Exam 2 Review WS Answers

1. Assume that carArray is an array that holds Car objects and that Car has a toString method. How would you iterate through the array and print out each object's toString method?

   Answer:

   ```
   for (Car c : carArray) {
       System.out.println(c);
   }
   ```

2. Which method headers would compile and properly overload the following method?

   public int addition(int a, int b) { }

   a) public static void addition(int a, int b) {}

   b) public void addition(int a, int b, int c) {}

   c) public static int addition(int a, int b, int c) {}

   d) public void addition(int c, int d) {}

3. What are valid ways of calculating a random int [10, 65]?
   a. (int) Math.random() * 55 + 10
   b. (int) (Math.random() * 56) + 10
   c. rand.nextInt(56) + 10
   d. rand.nextInt(55) + 10

4.
   ```
   public enum Bender {
       WATERBENDER, EARTHBENDER, FIREBENDER, AIRBENDER, CHIBLOCKER
   }
   ```

   In a Driver class:

   ```
   Bender b1 = Bender.FIREBENDER;
   ```

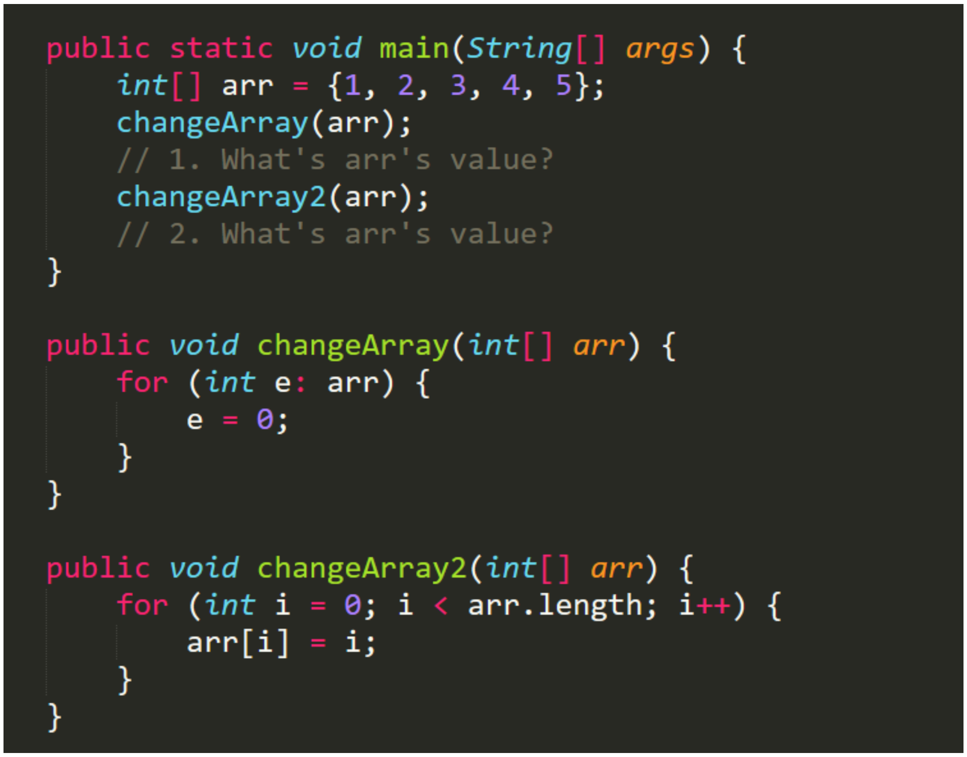
   a) What is `b1.ordinal()`?

   Answer: 2

   b) What is `b1.name()`?

   Answer: "FIREBENDER"

5.



```
public static void main(String[] args) {
    int[] arr = {1, 2, 3, 4, 5};
    changeArray(arr);
    // 1. What's arr's value?
    changeArray2(arr);
    // 2. What's arr's value?
}

public void changeArray(int[] arr) {
    for (int e: arr) {
        e = 0;
    }
}

public void changeArray2(int[] arr) {
    for (int i = 0; i < arr.length; i++) {
        arr[i] = i;
    }
}
```

a. For 1. What’s arr’s value?

