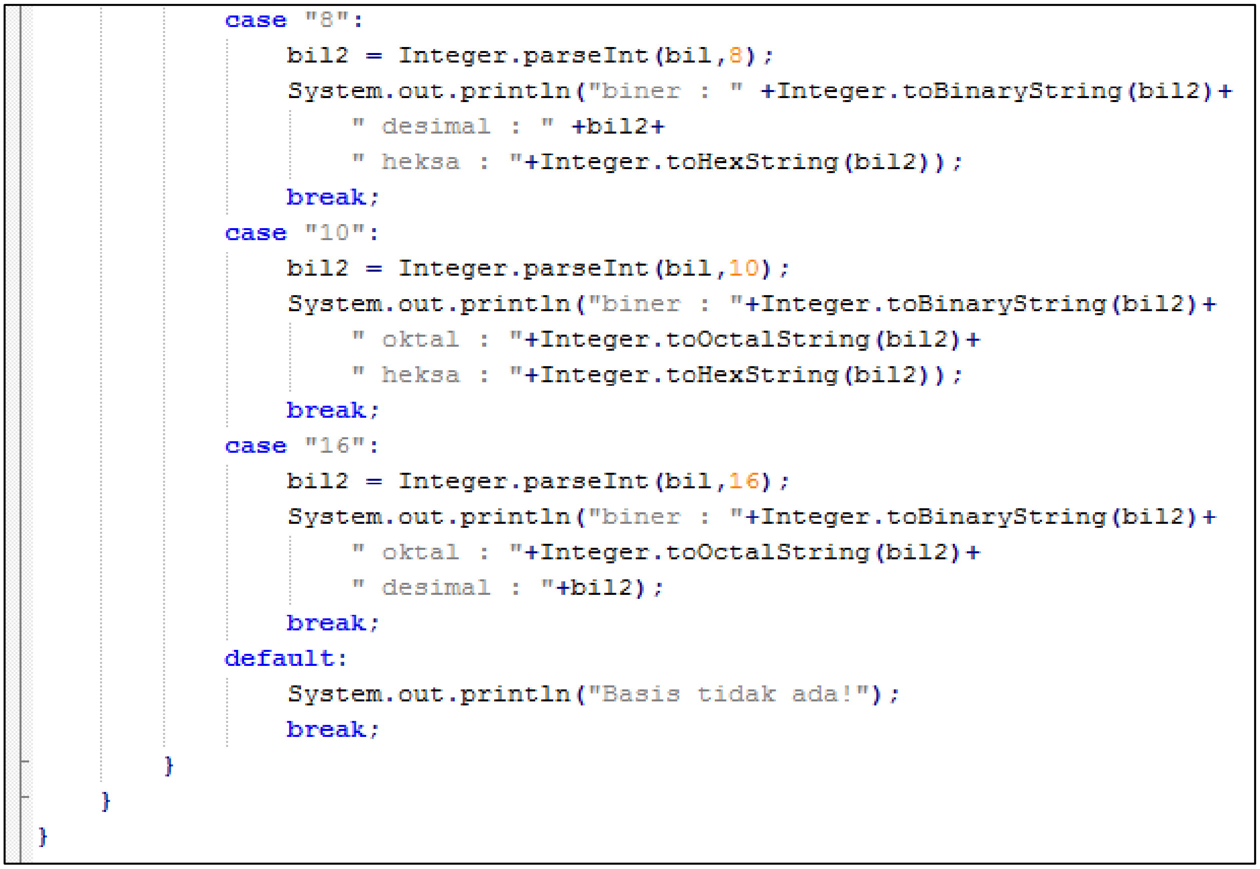


```
E:\DOCS\task.bbr\Programming\Java>javac Konversion.java

E:\DOCS\task.bbr\Programming\Java>java Konversion
Masukkan bilangan: 1234
Biner : 10011010010
Oktal : 2322
Heksadesimal : 000004D2
```

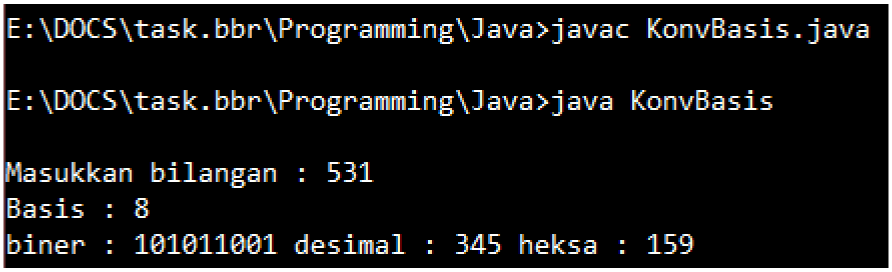
Latihan 10: Buatlah sebuah aplikasi menerima inputan user berupa bilangan dan basis dari bilangan tersebut, selanjutnya mengubah ke bilangan dengan basis yang lain (basis 2,8,10,16)



```
import java.util.Scanner;

class KonvBasis {
    public static void main(String[] args){
        String bil, basis;
        int bil2;
        Scanner in = new Scanner(System.in);
        System.out.print("\nMasukkan bilangan : ");
        bil = in.next();
        System.out.print("Basis : ");
        basis = in.next();

        switch(basis){
            case "2":
                bil2 = Integer.parseInt(bil,2);
                System.out.println("desimal : "+bil2+
                    " oktal : " +Integer.toOctalString(bil2)+
                    " heksa : " +Integer.toHexString(bil2));
                break;
```



```
            case "8":
                bil2 = Integer.parseInt(bil,8);
                System.out.println("biner : " +Integer.toBinaryString(bil2)+
                    " desimal : " +bil2+
                    " heksa : "+Integer.toHexString(bil2));
                break;
            case "10":
                bil2 = Integer.parseInt(bil,10);
                System.out.println("biner : "+Integer.toBinaryString(bil2)+
                    " oktal : "+Integer.toOctalString(bil2)+
                    " heksa : "+Integer.toHexString(bil2));
                break;
            case "16":
                bil2 = Integer.parseInt(bil,16);
                System.out.println("biner : "+Integer.toBinaryString(bil2)+
                    " oktal : "+Integer.toOctalString(bil2)+
                    " desimal : "+bil2);
                break;
            default:
                System.out.println("Basis tidak ada!");
                break;
        }
    }
}
```



