**SuperString.java (Main File)**

/\*\* Application Purpose: This file has two methods which perform different functions relating to string manipulation.

\* Author: Alex Dorodko

\* Date: 28/NOV/2020

\* Time: 010:00 PM

\*/

import java.util.Random;

import java.util.Scanner;

public class SuperString

{

public static String MethodOne()

{

//This is the state, including the list.

String joint = "";

int random;

String[] words = {"Car", "Art", "Bag", "Six", "Fox", "Acid", "Ball", "Lost", "Rose", "Love"};

//Picking the 5 random words and placing them together.

for (int i = 0; i < 5; i++)

{

random = (int)(Math.random() \* (9 - 0 + 1) + 0);

joint = joint + words[random];

}

//Printing out the words that are in the string.

for (int i = 0; i < 10; i++)

{

if (joint.contains(words[i]) == true)

{

System.out.println(words[i]);

}

}

//Outputting what the string was.

return ("\nThe string was: " + joint);

}

public static String MethodTwo(String characters, int number, char letter)

{

Scanner sc = new Scanner(System.in);

while (number > characters.length())

{

System.out.println("The number entered must be lower than the ammount of letters in the string. Please enter a different number value: ");

number = sc.nextInt();

}

//Creating the array which stores the sub strings

DataStorage[] substrings = new DataStorage[characters.length()];

//Filling the array with default data.

for (int i = 0; i < substrings.length; i++)

{

substrings[i] = new DataStorage("EMPTY");

}

//This is a variable which tracks the array indexes.

int arrayPlacement = 0;

//Here we go through the user's string, and if the letter of the index matches the

//letter the user specified, it creates a substring by taking the next ammount of characters after the letter, as specified by the user.

for (int i = 0; i < characters.length() - number; i++)

{

if (characters.charAt(i) == letter)

{

substrings[arrayPlacement].setSubstring(characters.substring(i, i + number));

arrayPlacement++;

}

}

//Outputting each induvidual substring.

for (int i = 0; i < substrings.length; i++)

{

if (substrings[i].getSubstring() != "EMPTY")

{

System.out.println(substrings[i].getSubstring());

}

}

//Outputting the characters which the user inputted.

return ("\nThe string was: " + characters);

}

}

**DataStorage.java (Storage File)**

/\*\* Application Purpose: This class serves as a storage location for a different method.

\* Author: Alex Dorodko

\* Date: 29/NOV/2020

\* Time: 02:00 PM

\*/

public class DataStorage

{

//Initiating the varible which stores a substring

private String substr = "";

//Making a simple constructor

public DataStorage(String substr)

{

this.substr = substr;

}

//Making a setter and getting to set and retrieve data.

public void setSubstring(String substr)

{

this.substr = substr;

}

public String getSubstring()

{

return substr;

}

}

**MethodsTestHarness.java**

/\*\* Application Purpose: This file is a test harness to test out both methods.

\* Author: Alex Dorodko

\* Date: 28/NOV/2020

\* Time: 03:00 PM

\*/

public class MethodsTestHarness

{

//The first method which outputs the words from a joint string.

public static void main(String[] args)

{

//Printing out the first method.

System.out.println("The words of method one are:\n");

System.out.println(SuperString.MethodOne());

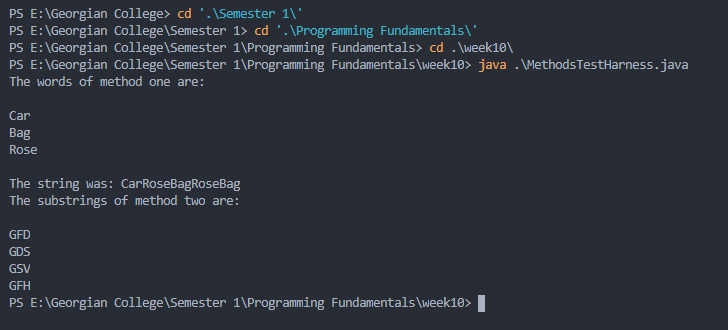
System.out.println("The substrings of method two are:\n");

SuperString.MethodTwo("GFDSGDSFGSVBNGFHG", 3, 'G');

}

}

**Output**

****