Question: Define a class *Television*. Define a method for displaying the attributes value of a TV. Use them from main() method by creating two objects.

Code:

```
import java.util.Scanner;
import java.util.Calendar;
import java.util.GregorianCalendar;

public class Television {
    String maker;
    int sizeOfScreen;
    Calendar purchaseDate;
    boolean isColorTV;

    Television(String maker, int sizeOfScreen, Calendar purchaseDate,
boolean isColorTV) {
        this.maker = maker;
        this.sizeOfScreen = sizeOfScreen;
        this.purchaseDate = purchaseDate;
        this.isColorTV = isColorTV;
    }

    String print() {
        return ("Company: " + this.maker + " Size: " + this.sizeOfScreen
+" Color: " + this.isColorTV + " PurchaseDate: "
                + this.purchaseDate.get(Calendar.DATE) + "/" +
this.purchaseDate.get(Calendar.MONTH) + "/"
                + this.purchaseDate.get(Calendar.YEAR));
    }

    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        String maker;
        int sizeOfScreen;
        Calendar date = new GregorianCalendar(2020, 9, 25, 00, 00, 00);
        System.out.println("Maker: ");
        maker = input.nextLine();
        System.out.println("Size Of Screen: ");
        sizeOfScreen = Integer.parseInt(input.nextLine());
        Television obj = new Television(maker, sizeOfScreen, date, true);
```

```
        System.out.println("Television: " + obj.print());
    }
}
```

Output:

```
PS D:\OOPS-PCC-CS593\Day 7 (25.09.2020)> javac Television.java
PS D:\OOPS-PCC-CS593\Day 7 (25.09.2020)> java Television
Maker:
Sony
Size Of Screen:
49
Television: Company: Sony Size: 49 Color: true PurchaseDate: 25/9/2020
```

Question: Show that static variable of a class only have one copy for different object but instance variable may have separate copy for individual object.

Code:

```
public class StaticNonStatic {
    int id;
    static String name = "Static Non-static";

    StaticNonStatic(int id) {
        this.id = id;
    }

    public String toString() {
        return "id = " + this.id + " name = " + StaticNonStatic.name;
    }

    public static void main(String[] args) {
        StaticNonStatic obj1 = new StaticNonStatic(12);
        StaticNonStatic obj2 = new StaticNonStatic(13);
        System.out.println("OBJ1 = " + obj1);
        System.out.println("OBJ2 = " + obj2);
    }
}
```

Output:

```
PS D:\OOPS-PCC-CS593\Day 7 (25.09.2020)> javac StaticNonStatic.java
PS D:\OOPS-PCC-CS593\Day 7 (25.09.2020)> java StaticNonStatic
OBJ1 = id = 12 name = Static Non-static
OBJ2 = id = 13 name = Static Non-static
```

Question: Write a **Tree** class to represent the trees in a firm. A Tree has the following attributes:

tree id, tree name, height, base width,
cost, amount.

Write a method named *annualUpdate( )* which updates the height, base width and the cost amount spent so far on this tree. Use them from main() method by creating two objects.

Code:

```
import java.util.Scanner;

public class Tree {
    int id;
    String name;
    double height, width, amount;

    Tree(int id, String name, double height, double width, double amount)
{
        this.id = id;
        this.name = name;
        this.height = height;
        this.width = width;
        this.amount = amount;
    }

    void annualUpdate(double height, double width, double amount) {
        this.height = height;
        this.width = width;
        this.amount = amount;
    }

    public String toString() {
        return "id = " + this.id + " name = " + this.name + " height = " +
this.height + " width = " + this.width
                + " amount = " + this.amount;
    }

    public static void main(String[] args) {
        Scanner  input = new Scanner(System.in);
        System.out.println("Enter: ");
```

```
        int id = Integer.parseInt(input.nextLine());
        String name = input.nextLine();
        double height = Double.parseDouble(input.nextLine());
        double width = Double.parseDouble(input.nextLine());
        double amount = Double.parseDouble(input.nextLine());

        Tree obj = new Tree(id, name, height, width, amount);
        System.out.println("Tree: " + obj);

    }
}
```

Output:

```
PS D:\OOPS-PCC-CS593\Day 7 (25.09.2020)> javac Tree.java
PS D:\OOPS-PCC-CS593\Day 7 (25.09.2020)> java Tree
Enter:
12
Mahogany
83.65
10.68
14000.99
Tree: id = 12 name = Mahogany height = 83.65 width = 10.68 amount = 14000.99
```