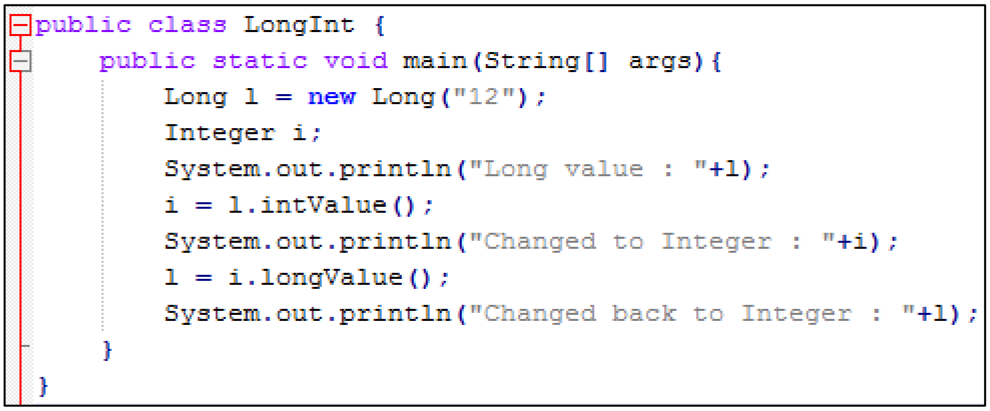
```
E:\DOCS\task.bbr\Programming\Java>javac KonvBasis.java

E:\DOCS\task.bbr\Programming\Java>java KonvBasis

Masukkan bilangan : 531
Basis : 8
biner : 101011001 desimal : 345 heksa : 159
```

Latihan 11:

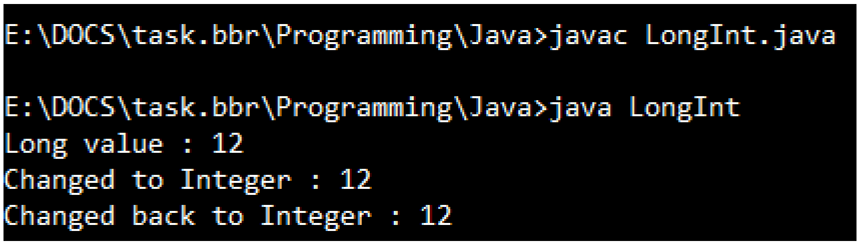
a. Bagaimana program ini jika dijalankan? Jelaskan!

Error terjadi karena variabel i dan j belum dideklarasikan.

b. Bagaimana cara mengubah objek Long menjadi objek Integer dan sebaliknya?

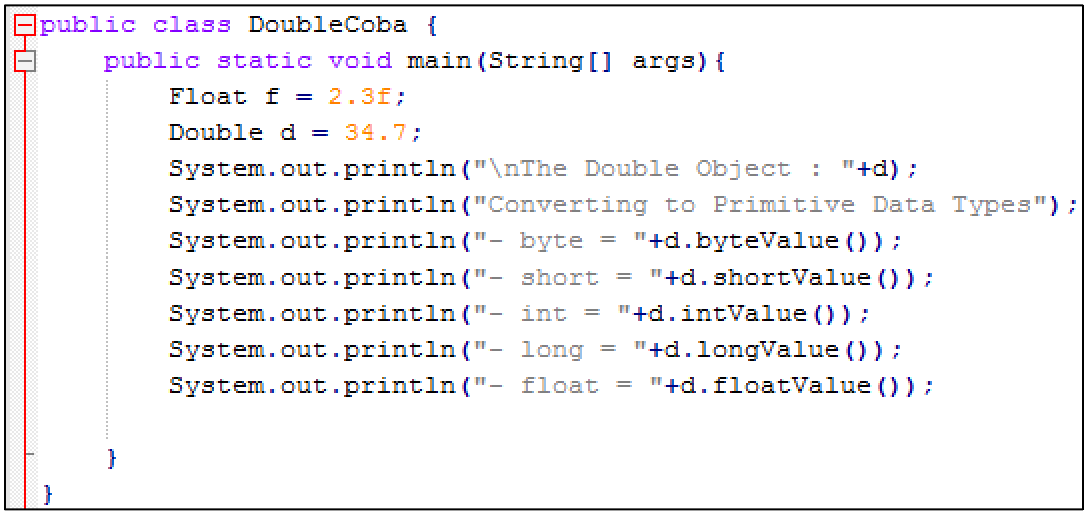


```java
public class LongInt {
    public static void main(String[] args){
        Long l = new Long("12");
        Integer i;
        System.out.println("Long value : "+l);
        i = l.intValue();
        System.out.println("Changed to Integer : "+i);
        l = i.longValue();
        System.out.println("Changed back to Integer : "+l);
    }
}
```

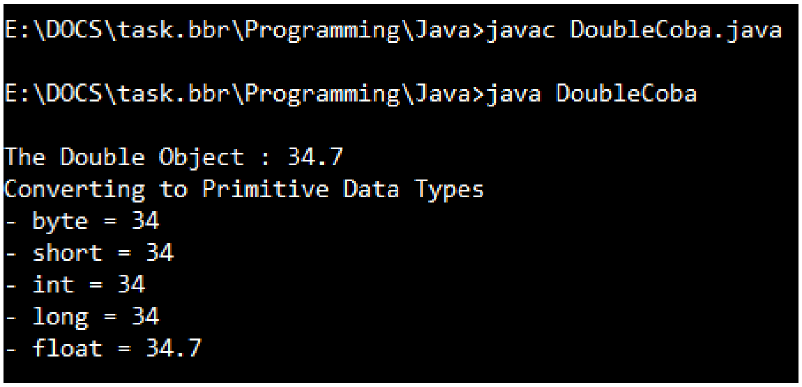


```
E:\DOCS\task.bbr\Programming\Java>javac LongInt.java

E:\DOCS\task.bbr\Programming\Java>java LongInt
Long value : 12
Changed to Integer : 12
Changed back to Integer : 12
```

Latihan 12: Ubahlah object Double menjadi tipe data primitif byte, short, int, long, float

