GIFT School of Engineering and Applied Sciences

Fall 2020

CS-124: Introduction to Programming - Lab

Lab-14 Manual Solutions

Methods and Strings

Reference: Sections 2.9 and 9.3 (Textbook)

v1.1
18/11/2020

Task #1: Using Strings and Methods

In this task, you are being asked to write methods that manipulate String objects in Java.

Write a method that returns the count of the number of space characters that appear in the String object **str**.

You may use the following header for this method:

```
public static int countSpaceCharacter(String str)
```

NOTE: You may use any class **String** methods.

HINTS:

- You may use a static method **`Character.isWhitespace(char c)`** that returns **true** if **c** is a whitespace.
- You may use the String **`charAt(index)`** method to get a character at an **index** from a String object.
- You may use the String **`length()`** method to get number of character in a String object.
- Use both methods in a loop to count all space characters.

1. Create a program called **StringMethodsLab14.java** and add this method in this class.
2. Use a **Scanner** object for the String arguments input.
3. Call the method with appropriate String arguments.
4. Display appropriate messages based on the returned values.

```
import java.util.Scanner;

public class StringMethodsLab14A{

   public static int countSpaceCharacter(String str){
        int strLength = str.length();
        int count = 0;

        for (int i = 0; i < strLength; ++i){
             char aCharacter = str.charAt(i);
             if (Character.isWhitespace(aCharacter)){
                  ++count;
             }//if
        }//for

        return count;

   }//countSpaceCharacter()

   public static void main (String [] args){
        Scanner input = new Scanner(System.in);

        System.out.print("Enter a string: ");
        String word = input.nextLine();

        int count = countSpaceCharacter(word);
        System.out.println(count + " spaces.");
   }//main
}//class
```

Task #2: Using Strings and Methods

In this task, you are being asked to write methods that manipulate String objects in Java.

Write a method that returns the count of the number of digits that appear in the String object **str**. A digit can be any one from **0** to **9.**

You may use the following header for this method:

```
public static int countDigits(String str)
```

NOTE: You may use any class **String** methods.

HINTS:

- You may use a static method **`Character.isDigit(char c)`** that returns **true** if **c** is a digit.
- You may use the String **`charAt(index)`** method to get a character at an **index** from a String object.
- You may use the String **`length()`** method to get number of character in a String object.
- Use both methods in a loop to count all digits.

1. Add this method to the **StringMethodsLab14.java** class.
2. Use a **Scanner** object for the String arguments input.
3. Call the method with appropriate String arguments.
4. Display appropriate messages based on the returned values.

```
import java.util.Scanner;

public class StringMethodsLab14B{

     public static int countDigits(String str){
          int strLength = str.length();
          int count = 0;

          for (int i = 0; i < strLength; ++i){
               char aCharacter = str.charAt(i);
               if (Character.isDigit(aCharacter)){
                    ++count;
               }//if
          }//for

          return count;

     }//countDigits()

     public static void main (String [] args){
          Scanner input = new Scanner(System.in);

          System.out.print("Enter a string: ");
          String word = input.nextLine();

          int count = countDigits(word);
          System.out.println(count + " digits.");
     }//main
}//class
```

Task #3: Using Strings and Methods

In this task, you are being asked to write methods that manipulate String objects in Java.

Write a method that returns the count of the number of lowercase characters that appear in the String object **str**. A lowercase character can be any one from **a** to **z.**

You may use the following header for this method:

```
public static int countLowercaseCharacters(String str)
```

NOTE: You may use any class **String** methods.

HINTS:

- You may use a static method **`Character.isLowerCase(char c)`** that returns **true** if **c** is a lowercase character.
- You may use the String **`charAt(index)`** method to get a character at an **index** from a String object.
- You may use the String **`length()`** method to get number of character in a String object.
- Use both methods in a loop to count all digits.

1. Add this method to the **StringMethodsLab14.java** class.
2. Use a **Scanner** object for the String arguments input.
3. Call the method with appropriate String arguments.
4. Display appropriate messages based on the returned values.

```
import java.util.Scanner;

public class StringMethodsLab14C{

   public static int countLowercaseCharacters(String str){
        int strLength = str.length();
        int count = 0;

        for (int i = 0; i < strLength; ++i){
             char aCharacter = str.charAt(i);
             if (Character.isLowerCase(aCharacter)){
                  ++count;
             }//if
        }//for

        return count;

   }//countLowercaseCharacters()

   public static void main (String [] args){
        Scanner input = new Scanner(System.in);

        System.out.print("Enter a string: ");
        String word = input.nextLine();

        int count = countLowercaseCharacters(word);
        System.out.println(count + " lower case characters.");

   }//main
}//class
```

Task #4: Using Strings and Methods

In this task, you are being asked to write methods that manipulate String objects in Java.

