Making the Word Class (4Pic1Word)

Create a new class. Answer is the word that is on the screen.

```
public class Word
{
    private String answer;
    private String possible[] = {"reflect", "network", "crumble"};

}
```

* Add more pictures and answers to the possible array. Here is a great site to help with that: https://4pics1word.ws/7-letter-words/

Add these methods inside the above class. Make sure they are commented!!

The constructor:

```
public Word ()
{
    int pos = (int) (Math.random () * possible.length);
    answer = possible [pos];
}
```

The accessor:

```
public String getWord ()
{
    return answer;
}
```

A mutator/accessor/reset:

```
public String getNewWord ()
{
    int pos = (int) (Math.random () * possible.length);
    answer = possible [pos];
    return answer;
}
```

A facilitator to generate a new clue is on the next page.

It has these steps:

- The clue has the initial word (say, “network”).
- Then, random letters are added to make up ten characters (say “network” + “gjs”)
- Then, the word is scrambled (“networkgjs” becomes “tsekgrjnow”.

```
public String getClue ()
{
    String alpha = "abcdefghijklmnompqrstuvwxyz";
    String clue = answer;
    //add random letters from the alphabet to make up 10 letters
    for (int i = answer.length () ; i < 10 ; i++)
    {
        int pos = (int) (Math.random () * alpha.length ());
        clue += alpha.charAt (pos);
    }

    //splice the clue 40 times to scramble it
    for (int i = 0 ; i < 40 ; i++)
    {
        int pos = (int) (Math.random () * clue.length ());
        int pos2 = (int) (Math.random () * clue.length ());
        //make sure that pos is less than pos2
        if (pos > pos2)
        {
            int temp = pos;
            pos = pos2;
            pos2 = temp;
        }
        //muddle up three sections of the word
        String first = clue.substring (0, pos);
        String mid = clue.substring (pos, pos2);
        String last = clue.substring (pos2, clue.length ());
        clue = last + mid + first;
    }
    //return the clue, muddled up
    return clue;
}
```

Add in these methods too:

1. Equals
2. CompareTo
3. toString
4. custom Constructor
5. setWord
6. getPictureName

Comment the class fully to indicate your understanding.

Possible changes and additions:

- Editing the alphabet to only add common letters is a great idea.
- A really great addition is to edit the clue method so that it never adds in letters that are already in the word. This is quite tricky.