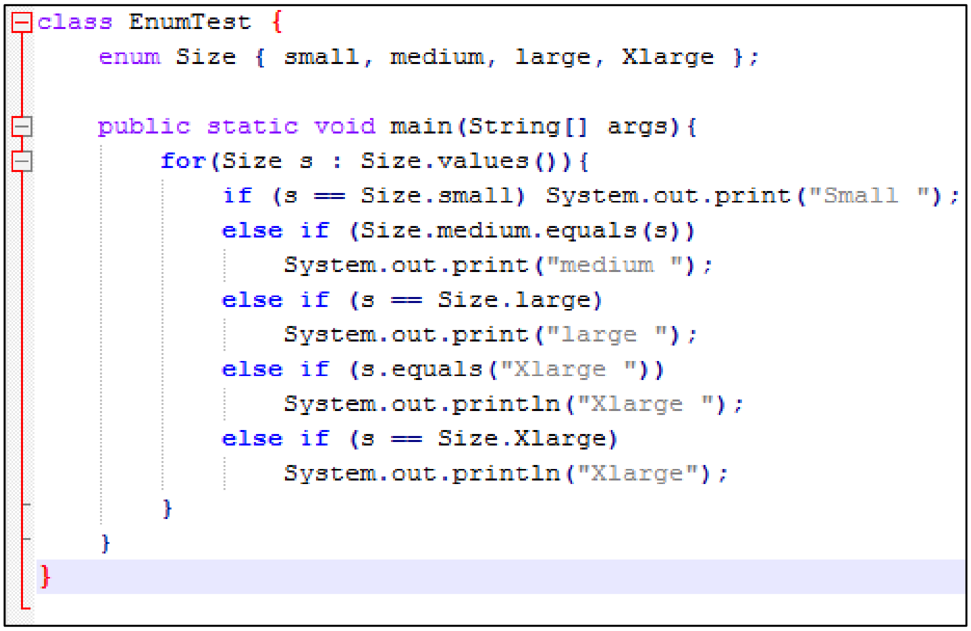


```java
public class DoubleCoba {
    public static void main(String[] args){
        Float f = 2.3f;
        Double d = 34.7;
        System.out.println("\nThe Double Object : "+d);
        System.out.println("Converting to Primitive Data Types");
        System.out.println("- byte = "+d.byteValue());
        System.out.println("- short = "+d.shortValue());
        System.out.println("- int = "+d.intValue());
        System.out.println("- long = "+d.longValue());
        System.out.println("- float = "+d.floatValue());

    }
}
```

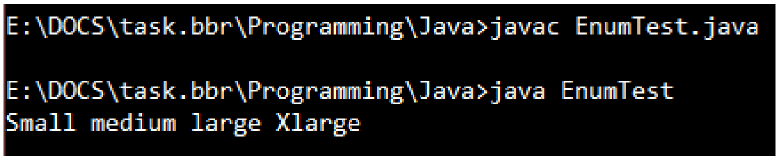


```
E:\DOCS\task.bbr\Programming\Java>javac DoubleCoba.java

E:\DOCS\task.bbr\Programming\Java>java DoubleCoba

The Double Object : 34.7
Converting to Primitive Data Types
- byte = 34
- short = 34
- int = 34
- long = 34
- float = 34.7
```

Latihan 13

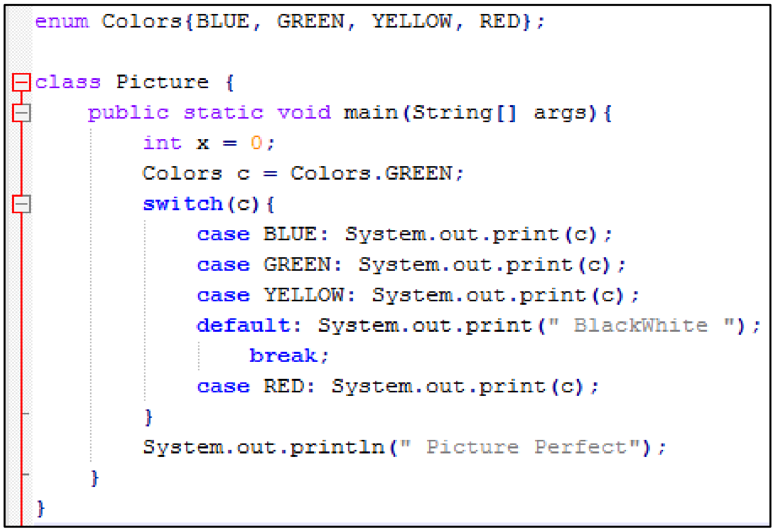


```java
class EnumTest {
    enum Size { small, medium, large, Xlarge };

    public static void main(String[] args){
        for(Size s : Size.values()){
            if (s == Size.small) System.out.print("Small ");
            else if (Size.medium.equals(s))
                System.out.print("medium ");
            else if (s == Size.large)
                System.out.print("large ");
            else if (s.equals("Xlarge "))
                System.out.println("Xlarge ");
            else if (s == Size.Xlarge)
                System.out.println("Xlarge");
        }
    }
}
```

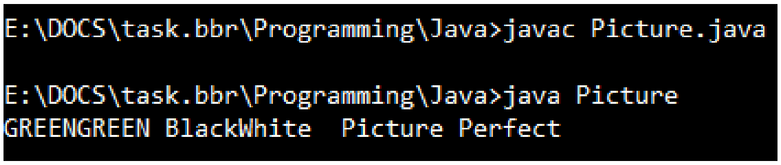


```
E:\DOCS\task.bbr\Programming\Java>javac EnumTest.java

E:\DOCS\task.bbr\Programming\Java>java EnumTest
Small medium large Xlarge
```

Latihan 14

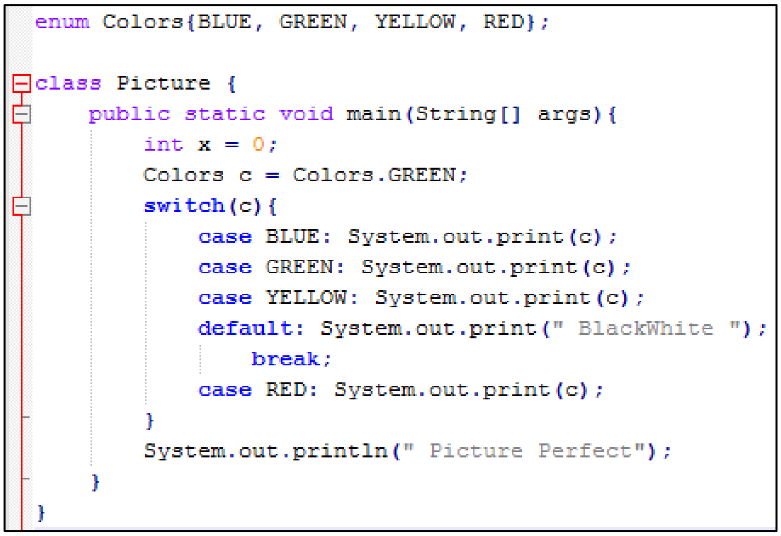


```java
enum Colors{BLUE, GREEN, YELLOW, RED};

class Picture {
    public static void main(String[] args){
        int x = 0;
        Colors c = Colors.GREEN;
        switch(c){
            case BLUE: System.out.print(c);
            case GREEN: System.out.print(c);
            case YELLOW: System.out.print(c);
            default: System.out.print(" BlackWhite ");
                break;
            case RED: System.out.print(c);
        }
        System.out.println(" Picture Perfect");
    }
}
```