   Answer: {1, 2, 3, 4, 5}

b. For 2. What’s arr’s value?

   Answer: {0, 1, 2, 3, 4}

6. What are valid ways of declaring an array?

   a. String[] strArray = {“a”, “b”, “c”};

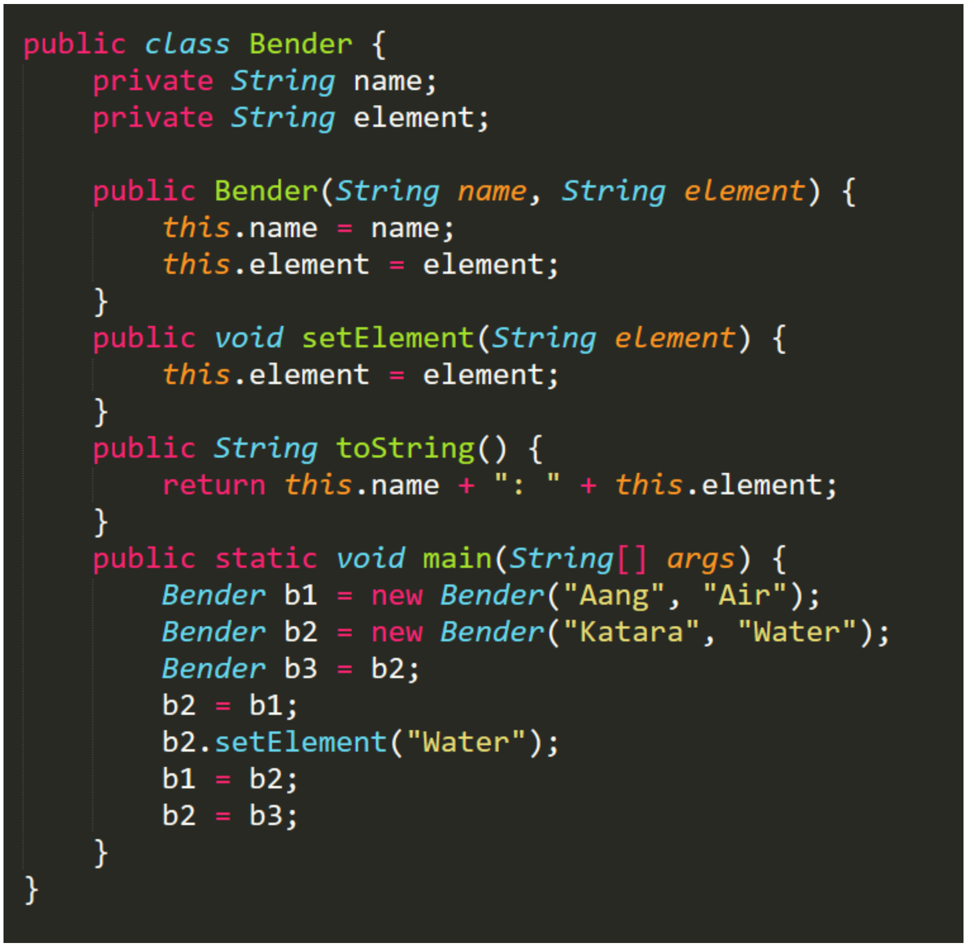
   b. String[] strArray = new String[]{“a”, “b”, “c”};

   c. String[][] strArray = new String[3][1];

   d. String[] strArray = new String[3][ ];

   e. String[] strArray = new String[ ][ ];

7.



```java
public class Bender {
    private String name;
    private String element;

    public Bender(String name, String element) {
        this.name = name;
        this.element = element;
    }
    public void setElement(String element) {
        this.element = element;
    }
    public String toString() {
        return this.name + ": " + this.element;
    }
    public static void main(String[] args) {
        Bender b1 = new Bender("Aang", "Air");
        Bender b2 = new Bender("Katara", "Water");
        Bender b3 = b2;
        b2 = b1;
        b2.setElement("Water");
        b1 = b2;
        b2 = b3;
    }
}
```

