String Project "Class Word"

Consider the incomplete Word class defined below. After looking at the code snippets, **read the instructions on the next page** for directions on how to complete each method.

```
public class Word
private String original;
{
	/**
	 * constructs a Word with String value s
	 * @param s is string value of Word
	 */
	public Word(String s)
	{
		original = s;
	}

	/**
	 * reverses letters in original string
	 * @return a string that is a reverse of original
	 */
	public String reverse()
	{
		// code goes here
	}
	/**
	 * returns the length of the string
	*/
	public Int getLength()
	{
		// code goes here
	}

	/**
	 * determines is word is a palindrome
	 * @return true if word is a palindrome, false otherwise
	 */
	public boolean isPalindrome()
	{
		// code goes here
	}

	/**
	 * Alternate method to determine if word is a palindrome
	 * @return true if word is a palindrome, false otherwise
	 */
	public boolean isPalindrome2()
	{

	}
   /**
	 * removes vowels in original string
	 * @return a string that removes all of the vowels
	 */
	public String removeVowels()
	{
		// code goes here
	}
```

```
/**
        * creates an anagram
        * @return a string that is a random anagram of the original word
        */
       public String anagram()
       {
             // code goes here
       }



}
```

INSTRUCTIONS FOR IMPLEMENTATION OF THE METHODS ABOVE

a. Add the `Word` method `reverse`. `reverse` will return a `String` that is the reverse of `original`.

b. Add the `Word` method `getLength` that will return the length of the string

c. Add the `Word` method `isPalindrome`. `isPalindrome` will determine if original is a palindrome (reads the same forwards and backwards). This version of `isPalindrome` will call `reverse` and compare the reversed original string with the original string and return `true` if the two strings are equal in value, `false` otherwise.

d. Add the `Word` method `isPalindrome2`. `isPalindrome2` will determine if original is a palindrome (reads the same forwards and backwards). This version of `isPalindrome` will check pairs of letters. Each pair will consist of a letter from the beginning of the string and the corresponding place letter from the end of the string.

L E V E L

The method `isPalindrome2` will return `true` if each pair has identical letters, `false` otherwise. Be efficient in writing this algorithm.

e. Add the `Word` method `removeVowels` that will return a string with the vowels removed.

f. Add the `Word` method `anagram` that will return a string with the letters in a random different order (not reverse).