CMIS 141 – Homework 3
Ben Brandhorst
November 29th 2018

Input	Expected Output	Actual Output	Pass?
DEFAULT CONSTRUCTOR: Volume: MEDIUM plugged in: false manufacturer: AKG headPhoneColor: black headPhoneModel: K240 Studio **SETTER VALUES:** setVolume: LOW setPluggedIn: true setManufacturer: Bose setheadPhoneModel: Quiet Comfort 35 setHeadPhoneColor: white **SCANNER INPUT:** changeVolume: 2	****OUTPUT**** Default constructor results of toString for headPhone1: (HeadPhone manufacturer = AKG, HeadPhone model = K240 Studio, HeadPhone color = black, HeadPhone volume = 2, HeadPhone is plugged in = false) The setter method has been used to change data for headPhone1. New results are below: Results of getVolume: 1 Results of getPluggedIn: true Results of getManufacturer: Bose Results of getHeadPhoneModel: Quiet Comfort 35 Results of getHeadPhoneColor: white toString results after changes: (HeadPhone manufacturer = Bose, HeadPhone model = Quiet Comfort 35, HeadPhone color = white, HeadPhone volume = 1, HeadPhone is plugged in = true) Using the changeVolume method on headPhone1: Select your HeadPhone volume (1 = LOW, 2 = MEDIUM, 3 = HIGH) The volume has been set to the MEDIUM setting. Results of getVolume after using changeVolume method: volume = 2	****OUTPUT**** Default constructor results of toString for headPhone1: (HeadPhone manufacturer = AKG, HeadPhone model = K240 Studio, HeadPhone color = black, HeadPhone volume = 2, HeadPhone is plugged in = false) The setter method has been used to change data for headPhone1. New results are below: Results of getVolume: 1 Results of getPluggedIn: true Results of getManufacturer: Bose Results of getHeadPhoneModel: Quiet Comfort 35 Results of getHeadPhoneColor: white toString results after changes: (HeadPhone manufacturer = Bose, HeadPhone model = Quiet Comfort 35, HeadPhone color = white, HeadPhone volume = 1, HeadPhone is plugged in = true) Using the changeVolume method on headPhone1: Select your HeadPhone volume (1 = LOW, 2 = MEDIUM, 3 = HIGH) The volume has been set to the MEDIUM setting. Results of getVolume after using changeVolume method: volume = 2	YES
CONSTRUCTOR: Volume: LOW plugged in: false manufacturer: Sony headPhoneColor: violet headPhoneModel: PS4 Gold **SETTER VALUES:** setVolume: HIGH setPluggedIn: true	****OUTPUT**** toString results after using modified contructor for headPhone2: (HeadPhone manufacturer = Sony, HeadPhone model = PS4 Gold, HeadPhone color = violet, HeadPhone volume = 1, HeadPhone is plugged in = false) The setter method has been used to change data for headPhone2. New results are below: Results of getVolume: 3 Results of getPluggedIn: true	****OUTPUT**** toString results after using modified contructor for headPhone2: (HeadPhone manufacturer = Sony, HeadPhone model = PS4 Gold, HeadPhone color = violet, HeadPhone volume = 1, HeadPhone is plugged in = false) The setter method has been used to change data for headPhone2. New results are below: Results of getVolume: 3 Results of getPluggedIn: true	YES