Write a method that searches for a target string in a given source String argument. If the target string is found, it returns the index of the starting character of the target string. If the target string is not found, returns **-1.**

You may use the following header for this method:

```
public static int searchString(String source, String target)
```

For example:

```
int index = searchString("Education", "cat");
```

searches for the string **cat** from the string **Education.**

NOTE: You may use any class **String** methods.

1. Add this method to the **StringMethodsLab14.java** class.
2. Use a **Scanner** object for the both String arguments input.
3. Call the method with appropriate String arguments.
4. Display appropriate messages based on the returned values.

```
import java.util.Scanner;

public class StringMethodsLab14D{

   public static int searchString(String source, String
target){
        return source.indexOf(target);
   }//searchString()

   public static void main (String [] args){
        Scanner input = new Scanner(System.in);

        System.out.print("Enter the source string: ");
        String source = input.nextLine();
        System.out.print("Enter the target string: ");
        String target = input.nextLine();
```

```
        int index = searchString(source, target);

        if (index < 0){
             System.out.println("The string \"" + target +"\"
was not found in \"" + source + "\"");
        }
        else{
             System.out.println("The index of \"" + target
+"\" in \"" + source + "\" is " + index);
        }//if

   }//main
}//class
```

Task #5: Using Strings and Methods

In this task, you are being asked to write methods that manipulate String objects in Java.

Write a method that accepts a String object as an argument and displays its contents backward. For instance, if the string argument is **"gravity"** the method should display **"ytivarg"**. Demonstrate the method in a program that asks the user to input a string and then passes it to the method.

You may use the following header for this method:

```
public static void displayBackwards(String str)
```

NOTE: You may use any class **String** methods.

1. Create a program called **StringBackwardsLab14.java** class.
2. Use a **Scanner** object for the String arguments input.
3. Call the method with appropriate String arguments.
4. Display appropriate messages.

```
import java.util.Scanner;

public class StringBackwardsLab14{

   public static void displayBackwards(String str){
        int strLength = str.length();

        for (int i = strLength - 1; i >= 0; --i){
             System.out.print(str.charAt(i));
        }//for
   }//displayBackwards()

   public static void main (String [] args){
        Scanner input = new Scanner(System.in);
        System.out.print("Enter a string: ");
        String word = input.nextLine();

        displayBackwards(word);

   }//main
}//class
```

Task #6: Using Strings and Methods

In this task, you are being asked to write methods that manipulate String objects in Java.

Write a method that accepts a String object as an argument and returns the number of words it contains. For instance, if the argument is **"Four score and seven years ago"** the method should return the number **6**.

Demonstrate the method in a program that asks the user to input a string and then passes it to the method. The number of words in the string should be displayed on the screen.

You may use the following header for this method:

```
public static int countNumberOfWords(String str)
```

NOTE: You may use any class **String** methods.

1. Create a program called **StringWordsCountLab14.java** class.
2. Use a **Scanner** object for the String arguments input.
3. Call the method with appropriate String arguments.
4. Display appropriate messages based on the returned values.

```
import java.util.Scanner;

public class StringWordsCountLab14{

   public static int countNumberOfWords(String str){
        int strLength = str.length();
        int count = 1;

        for (int i = 0; i < strLength; ++i){
             char aCharacter = str.charAt(i);
             if (Character.isWhitespace(aCharacter)){
                  ++count;
             }//if
        }//for

        return count;

   }//countNumberOfWords()
```

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

         System.out.print("Enter a string: ");
         String word = input.nextLine();

         int count = countNumberOfWords(word);
         System.out.println(count + " words.");
    }//main
}//class
```

Task #7: Using Strings and Methods

In this task, you are being asked to write methods that manipulate String objects in Java.

Write a method that accepts a String object as an argument and returns the number of vowels it contains. A vowel is any one of **a, e, i, o, u.**

Demonstrate the method in a program that asks the user to input a string and then passes it to the method. The number of vowels in the string should be displayed on the screen.

You may use the following header for this method:

```
public static int countNumberOfVowels(String str)
```

NOTE: You may use any class **String** methods.

1. Create a program called **StringVowelsCountLab14.java** class.
2. Use a **Scanner** object for the String arguments input.
3. Call the method with appropriate String arguments.
4. Display appropriate messages based on the returned values.

