Your Next Assignment will be in Java:

- Create a text File, “input.txt”

- Give your text file 10 records of the following
  - StringName and intNumber
    - They needn’t be on separate lines, just separated by spaces
  - My input.txt looks like the following:

Otto
100
Fred
220
mUTATANK
123
CEZTRUCTOR
990
ibstantIa
001
Astance
500

- Create a class with three instance variables
  - String: stringName
  - int: intNumber
  - boolean: isPalindrome

- Create an appropriate constructor
  - When you instantiate you will read-in names and numbers as seen
  - You may initially set all isPalindrome variables to false upon instantiation if you wish

- Create appropriate accessors and mutators

- Once you have read in all records **(my test file will have 50 records)** you are to do the following:

  - Check each stringName to see if it is a non-case-sensitive palindrome (Otto is considered a palindrome), and set the Boolean isPalindrome to “true” if is is.
  - You are then to go through and capitalize the first letter of stringName, and set the remaining characters to lowercase.
  - You are then to write this altered data to the console (System.out.println())
    - Note P for Palindrome and NP is not a Palindrome

Otto 100 P
Fred 220 NP
Mutatank 123 NP
Ceztructor 990 NP
Ibstantia 001 NP
Astance 500 NP

Etc…

- Note that we are creating an array of 50 elts of our Class type. This code is provided for you as is the reading-in of strings and integers

Some helpful info:

My main file:

```
import java.io.*;
import java.util.*;
public class september28 {
    public static void readInRecords (String fileName, ToolBox[] x)
    throws FileNotFoundException            {
        //int count = 0; ??
        File fileText = new File(fileName);
        Scanner s = new Scanner(fileText);
        while (s.hasNext() == true ) {
            String word = s.next();
            int num = s.nextInt();
            //call to creator here ??
        }//while
    }//end method
    public static void main(String[] args)
            throws FileNotFoundException{
        //array declaration of type ToolBox
        ToolBox[] records = new ToolBox[50];
        readInRecords ("input.txt", records);
    }//main
}//class
```

My class file:

```
public class ToolBox {
    //these are class instance variables
    private String classText;
    private int classNumber;
    private Boolean isPalindrome;
    //will need a creator for sure here
}//end class
```

As you can see we wrote a method, readInRecords within our *main class,* September 28. Anything that belongs to a class, but an instance of a class is **static**. So, if you chose to write functions in your main program, they must be static.

Mark Breakdown:

- **Part Marks:**
    - **Appropriate files (class, main input file) structure** **5**
    - **Accessors** **5**
    - **Mutators** **5**
    - **Constructors** **5**
    - **File Reading** **5**
    - **Commenting** **5**
    - **Output (upper/lower case formatting)** **5**
- **The darn[1] thing works** **15**

Total: **50 marks**

[1] People sometimes use darn or darned **to emphasize what they are saying**, often when they are annoyed. [informal, emphasis] There's not a darn thing he can do about it. Collins.