setManufacturer: Sennheiser setheadPhoneModel: PXC 550 setHeadPhoneColor: orange **SCANNER INPUT:** changeVolume: 1	Results of getManufacturer: Sennheiser Results of getHeadPhoneModel: PXC 550 Results of getHeadPhoneColor: orange toString results after changes: (HeadPhone manufacturer = Sennheiser, HeadPhone model = PXC 550, HeadPhone color = orange, HeadPhone volume = 3, HeadPhone is plugged in = true) Using the changeVolume method on headPhone2: Select your HeadPhone volume (1 = LOW, 2 = MEDIUM, 3 = HIGH) The volume has been set to the LOW setting. Results of getVolume after using changeVolume method: volume = 1	Results of getManufacturer: Sennheiser Results of getHeadPhoneModel: PXC 550 Results of getHeadPhoneColor: orange toString results after changes: (HeadPhone manufacturer = Sennheiser, HeadPhone model = PXC 550, HeadPhone color = orange, HeadPhone volume = 3, HeadPhone is plugged in = true) Using the changeVolume method on headPhone2: Select your HeadPhone volume (1 = LOW, 2 = MEDIUM, 3 = HIGH) The volume has been set to the LOW setting. Results of getVolume after using changeVolume method: volume = 1	
CONSTRUCTOR: Volume: IHGH plugged in: true manufacturer: Dr. Dre headPhoneColor: yellow headPhoneModel: Beats **SETTER VALUES:** setVolume: LOW setPluggedIn: false setManufacturer: Audio Technica setheadPhoneModel: ATH M20x setHeadPhoneColor: blue **SCANNER INPUT:** changeVolume: 2	****OUTPUT**** toString results after using modified contructor for headPhone3: (HeadPhone manufacturer = Dr. Dre, HeadPhone model = Beats, HeadPhone color = yellow, HeadPhone volume = 3, HeadPhone is plugged in = true) The setter method has been used to change data for headPhone3. New results are below: Results of getVolume: 1 Results of getPluggedIn: false Results of getManufacturer: Audio Technica Results of getHeadPhoneModel: ATH M20x Results of getHeadPhoneColor: blue toString results after changes: (HeadPhone manufacturer = Audio Technica, HeadPhone model = ATH M20x, HeadPhone color = blue, HeadPhone volume = 1, HeadPhone is plugged in = false) Using the changeVolume method on headPhone3: Select your HeadPhone volume (1 = LOW, 2 = MEDIUM, 3 = HIGH)	****OUTPUT**** toString results after using modified contructor for headPhone3: (HeadPhone manufacturer = Dr. Dre, HeadPhone model = Beats, HeadPhone color = yellow, HeadPhone volume = 3, HeadPhone is plugged in = true) The setter method has been used to change data for headPhone3. New results are below: Results of getVolume: 1 Results of getPluggedIn: false Results of getManufacturer: Audio Technica Results of getHeadPhoneModel: ATH M20x Results of getHeadPhoneColor: blue toString results after changes: (HeadPhone manufacturer = Audio Technica, HeadPhone model = ATH M20x, HeadPhone color = blue, HeadPhone volume = 1, HeadPhone is plugged in = false) Using the changeVolume method on headPhone3: Select your HeadPhone volume (1 = LOW, 2 = MEDIUM, 3 = HIGH)	YES

	The volume has been set to the MEDIUM setting. Results of getVolume after using changeVolume method: volume = 2	The volume has been set to the MEDIUM setting. Results of getVolume after using changeVolume method: volume = 2	

TEST 1 Output:

```
Default constructor results of toString for headPhone1: (HeadPhone manufacturer = AKG, HeadPhone model = K240 Studio, HeadPhone color = black, HeadPhone volume = 2, HeadPhone is plugged in = false)
The setter method has been used to change data for headPhone1. New results are below:
Results of getVolume: 1
Results of getPluggedIn: true
Results of getManufacturer: Bose
Results of getHeadPhoneModel: Quiet Comfort 35
Results of getHeadPhoneColor: white
toString results after changes: (HeadPhone manufacturer = Bose, HeadPhone model = Quiet Comfort 35, HeadPhone color = white, HeadPhone volume = 1, HeadPhone is plugged in = true)
Using the changeVolume method on headPhone1:
Select your HeadPhone volume (1 = LOW, 2 = MEDIUM, 3 = HIGH)
2
The volume has been set to the MEDIUM setting.
Results of getVolume after using changeVolume method: volume = 2
```

TEST 2 Ouput:

```
toString results after using modified contructor for headPhone2: (HeadPhone manufacturer = Sony, HeadPhone model = PS4 Gold, HeadPhone color = violet, HeadPhone volume = 1, HeadPhone is plugged in = false)
The setter method has been used to change data for headPhone2. New results are below:
Results of getVolume: 3
Results of getPluggedIn: true
Results of getManufacturer: Sennheiser
Results of getHeadPhoneModel: PXC 550
Results of getHeadPhoneColor: orange
toString results after changes: (HeadPhone manufacturer = Sennheiser, HeadPhone model = PXC 550, HeadPhone color = orange, HeadPhone volume = 3, HeadPhone is plugged in = true)
Using the changeVolume method on headPhone2:
Select your HeadPhone volume (1 = LOW, 2 = MEDIUM, 3 = HIGH)
1
The volume has been set to the LOW setting.
Results of getVolume after using changeVolume method: volume = 1
```