```
import java.util.Scanner;

public class StringVowelsCountLab14{

   public static int countNumberOfVowels(String word){
        int wordLength = word.length();
        word = word.toLowerCase();
        int count = 0;
        for (int i = 0; i < wordLength; ++i){
             int character = word.charAt(i);
             if (character == 'a' || character == 'e' ||
character == 'i' || character == 'o' || character == 'u' ||
character == 'y'){
                  ++count;
             }//if
        }//for

        return count;

   }//countNumberOfVowels()
```

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

         System.out.print("Enter a string: ");
         String word = input.nextLine();
         int count = countNumberOfVowels(word);
         System.out.println(count + " vowels.");
    }//main
}//class
```

Task #8: Using Strings and Methods

In this task, you are being asked to write methods that manipulate String objects in Java.

Imagine you are developing a software package for Amazon.com that requires users to enter their own passwords. Your software requires that users' passwords meet the following criteria:

- The password should be at least six characters long.
- The password should contain at least one uppercase and at least one lowercase letter.
- The password should have at least one digit.

Write a method that verifies that a password meets the stated criteria. Demonstrate the method in a program that allows the user to enter a password and then displays a message indicating whether it is valid or not.

You may use the following header for this method:

```
public static void verifyPassword(String password)
```

For example:

```
verifyPassword("HelloWorld123");
```

should display:

```
Password length: 13
Uppercase letters: 2
Lowercase letters: 8
Digits: 3

Password Verified.
```

If the password string does not meet any one of the above criteria, the method should display:

```
Password Not Verified!
```

NOTE: You may use any class **String** methods.

HINTS:

- You may use a static method **`Character.isLowerCase(char c)`** that returns **true** if **c** is a lowercase character. Similarly, you may use **`Character.isUpperCase(char c)`** that returns **true** if **c** is an uppercase character.

1. Create a program called **StringPasswordVerifierLab14.java** class.
2. Use a **Scanner** object for the String arguments input.
3. Call the method with appropriate String arguments, both for valid and invalid passwords.
4. Display appropriate messages.

```
import java.util.Scanner;

public class StringPasswordVerifierLab14{

   public static void verifyPassword(String password){
        int passwordLength = password.length();
        int lowerCaseCount = 0;
        int upperCaseCount = 0;
        int digitCount = 0;

        for (int i = 0; i < passwordLength; ++i){
             char aCharacter = password.charAt(i);
             if (Character.isLowerCase(aCharacter)){
                  ++lowerCaseCount;
             }
             else if (Character.isUpperCase(aCharacter)){
                  ++upperCaseCount;
             }
             else if (Character.isDigit(aCharacter)){
                  ++digitCount;
             }//if
        }//for

        if (passwordLength >=6 && lowerCaseCount >=1 &&
upperCaseCount >=1 && digitCount >=1){
             System.out.println("Password length: " +
passwordLength);
             System.out.println("Uppercase letters: " +
upperCaseCount);
             System.out.println("Lowercase letters: " +
lowerCaseCount);
             System.out.println("Digits: " + digitCount);

             System.out.println("\nPassword Verified.");
        }
```

```
            else{
                  System.out.println("Password Not Verified.");
            }//if

      }//verifyPassword()

      public static void main (String [] args){
            Scanner input = new Scanner(System.in);

            System.out.print("Enter the password: ");
            String password = input.next();

            verifyPassword(password);

      }//main
}//class
```

Task #9: Using Strings and Methods

In this task, you are being asked to write methods that manipulate String objects in Java.

Write a program that asks the user to enter a series of single digit numbers with nothing separating them. The program should display the sum of all the single digit numbers in the string. For example, if the user enters **2514**, the method should return **12**, which is the sum of **2, 5, 1,** and **4**.

You may use the following header for this method:

```
public static int sumStringDigits(String str)
```

NOTE: You may use any class **String** methods.

HINTS:

- You may use a static method **`Character.getNumericValue(char c)`** that returns the **int** value of the character **c**.

1. Create a program called **StringDigitsSumLab14.java** class.
2. Use a **Scanner** object for the String arguments input. Make sure to input only digits as the argument.
3. Call the method with appropriate String arguments.
4. Display appropriate messages.

```
import java.util.Scanner;

public class StringDigitsSumLab14{

   public static int sumStringDigits(String str){
        int strLength = str.length();
        int sum = 0;

        for (int i = 0; i < strLength; ++i){
             char aCharacter = str.charAt(i);
             sum += Character.getNumericValue(aCharacter);
        }//for

        return sum;

   }//sumStringDigits()
```

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

        System.out.print("Enter a string: ");
        String word = input.nextLine();

        int sum = sumStringDigits(word);
        System.out.println("The sum of digits in the string
\"" + word + "\" is " + sum);
   }//main
}//class
```