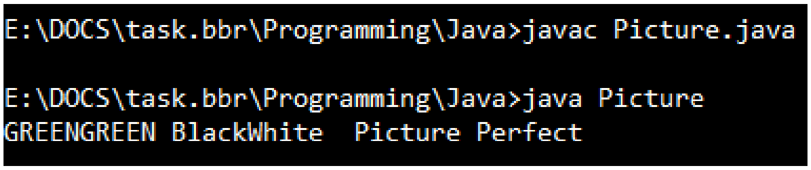


```
E:\DOCS\task.bbr\Programming\Java>javac Picture.java

E:\DOCS\task.bbr\Programming\Java>java Picture
GREENGREEN BlackWhite  Picture Perfect
```

Latihan 15

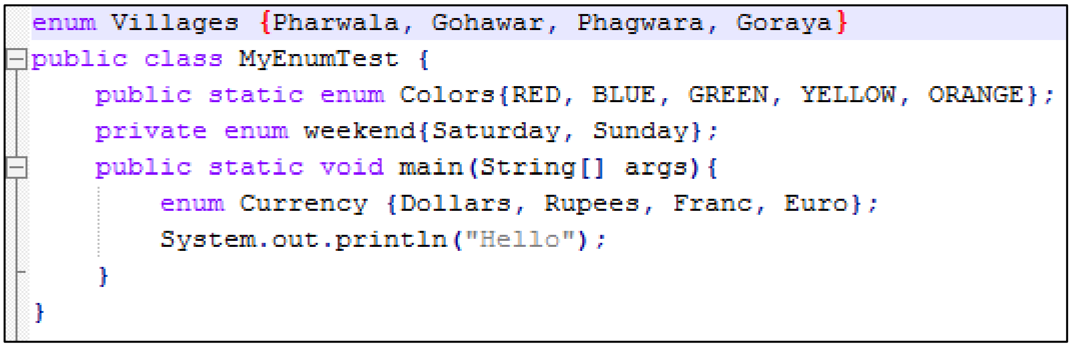


```java
enum Colors{BLUE, GREEN, YELLOW, RED};

class Picture {
    public static void main(String[] args){
        int x = 0;
        Colors c = Colors.GREEN;
        switch(c){
            case BLUE: System.out.print(c);
            case GREEN: System.out.print(c);
            case YELLOW: System.out.print(c);
            default: System.out.print(" BlackWhite ");
                break;
            case RED: System.out.print(c);
        }
        System.out.println(" Picture Perfect");
    }
}
```

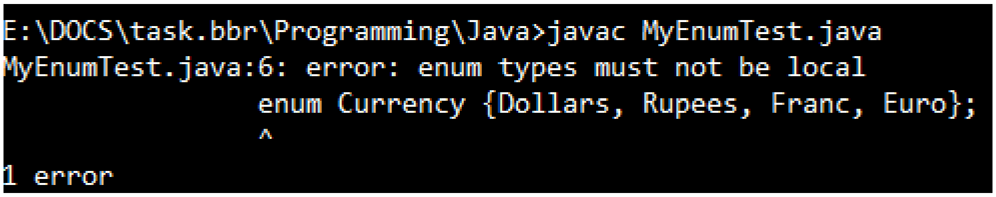


```
E:\DOCS\task.bbr\Programming\Java>javac Picture.java

E:\DOCS\task.bbr\Programming\Java>java Picture
GREENGREEN BlackWhite  Picture Perfect
```

Latihan 16

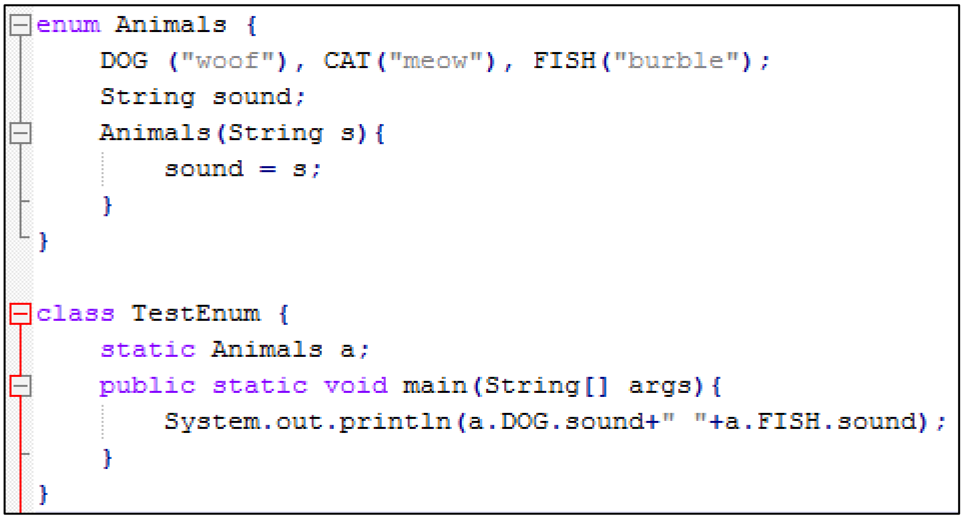


```java
enum Villages {Pharwala, Gohawar, Phagwara, Goraya}
public class MyEnumTest {
    public static enum Colors{RED, BLUE, GREEN, YELLOW, ORANGE};
    private enum weekend{Saturday, Sunday};
    public static void main(String[] args){
        enum Currency {Dollars, Rupees, Franc, Euro};
        System.out.println("Hello");
    }
}
```

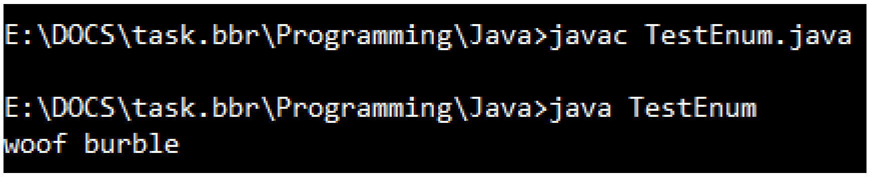


```
E:\DOCS\task.bbr\Programming\Java>javac MyEnumTest.java
MyEnumTest.java:6: error: enum types must not be local
                enum Currency {Dollars, Rupees, Franc, Euro};
                ^
1 error
```