TEST 3 Output:

```
toString results after using modified contructor for headPhone3: (HeadPhone manufacturer = Dr. Dre, HeadPhone model = Beats, HeadPhone color = yellow, HeadPhone volume = 3, HeadPhone is plugged in = true)
The setter method has been used to change data for headPhone3. New results are below:
Results of getVolume: 1
Results of getPluggedIn: false
Results of getManufacturer: Audio Technica
Results of getHeadPhoneModel: ATH M20x
Results of getHeadPhoneColor: blue
toString results after changes: (HeadPhone manufacturer = Audio Technica, HeadPhone model = ATH M20x, HeadPhone color = blue, HeadPhone volume = 1, HeadPhone is plugged in = false)
Using the changeVolume method on headPhone3:
Select your HeadPhone volume (1 = LOW, 2 = MEDIUM, 3 = HIGH)
2
The volume has been set to the MEDIUM setting.
Results of getVolume after using changeVolume method: volume = 2
```

HeadPhone.java Example:

```java
import java.util.*;


/*
 * File: HeadPhone.java
 * Author: Ben Brandhorst
 * Date: November 27, 2018
 * Purpose: CMIS 141 Homework 3- Create a HeadPhone class
 * Outside References: http://www.vogella.com/tutorials/JavaRegularExpressions/article.html
 * Used to figure out how to use Regular Expressions to scan input and help make a loop that
 * prevents Scanner exceptions
 */
public class HeadPhone {
  // Define constant variables
  public static final int LOW = 1;
  public static final int MEDIUM = 2;
  public static final int HIGH = 3;
  // Define instance variables
  private int volume;
  private boolean pluggedIn;
  private String manufacturer;
  private Color headPhoneColor;
  private String headPhoneModel;

  // enum Color is used here to avoid the headache of trying to get RGB colors to display as a word
  // rather than RGB values. It does have the limitation of forcing the user to know what values are
  // usable beforehand.
  enum Color {
    red, green, blue, yellow, black, white, orange, violet, silver, grey, gold
  }



  // Define getter methods
  // Get the Headphone volume
  public int getVolume() {
    return this.volume;
  }

  // Get plugged in status
  public boolean getPluggedIn() {
    return this.pluggedIn;
  }

  // Get Headphone manufacturer
  public String getManufacturer() {
    return this.manufacturer;
  }

  // Get Headphone color
  public Color getHeadPhoneColor() {
    return this.headPhoneColor;
  }

  // Get Headphone model
  public String getHeadPhoneModel() {
    return this.headPhoneModel;
  }

  // Define setter Methods
  // Set the Headphone volume
  public void setVolume(int vol) {
    this.volume = vol;
  }
```

HeadPhone.java Example Continued:

```java
// Set plugged in status
public void setPluggedIn(boolean plug) {
  this.pluggedIn = plug;
}

// Set Headphone manufacturer
public void setManufacturer(String maker) {
  this.manufacturer = maker;
}

// Set Headphone color
public void setHeadPhoneColor(Color color) {
  this.headPhoneColor = color;
}

// Set Headphone model
public void setHeadPhoneModel(String model) {
  this.headPhoneModel = model;
}


// Default constructor
public HeadPhone() {
  volume = MEDIUM;
  pluggedIn = false;
  manufacturer = ("AKG");
  headPhoneColor = Color.black;
  headPhoneModel = ("K240 Studio");
}

// Constructor which allows modification of data when creating the object
public HeadPhone(int volume, boolean pluggedIn, String manufacturer, Color headPhoneColor,
    String headPhoneModel) {
  this.volume = volume;
  this.pluggedIn = pluggedIn;
  this.manufacturer = manufacturer;
  this.headPhoneColor = headPhoneColor;
  this.headPhoneModel = headPhoneModel;


}

// Lists all the HeadPhone getter info in an easily readable format
public String toString() {
  String headPhoneInfo = "(HeadPhone manufacturer = " + this.manufacturer + ", HeadPhone model = "
      + this.headPhoneModel + ", HeadPhone color = " + this.headPhoneColor
      + ", HeadPhone volume = " + this.volume + ", HeadPhone is plugged in = " + this.pluggedIn
      + ")";
  return headPhoneInfo;
}
```

