

Week 13-1 : String

Part1. String

- Generate a String

Generate a String as a literal : `String s = "Hello";`

Generate a String as a object : `String t = new String("Hello");`

Not possibly modifying

- Method

<code>equals()</code>	Comparing each strings, <code>==(x)</code>
<code>compareTo(String str)</code>	Comparing each strings based on an order of the dictionary. if each is same, return a 0. if a string object precedes a str, return a negative. If a string object follows a str, return a positive.
<code>String concat(String str)</code>	Returning a string adding a str.
<code>String trim()</code>	Removing the spaces around a string, an return it.
<code>char charAt(int index)</code>	Returning a char value of index.
<code>int length()</code>	Returning a size of string.
<code>String toLowerCase()</code>	Returning a string changed to lowercase
<code>String toUpperCase()</code>	Returning a string changed to uppercase
<code>int indexOf(int ch)</code>	Returning a index of containing a ch. if a string do not contain a ch, then return -1
<code>String subString(int beginIndex)</code>	Returning a sub string placed after beginIndex
<code>boolean contains(CharSequence s)</code>	Return a true value, if a string

contains(CharSequence s)	object contains a CharSequence
--------------------------	--------------------------------

```

public class String {
    public static void main(String[] args) {
        String a = new String(" Computer ");
        String b = new String("Programming");

        a = a.concat(b);
        System.out.println(a);

        a = a.trim();
        System.out.println(a);
    }
}

```

☞ Result : Computer Programming / ComputerProgramming

Part2. StringBuffer

Possibly change a string value after generate it.

manipulating method : append, insert

A size of a StringBuffer can be different about a length of a string.

Constructor	Description
StringBuffer()	Generating a buffer that size is 16. and this buffer do not contain a character.
StringBuffer(charSequence seq)	Generating a buffer that include characters defined by seq
StringBuffer(int capacity)	Generating a buffer which include size of capacity
StringBuffer(String str)	Generating a buffer that initialized by str

Method	Description
StringBuffer	Appending a str after a stringbuffer.

append(String str)	
StringBuffer append(StringBuffer sb)	Appending a stringbuffer after a stringbuffer.
int capacity()	Returning a current size of the stringbuffer.
StringBuffer delete(int start, int end)	Removing a sub string from the stringbuffer.
StringBuffer insert(int offset, String str)	Inserting a str at offset.
StringBuffer replace(int start, int end, String str)	Replacing a sub string to the str.
StringBuffer reverse()	Changing a stringbuffer reversely.
void setLength()	Setting a length of string saved in the stringbuffer.

- Example

```
public class UsingStringBuilder {
    public static Random rand = new Random(47);
    public String toString() {
        StringBuilder result = new StringBuilder("");
        for(int i = 0; i < 25; i++) {
            result.append(rand.nextInt(100));
            result.append(", ");
        }
        result.delete(result.length()-2, result.length());
        result.append("]");
        return result.toString();
    }
    public static void main(String[] args) {
        UsingStringBuilder usb = new UsingStringBuilder();
        System.out.println(usb);
    }
}
```

☞ Appending a random number and ',' using a Random class. after 25, using a delete for removing a final ',' and appending a ']'.
'].

Part3. String Tokenizer

Class for splitting a string.

```
import java.util.StringTokenizer;
public class StringTokenizerEx {
    public static void main(String[] args) {
        StringTokenizer st = new
StringTokenizer("Seoul-Incheon-Busan-Daejeon-
Dague", "-");
        while (st.hasMoreTokens())
            System.out.println(st.nextToken());
    }
}
```

☞ Result : Seoul / Incheon / Busan / Daejeon / Daegu
Splitting a string based on a '-' delimiter and printing out those splitted strings.

- Tokenizer method

StringTokenizer(String str, String delim)	Generating a string tokenizer initialized by a str and a delim
int countTokens()	Returning a number of tokens remaining on a string
boolean hasMoreTokens()	Return a true value, When tokens remain.
String nextToken()	Returning a next token

[Exercise]

1. Make a MyString class and manipulate string variables using several methods(first page). and save the output strings on a new file "string.txt".

```
String s1 = new string(" hello");  
String s2 = new string(",everyone");
```

you print out final value using string class methods.

```
hello,everyone  
hello,everyone  
hi,everyone  
[1] hi  
[2] everyone  
i,everyone  
e
```

2. Attach the WeeklyCalendar Code file. Then, write a code about complete 'loadWeekPlan' method which are located in HCalendar class using string tokenizer.