Latihan 17



```java
enum Animals {
    DOG ("woof"), CAT("meow"), FISH("burble");
    String sound;
    Animals(String s){
        sound = s;
    }
}

class TestEnum {
    static Animals a;
    public static void main(String[] args){
        System.out.println(a.DOG.sound+" "+a.FISH.sound);
    }
}
```



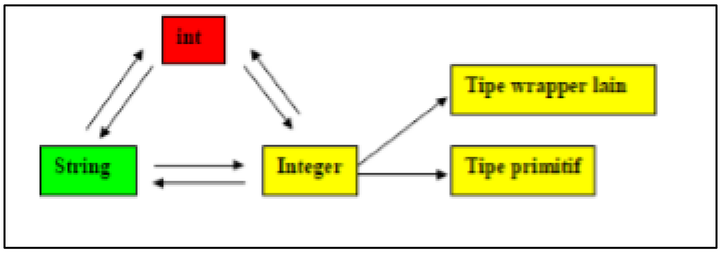
```
E:\DOCS\task.bbr\Programming\Java>javac TestEnum.java

E:\DOCS\task.bbr\Programming\Java>java TestEnum
woof burble
```

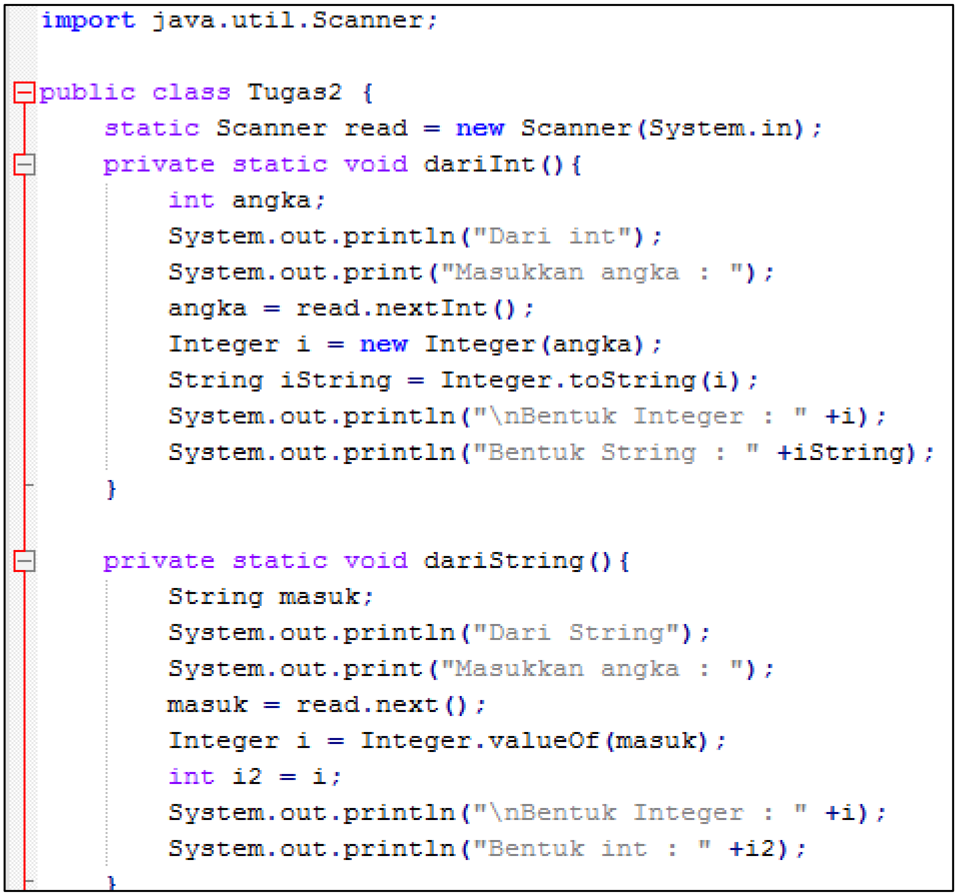
Tugas

1. Jelaskan kelebihan dan kelemahan penggunaan inner class!

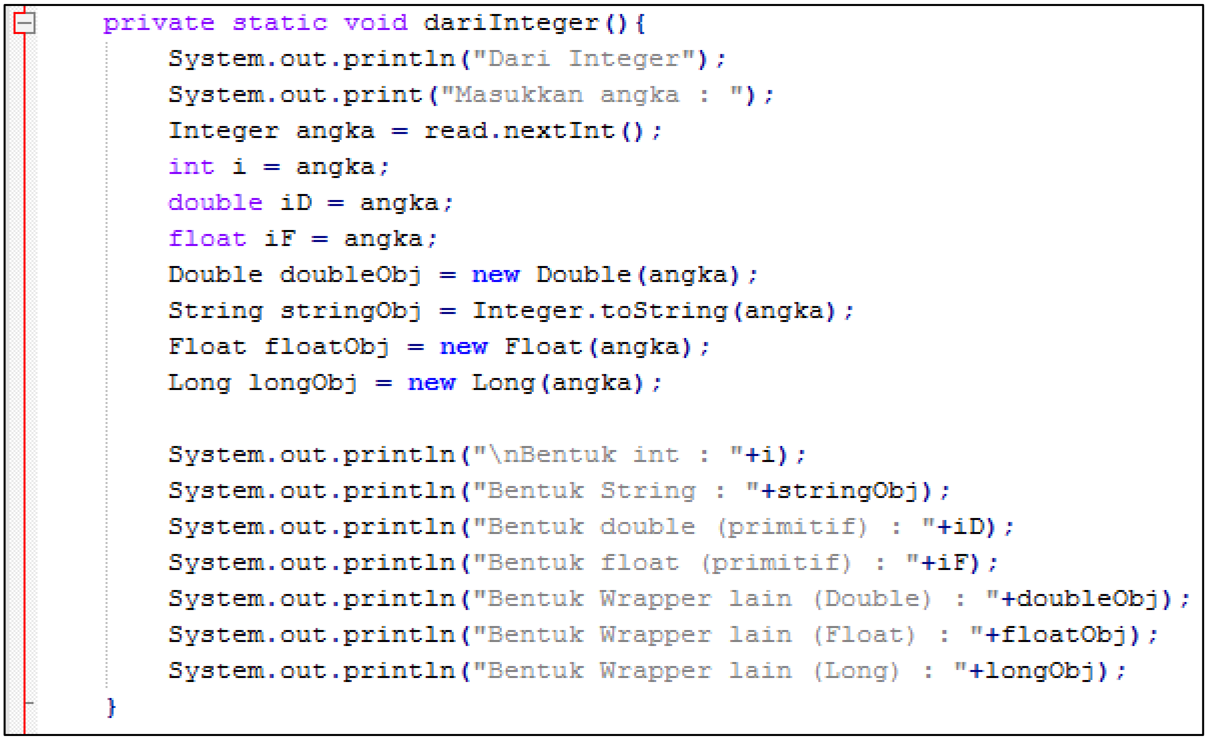
Dari sudut pandang pengorganisasian, inner class memungkinkan kita mengatur struktur paket. Daripada menjadikan segala sesuatu menjadi flat-package, kelas dapat disarangkan dalam kelas lain.