HeadPhone.java Example Continued:

```java
/*
 * A method to allow user input to change the HeadPhone volume. I finished the homework early this
 * week and really wanted to figure out how to handle scanner exceptions. This method is probably
 * the 4th or 5th version I tried. I started with .nextInt() and using a switch to handle the
 * selection but the program couldn't handle non integer inputs. Then I went with a while loop
 * for users to select an integer greater than 0 and less than 4. The problem was that the input
 * errors happened before the program tried to do anything with the data. I played around with
 * try/catch and converting String to int but i couldn't get the method to loop back after the
 * catch. I tried messing with regex to use a boolean return as a kind of switch. I finally
 * settled on accepting String input and using a while loop to prompt input until specified input
 * was used.
 */
    public void changeVolume() {
   // Scanner for user input
   Scanner keyboard = new Scanner(System.in);
   // variable to hold keyboard input
   String setting;
   System.out.println("Select your HeadPhone volume (1 = LOW, 2 = MEDIUM, 3 = HIGH)");
   setting = keyboard.next();
   // while loop keeps prompting user for input until they select 1, 2, or 3
   while (!setting.matches("(1|2|3)")) {
     System.out.println("You have entered invalid input. Please try again.");
     System.out.println("Select your HeadPhone volume (1 = LOW, 2 = MEDIUM, 3 = HIGH)");
     setting = keyboard.next();
   }
   if (setting.equals("1")) {
     // If user inputs '1' volume is defined as LOW and users are prompted of the change
     volume = LOW;
     System.out.println("The volume has been set to the LOW setting.");
   }
   // If user inputs '2' volume is defined as MEDIUM and users are prompted of the change
   else if (setting.equals("2")) {
     volume = MEDIUM;
     System.out.println("The volume has been set to the MEDIUM setting.");
   }
   // If user inputs '3' volume is defined as HIGH and users are prompted of the change
   else if (setting.equals("3")) {
     volume = HIGH;
     System.out.println("The volume has been set to the HIGH setting.");


   }
  }
}
```

TestHeadPhone.java Example:

```java
/*
 * File: TestHeadPhone.java
 * Author: Ben Brandhorst
 * Date: November 27, 2018
 * Purpose: Test HeadPhone class created as part of CMIS 141 Homework 3
 */
public class TestHeadPhone {
  public static void main(String[] args) {
    // Create 3 HeadPhone objects
    // First HeadPhone is created using the default contructor
    HeadPhone headPhone1 = new HeadPhone();
    // The next two HeadPhones are created with different data than default constructor uses
    HeadPhone headPhone2 = new HeadPhone(HeadPhone.LOW, false, "Sony", HeadPhone.Color.violet, "PS4 Gold");
    HeadPhone headPhone3 = new HeadPhone(HeadPhone.HIGH, true, "Dr. Dre", HeadPhone.Color.yellow, "Beats");

    // Verifying the default constructor settings by using the toString method
    System.out.println("Default constructor results of toString for headPhone1: " + headPhone1.toString());
    // Set volume to low
    headPhone1.setVolume(HeadPhone.LOW);
    // Set plugged in status to true
    headPhone1.setPluggedIn(true);
    // Set HeadPhone manufacturer to Bose
    headPhone1.setManufacturer("Bose");
    // Set HeadPhone model to Quiet Comfort 35
    headPhone1.setHeadPhoneModel("Quiet Comfort 35");
    // Set HeadPhone color to white
    headPhone1.setHeadPhoneColor(HeadPhone.Color.white);

    System.out.println("The setter method has been used to change data for headPhone1. New results are below:");
    // Displays new volume setting
    System.out.println("Results of getVolume: " + headPhone1.getVolume());
    // Displays plugged in status
    System.out.println("Results of getPluggedIn: " + headPhone1.getPluggedIn());
    // Display HeadPhone manufacturer
    System.out.println("Results of getManufacturer: " + headPhone1.getManufacturer());
    // Display HeadPhone model
    System.out.println("Results of getHeadPhoneModel: " + headPhone1.getHeadPhoneModel());
    // Display HeadPhone color
    System.out.println("Results of getHeadPhoneColor: " + headPhone1.getHeadPhoneColor());
    // Verify all changes have been reflected in the toString
    System.out.println("toString results after changes: " + headPhone1.toString());
    // Allows user input to change volume
    System.out.println("Using the changeVolume method on headPhone1: ");
    headPhone1.changeVolume();
    System.out.println(
        "Results of getVolume after using changeVolume method: volume = " + headPhone1.getVolume());
    System.out.println("");

    // Verifying the modified constructor settings by using the toString method
    System.out.println("toString results after using modified contructor for headPhone2: " + headPhone2.toString());
    // Set volume to high
    headPhone2.setVolume(HeadPhone.HIGH);
    // Set plugged in status to true
    headPhone2.setPluggedIn(true);
    // Set HeadPhone manufacturer to Sennheiser
    headPhone2.setManufacturer("Sennheiser");
    // Set HeadPhone model to PXC 550
    headPhone2.setHeadPhoneModel("PXC 550");
    // Set HeadPhone color to orange
    headPhone2.setHeadPhoneColor(HeadPhone.Color.orange);
```

