PLANNING: the program helps users to figure out if the word can read backwards as the same word(Palindrome)

```
public class Palindrome1 {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
```

- Created a new scanner

```
// Prompt the user to enter a word or phrase
System.out.print("Enter a word or phrase, and this program will identify if it's a palindrome: ");
String input = scanner.nextLine();
```

- Ask the user to enter a word
- Then that word is stored in a variable called input by using string

```
if (Palindrome.isPalindrome(input)) {
    System.out.println("The word is a palindrome.");
} else {
    System.out.println("The word is not a palindrome.");
}
```

- If the word fits the requirement of isPAlindrome method then it continues to say the word is a palindrome if not then it says it isn't.

```
public static boolean isPalindrome(String string) {
```

- Created a method

```
 string = string.replaceAll(" ", "").toLowerCase();
```

- Remove non-alphabetic characters and convert to lowercase

```
int LEFT1 = 0;
int RIGHT2 = string.length() - 1;
```

- The variable called left is for reading the word regularly by setting it to) to store the user word properly.
- Right2 is for reading it by sorting it backwards

```
  while (LEFT1 < RIGHT2) {
```

- While loop is the regular word is smaller than the backward word

```
if (string.charAt(LEFT1) != string.charAt(RIGHT2)) {

    return false;  // if not a palindrome than the program ends here
}
```

- Then the variable is turned into a char for the program to read to see if both matches
- If not then it returns false.

```
LEFT1++;
RIGHT2--;
```

- To distinguish between left and right bc the string integer is the same for both.
- As the ++ and - - are used to read the string differently as - - reads it from left to right and the opposite for the ++.

```
return true;
```

- Give the sign that when the word complets are the requirements the code comes to use as an output.