2. Buatlah suatu aplikasi untuk mengubah:

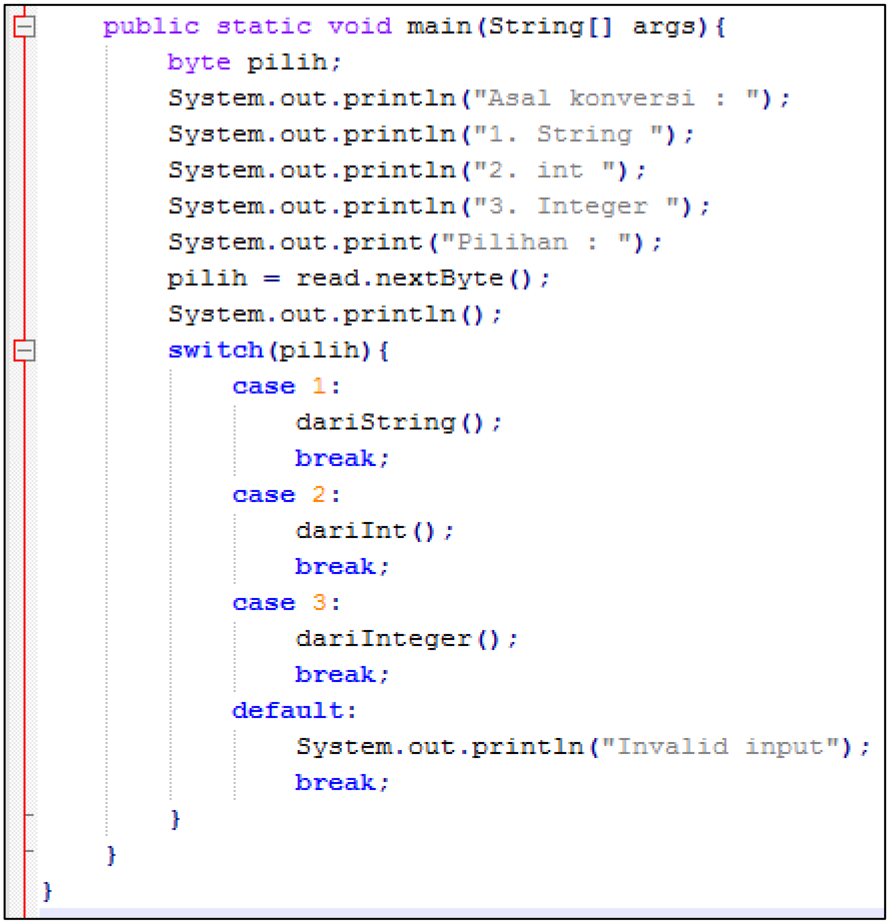




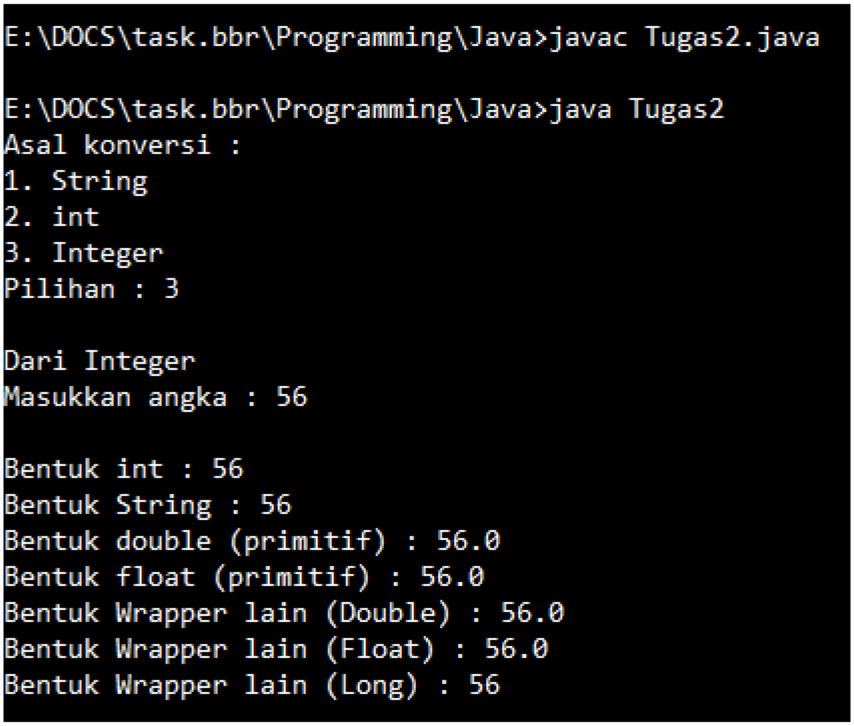
```java
import java.util.Scanner;

public class Tugas2 {
    static Scanner read = new Scanner(System.in);
    private static void dariInt(){
        int angka;
        System.out.println("Dari int");
        System.out.print("Masukkan angka : ");
        angka = read.nextInt();
        Integer i = new Integer(angka);
        String iString = Integer.toString(i);
        System.out.println("\nBentuk Integer : " +i);
        System.out.println("Bentuk String : " +iString);
    }

    private static void dariString(){
        String masuk;
        System.out.println("Dari String");
        System.out.print("Masukkan angka : ");
        masuk = read.next();
        Integer i = Integer.valueOf(masuk);
        int i2 = i;
        System.out.println("\nBentuk Integer : " +i);
        System.out.println("Bentuk int : " +i2);
    }
```



```java
    private static void dariInteger(){
        System.out.println("Dari Integer");
        System.out.print("Masukkan angka : ");
        Integer angka = read.nextInt();
        int i = angka;
        double iD = angka;
        float iF = angka;
        Double doubleObj = new Double(angka);
        String stringObj = Integer.toString(angka);
        Float floatObj = new Float(angka);
        Long longObj = new Long(angka);

        System.out.println("\nBentuk int : "+i);
        System.out.println("Bentuk String : "+stringObj);
        System.out.println("Bentuk double (primitif) : "+iD);
        System.out.println("Bentuk float (primitif) : "+iF);
        System.out.println("Bentuk Wrapper lain (Double) : "+doubleObj);
        System.out.println("Bentuk Wrapper lain (Float) : "+floatObj);
        System.out.println("Bentuk Wrapper lain (Long) : "+longObj);
    }
```