TestHeadPhone.java Example Continued:

```
    System.out.println("The setter method has been used to change data for headPhone2. New results are below:");
    // Displays new volume setting
    System.out.println("Results of getVolume: " + headPhone2.getVolume());
    // Displays plugged in status
    System.out.println("Results of getPluggedIn: " + headPhone2.getPluggedIn());
    // Display HeadPhone manufacturer
    System.out.println("Results of getManufacturer: " + headPhone2.getManufacturer());
    // Display HeadPhone model
    System.out.println("Results of getHeadPhoneModel: " + headPhone2.getHeadPhoneModel());
    // Display HeadPhone color
    System.out.println("Results of getHeadPhoneColor: " + headPhone2.getHeadPhoneColor());
    // Verify all changes have been reflected in the toString
    System.out.println("toString results after changes: " + headPhone2.toString());
    // Allows user input to change volume
    System.out.println("Using the changeVolume method on headPhone2: ");
    headPhone2.changeVolume();
    System.out.println(
        "Results of getVolume after using changeVolume method: volume = " + headPhone2.getVolume());
    System.out.println("");

    // Verifying the default constructor settings by using the toString method
    System.out.println("toString results after using modified contructor for headPhone3: " + headPhone3.toString());
    // Set volume to low
    headPhone3.setVolume(HeadPhone.LOW);
    // Set plugged in status to false
    headPhone3.setPluggedIn(false);
    // Set HeadPhone manufacturer to Audio Technica
    headPhone3.setManufacturer("Audio Technica");
    // Set HeadPhone model to ATH-M20x
    headPhone3.setHeadPhoneModel("ATH M20x");
    // Set HeadPhone color to blue
    headPhone3.setHeadPhoneColor(HeadPhone.Color.blue);

    System.out.println("The setter method has been used to change data for headPhone3. New results are below:");
    // Displays new volume setting
    System.out.println("Results of getVolume: " + headPhone3.getVolume());
    // Displays plugged in status
    System.out.println("Results of getPluggedIn: " + headPhone3.getPluggedIn());
    // Display HeadPhone manufacturer
    System.out.println("Results of getManufacturer: " + headPhone3.getManufacturer());
    // Display HeadPhone model
    System.out.println("Results of getHeadPhoneModel: " + headPhone3.getHeadPhoneModel());
    // Display HeadPhone color
    System.out.println("Results of getHeadPhoneColor: " + headPhone3.getHeadPhoneColor());
    // Verify all changes have been reflected in the toString
    System.out.println("toString results after changes: " + headPhone3.toString());
    // Allows user input to change volume
    System.out.println("Using the changeVolume method on headPhone3: ");
    headPhone3.changeVolume();
    System.out.println(
        "Results of getVolume after using changeVolume method: volume = " + headPhone3.getVolume());
    System.out.println("");
  }
}
```