ASSESSED EXERCISE 3: Return of Word Game

For this exercise you will create a **wordGame.**

You do not need to have any prior knowledge of the game. Everything that you will need to know to complete this practical is explained below and example runs of the solution are provided at the end of this document.

Rules of the gameplay:

In this game the player is the user who enters a word at a time when prompted by the computer. The computer decides whether the player can continue or the game could be terminated. The following four rules must be maintained at any time of the game:

1. The **first** character of the next word should start with the **last** character of the previous word and they are case sensitive (for example "`Apple`" and "`apple`" are different). The game must be terminated if this rule is violated with a message "`Game over`".
2. The word **must not** be entered in the game before. The game must be terminated if any word is repeated with a message "`Game over`".
3. The word should have **only** the alphanumeric characters (such as [`A-Z`] and [`a` to `z`]), and should **not** have any numeric character (such as numbers `4` or `7`) or space or hyphenated (such as co-worker) or special characters (such as `;` `,` `:`) within itself. For an invalid word, the game should continue with a message "`Entered word is invalid, please try again`".
4. If nothing is entered by the user (empty string or word with length 0), the game must **restart** from the beginning.

To achieve the rules of the above gameplay, write a `wordGame` class with:

A. Two instance variables/array:
1. `words`: An array of words listed while playing the game.
2. `outcome`: A Boolean variable which is true when game is over.

B. Two instance methods
1. `gameStart:` this should check all the necessary conditions inside.
2. `gameRestart:` should initialize the array again and start looping again.

Additionally, write an `application` to test out the above functionalities, display all the previously entered words, prompt the user to enter a new word and play the game. This application should have some default word to start with, such as "`lamp`" and

should prompt the user to enter a word for playing this game. Sensible messages should be displayed to the user as information to the user for taking appropriate action(s).

Please submit your completed project (as a zip/rar file of the entire NetBeans project folder) for assessment. Ask for help if you are unsure about how to compress your project folder.

Marking scheme (total 100%)

Description	Mark
Implementation of the class with instance variables/arrays and methods.	20%
Implementation of the four rules (each 10%, totalling to 40%).	40%
Implementation of application with user input and display of existing words.	20%
Implementation of correct display with relevant messages.	10%
Game termination with appropriate reporting/display of messages.	10%

Example runs

Example 1:

```
Word Game started with lamp
lamp
Enter a word (caseSensitive) to play the game, such as
pen->net->top...: pen
lamp pen
Enter a word (caseSensitive) to play the game, such as
pen->net->top...: net
lamp pen  net
Enter a word (caseSensitive) to play the game, such as
pen->net->top...: top
lamp pen  net  top
Enter a word (caseSensitive) to play the game, such as
pen->net->top...: pencil
lamp pen  net  top  pencil
Enter a word (caseSensitive) to play the game, such as
pen->net->top...: london
lamp pen  net  top  pencil    london
Enter a word (caseSensitive) to play the game, such as
pen->net->top...: net
Game over
```

In the above example, the game was terminated because the user entered a word which was already existing in the list (such as 'net').

Example 2:

```
Word Game started with lamp
```

```
lamp
Enter a word (caseSensitive) to play the game, such as
pen->net->top...: pen
lamp pen
Enter a word (caseSensitive) to play the game, such as
pen->net->top...: netball
lamp pen  netball
Enter a word (caseSensitive) to play the game, such as
pen->net->top...:
Game restarting as the entered word was empty...Word game
restarted with lamp
lamp
Enter a word (caseSensitive) to play the game, such as
pen->net->top...: pencil
lamp pencil
Enter a word (caseSensitive) to play the game, such as
pen->net->top...: london
lamp pencil    london
Enter a word (caseSensitive) to play the game, such as
pen->net->top...: net
lamp pencil    london    net
Enter a word (caseSensitive) to play the game, such as
pen->net->top...: tip
lamp pencil    london    net  tip
Enter a word (caseSensitive) to play the game, such as
pen->net->top...: pencil
Game over
```

In the above example, the game restarted in between because the user entered an empty word (such as nothing was entered when prompted).

Example 3:

```
Word Game started with lamp
lamp
Enter a word (caseSensitive) to play the game, such as
pen->net->top...: paper
lamp paper
Enter a word (caseSensitive) to play the game, such as
pen->net->top...: red
lamp paper     red
Enter a word (caseSensitive) to play the game, such as
pen->net->top...: draft2
Entered word is invalid, please try again
lamp paper     red
Enter a word (caseSensitive) to play the game, such as
pen->net->top...: dim;
Entered word is invalid, please try again
lamp paper     red
```

```
Enter a word (caseSensitive) to play the game, such as
pen->net->top...: dim
lamp paper     red  dim
Enter a word (caseSensitive) to play the game, such as
pen->net->top...: monday:
Entered word is invalid, please try again
lamp paper     red  dim
Enter a word (caseSensitive) to play the game, such as
pen->net->top...: mon-day
Entered word is invalid, please try again
lamp paper     red  dim
Enter a word (caseSensitive) to play the game, such as
pen->net->top...: mon day
Entered word is invalid, please try again
lamp paper     red  dim
Enter a word (caseSensitive) to play the game, such as
pen->net->top...: monday
lamp paper     red  dim  monday
Enter a word (caseSensitive) to play the game, such as
pen->net->top...: yellow
lamp paper     red  dim  monday    yellow
Enter a word (caseSensitive) to play the game, such as
pen->net->top...: weed
lamp paper     red  dim  monday    yellow    weed
Enter a word (caseSensitive) to play the game, such as
pen->net->top...: dim
Game over
```

In the above example, the user is informed that word is invalid and prompted to enter another word. Several cases are tried such as '`dim;`', '`monday:`', '`mon-day`' and '`mon day`'. Kindly note that the examples shown here are not exhaustive and do not cover all the cases.