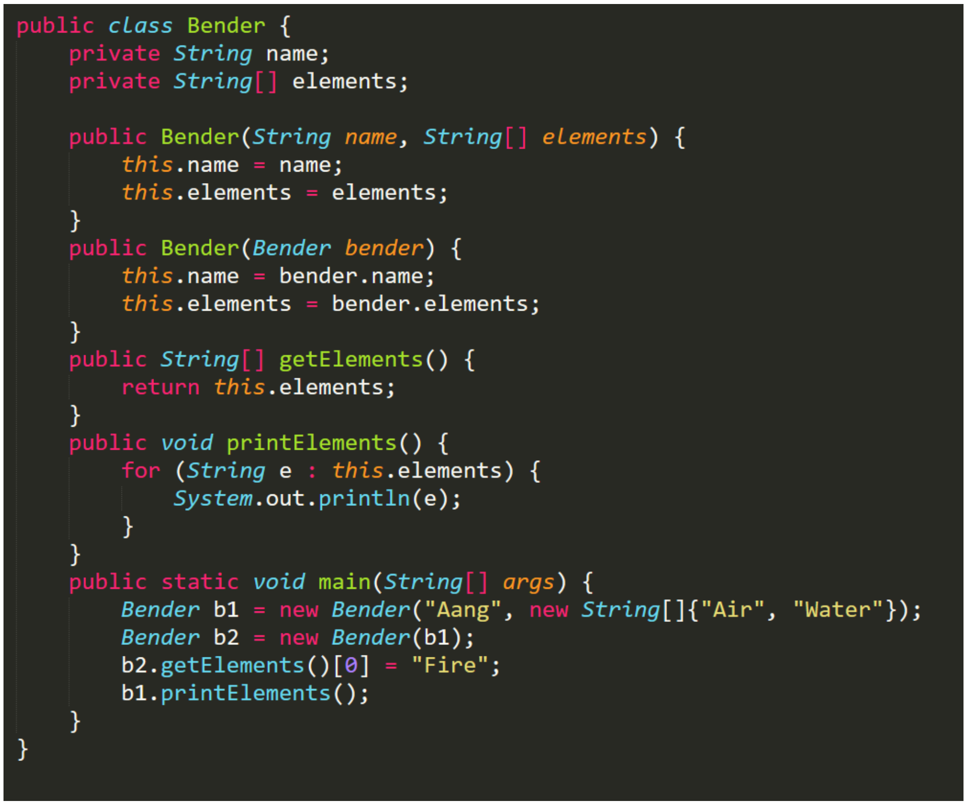
a. After everything in main, what is b1.toString()

Answer: Aang: Water

b. What is b2.toString()?

Answer: Katara: Water

8.



```java
public class Bender {
    private String name;
    private String[] elements;

    public Bender(String name, String[] elements) {
        this.name = name;
        this.elements = elements;
    }
    public Bender(Bender bender) {
        this.name = bender.name;
        this.elements = bender.elements;
    }
    public String[] getElements() {
        return this.elements;
    }
    public void printElements() {
        for (String e : this.elements) {
            System.out.println(e);
        }
    }
    public static void main(String[] args) {
        Bender b1 = new Bender("Aang", new String[]{"Air", "Water"});
        Bender b2 = new Bender(b1);
        b2.getElements()[0] = "Fire";
        b1.printElements();
    }
}
```

a. What gets printed out?

Answer: Fire Water

b. Does the copy constructor make a deep copy or shallow copy?

Answer: Shallow

c. If it is a shallow copy, write a deep copy constructor. If it is a deep copy, write a shallow copy constructor.

Answer:

```java
public Bender(Bender bender) {
    this.name = bender.name;
    this.elements = new
String[bender.elements.length];
    for (int i = 0; i < this.elements.length; i++) {
        this.elements[i] = bender.elements[i];
    }
}
```