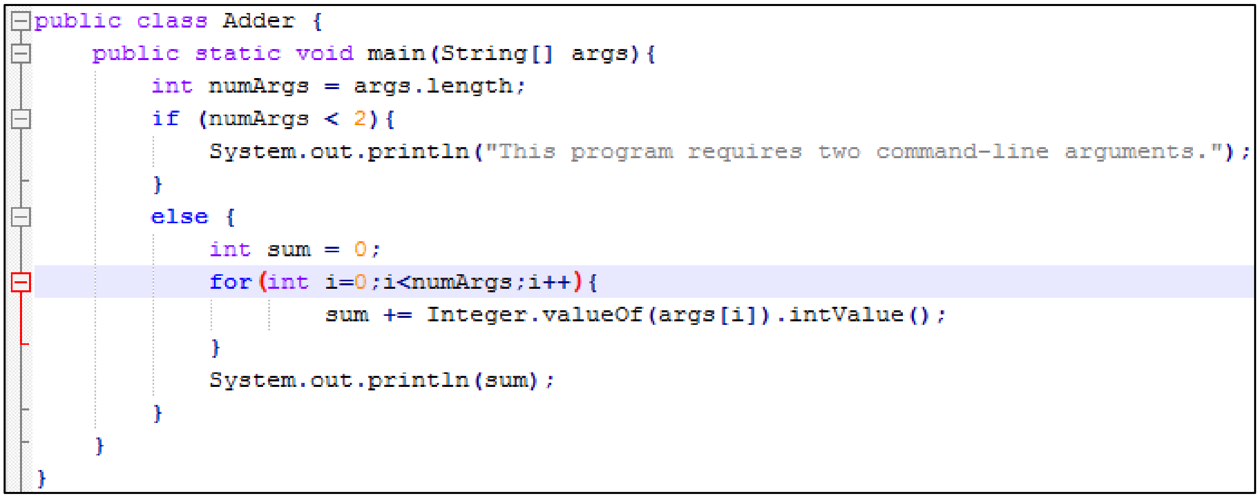


```java
    public static void main(String[] args){
        byte pilih;
        System.out.println("Asal konversi : ");
        System.out.println("1. String ");
        System.out.println("2. int ");
        System.out.println("3. Integer ");
        System.out.print("Pilihan : ");
        pilih = read.nextByte();
        System.out.println();
        switch(pilih){
            case 1:
                dariString();
                break;
            case 2:
                dariInt();
                break;
            case 3:
                dariInteger();
                break;
            default:
                System.out.println("Invalid input");
                break;
        }
    }
}
```

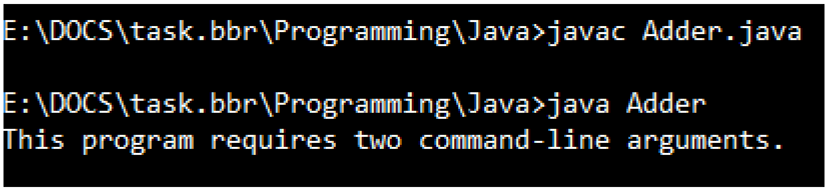


```
E:\DOCS\task.bbr\Programming\Java>javac Tugas2.java

E:\DOCS\task.bbr\Programming\Java>java Tugas2
Asal konversi :
1. String
2. int
3. Integer
Pilihan : 3

Dari Integer
Masukkan angka : 56

Bentuk int : 56
Bentuk String : 56
Bentuk double (primitif) : 56.0
Bentuk float (primitif) : 56.0
Bentuk Wrapper lain (Double) : 56.0
Bentuk Wrapper lain (Float) : 56.0
Bentuk Wrapper lain (Long) : 56
```

3. Buatlah sebuah program untuk menerima inputan beberapa bilangan (Subclass Number) dari command line dan menambahkannya.

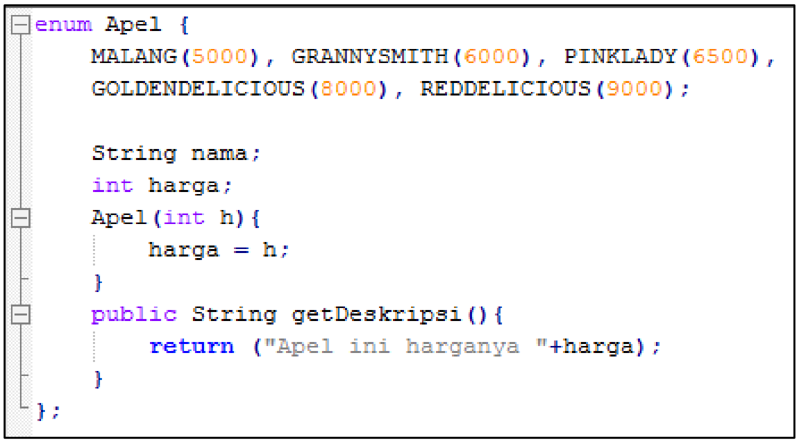


```java
public class Adder {
    public static void main(String[] args){
        int numArgs = args.length;
        if (numArgs < 2){
            System.out.println("This program requires two command-line arguments.");
        }
        else {
            int sum = 0;
            for(int i=0;i<numArgs;i++){
                sum += Integer.valueOf(args[i]).intValue();
            }
            System.out.println(sum);
        }
    }
}
```

