1 Boxes and Pointers II

Draw a box and pointer diagram for each code block.

(a) `char a = 'a'; char b = 'a';`  
`b = 'b';`  
`int[] x = {1, 2, 3}; int[] y = x;`  
`y[2] = 7;`

(b) `IntList l = IntList.list(1, 2, 3);`  
`IntList l2 = l;`  
`l.tail.tail.head = 7;`

(c) `IntList[] ll = new IntList[3];`  
`ll[0] = IntList.list(1, 2);`  
`ll[1] = IntList.list(2);`

2 Objects Refresher

Answer the following questions about the Avatar class.

```java
public class Avatar {
    public static String electricity; public String fluid;

    public Avatar(String str1, String str2) {
        Avatar.electricity = str1;
        this.fluid = str2;
    }

    public static void main(String[] args) {
        Avatar foo1 = new Avatar("one ", "two");
        Avatar foo2 = new Avatar("three ", "four");
        System.out.println(foo1.electricity + foo1.fluid);
        foo1.electricity = "I declare ";
        foo1.fluid = "a thumb war";
        System.out.println(foo2.electricity + foo2.fluid);
    }
}
```

(a) Determine what would be printed after executing the main method of class Avatar.

(b) If we made electricity an instance variable and fluid a class variable instead, would this code still compile? If not, identify what would cause the error.
(c) Reverting our changes from part (b) and starting from the original code, will adding the following method to class Avatar cause any errors during compilation or execution? Why or why not?

```java
public static String getFluid() {
    return fluid;
}
```

3  Min/Max

Given an array A, return a 2 element array B where B[0] is the minimum element of A and B[1] is the maximum element of A.

```java
import static java.lang.Math.max; // max(a, b) returns max of a, b
import static java.lang.Math.min; // min(a, b) returns min of a, b

public static int[] minMax(int[] A) {
    int maxVal = Integer.MIN_VALUE; // smallest int in Java
    int minVal = Integer.MAX_VALUE; // largest int in Java
    
    return new int[]{minVal, maxVal};
}
```

4  Reverse

Given an array A, reverse its elements in place (do not create any new arrays; this should be a destructive method).

```java
public static void reverse(int[] A) {
    
    return;
}
```