Strings, StringBuilder, StringBuffer

Mindaugas Karpinskas
2017

String

- Strings in java are immutable
- Once created they cannot be altered and hence any alterations will lead to creation of new string object

Example

- String s1 = “Example”
- String s2 = new String(“Example”)
- String s3 = “Example”
- The difference between the three statements is that, s1 and s3 are pointing to the same memory location i.e. the string pool. s2 is pointing to a memory location on the heap.
- Using a new operator creates a memory location on the heap.
- Concatinting s1 and s3 leads to creation of a new string in the pool.

***StringBuffer

- StringBuffer is a synchronized and allows us to mutate the string.
- StringBuffer has many utility methods to manipulate the string.
- This is more useful when using in a multithreaded environment.
- Always has a locking overhead.

Example

```
public class mybuffers{
    public static void main(String args[]){
        StringBuffer buffer =  new
   StringBuffer(“Hi”);
        buffer.append(“Bye”);
        System.out.println(buffer);
    }
}
```

- This program appends the string Bye to Hi and prints it to the screen.

Class methods

- **StringBuilder append(boolean b)**
- **StringBuilder append(char c)**
- **StringBuilder append(char[] str)**
- …
-
- **StringBuilder deleteCharAt(int index)**
- **StringBuilder delete(int start, int end)**
-

U1

Užpildykime StringBuffer ir išspausdinti skaičių seką atskirta kableliais, pasinaudokime for (int i=1;i<=N;i++)

1, 2, 3, 4, 5, 6…

- StringBuffer

StringBuilder

- StringBuilder is the same as the StringBuffer class
- The StringBuilder class is not synchronized and hence in a single threaded environment, the overhead is less than using a StringBuffer.

```
public final class StringBuilder
   extends Object
   implements Serializable, CharSequence
```

S.N.	Constructor & Description
1	**StringBuilder()** This constructs a string builder with no characters in it and an initial capacity of 16 characters.
2	**StringBuilder(CharSequence seq)** This constructs a string builder that contains the same characters as the specified CharSequence.
3	**StringBuilder(int capacity)** This constructs a string builder with no characters in it and an initial capacity specified by the capacity argument.
4	**StringBuilder(String str)** This constructs a string builder initialized to the contents of the specified string.

U2

Programa turi paprašyti vartotojo įvesti: vardą, pavardę, gim. metus;

Išvedame suformuotą tekstą į ekraną:

Vartotojas **Petras Petraitis** gimė **1980** metais.

- StringBuilder

U3

Programa turi paprašyti vartotojo įvesti: vardą, pavardę, gim. metus;

- Sukurkime metodus:
  - **void** nuskaitytiVarda(StringBuilder builder)
  - **void** nuskaitytiPavarde(StringBuilder builder)
  - **void** nuskaitytiGimimoMetus(StringBuilder builder)

Išvedame suformuotą tekstą į ekraną:

Vartotojas **Petras Petraitis** gimė **1980** metais.

- StringBuilder

Further Reading

- http://javarevisited.blogspot.com/2011/07/string-vs-stringbuffer-vs-stringbuilder.html
- http://docs.oracle.com/javase/tutorial/java/data/buffers.html
- https://www.tutorialspoint.com/java/lang/java_lang_stringbuilder.htm