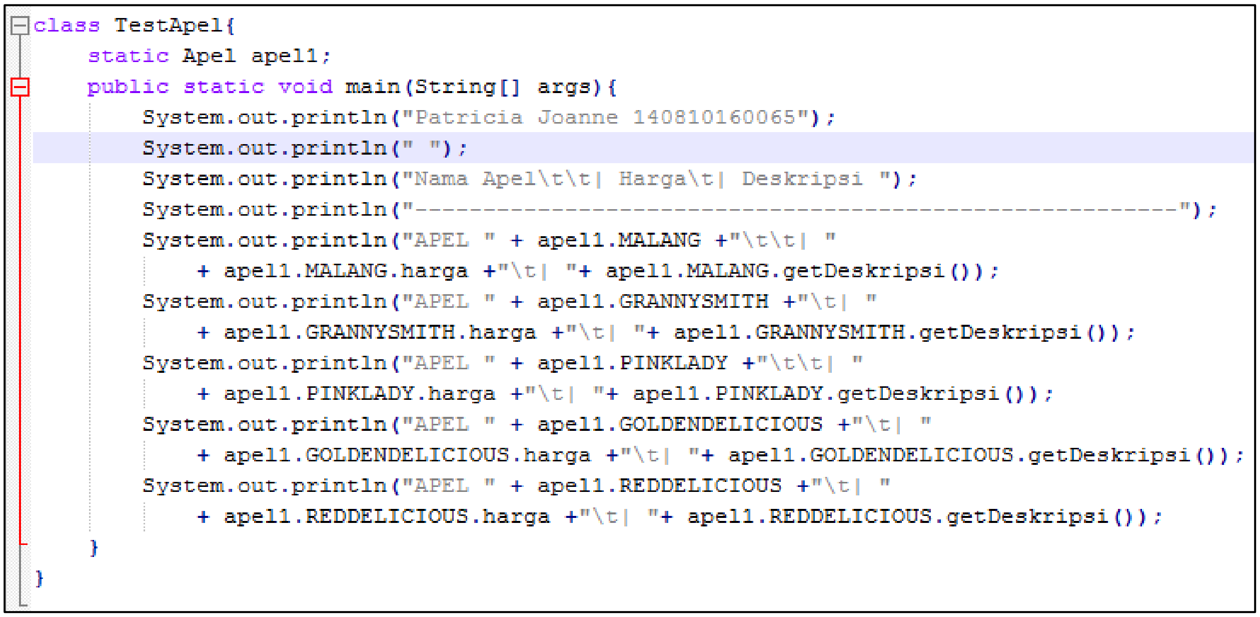


```
E:\DOCS\task.bbr\Programming\Java>javac Adder.java

E:\DOCS\task.bbr\Programming\Java>java Adder
This program requires two command-line arguments.
```

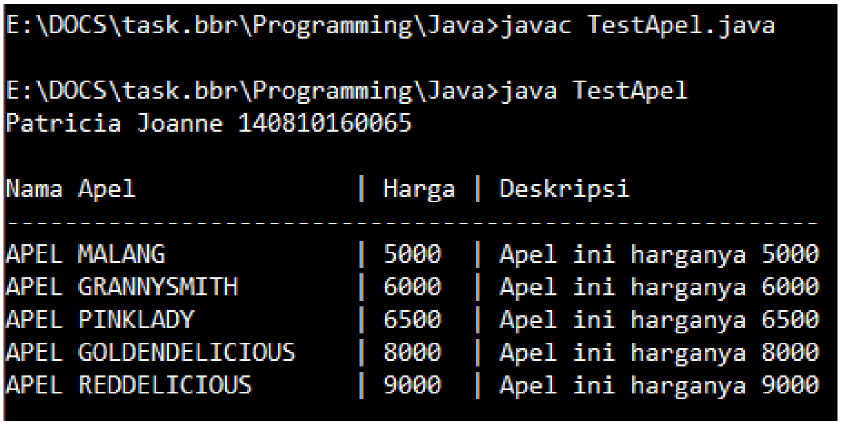
4. Pada supermarket terdapat beberapa macam jenis buah Apel yaitu Apel Malang, Granny Smith, Pink Lady, Golden Delicious, Gala dan Red Delicious. Buatlah enum Apel dengan berbagai jenis apel dan harganya. Berikan deskripsi dari apel tersebut pada method getDeskripsi(). Selanjutnya tampilkan data semua Apel, harga dan deskripsi pada Supermarket tersebut.



```
enum Apel {
    MALANG(5000), GRANNYSMITH(6000), PINKLADY(6500),
    GOLDENDELICIOUS(8000), REDDELICIOUS(9000);

    String nama;
    int harga;
    Apel(int h){
        harga = h;
    }
    public String getDeskripsi(){
        return ("Apel ini harganya "+harga);
    }
};
```



```
class TestApel{
    static Apel apel1;
    public static void main(String[] args){
        System.out.println("Patricia Joanne 140810160065");
        System.out.println(" ");
        System.out.println("Nama Apel\t\t| Harga\t| Deskripsi ");
        System.out.println("--------------------------------------------------------");
        System.out.println("APEL " + apel1.MALANG +"\t\t| "
            + apel1.MALANG.harga +"\t| "+ apel1.MALANG.getDeskripsi());
        System.out.println("APEL " + apel1.GRANNYSMITH +"\t| "
            + apel1.GRANNYSMITH.harga +"\t| "+ apel1.GRANNYSMITH.getDeskripsi());
        System.out.println("APEL " + apel1.PINKLADY +"\t\t| "
            + apel1.PINKLADY.harga +"\t| "+ apel1.PINKLADY.getDeskripsi());
        System.out.println("APEL " + apel1.GOLDENDELICIOUS +"\t| "
            + apel1.GOLDENDELICIOUS.harga +"\t| "+ apel1.GOLDENDELICIOUS.getDeskripsi());
        System.out.println("APEL " + apel1.REDDELICIOUS +"\t| "
            + apel1.REDDELICIOUS.harga +"\t| "+ apel1.REDDELICIOUS.getDeskripsi());
    }
}
```



```
E:\DOCS\task.bbr\Programming\Java>javac TestApel.java

E:\DOCS\task.bbr\Programming\Java>java TestApel
Patricia Joanne 140810160065

Nama Apel               | Harga | Deskripsi
--------------------------------------------------------
APEL MALANG             | 5000  | Apel ini harganya 5000
APEL GRANNYSMITH        | 6000  | Apel ini harganya 6000
APEL PINKLADY           | 6500  | Apel ini harganya 6500
APEL GOLDENDELICIOUS    | 8000  | Apel ini harganya 8000
APEL REDDELICIOUS       | 9000  | Apel ini harganya 9000
```