

Report_Erfan_Majedi_9831099

Google Guava libraries :

Guava: Google Core Libraries for Java 28.2-jre API

Packages	
Package	Description
com.google.common.annotations	Common annotation types.
com.google.common.base	Basic utility libraries and interfaces.
com.google.common.cache	This package contains caching utilities.
com.google.common.collect	This package contains generic collection interfaces and implementations, and other utilities for working with collections.
com.google.common.escape	Interfaces, utilities, and simple implementations of escapers and encoders.
com.google.common.eventbus	The EventBus allows publish-subscribe-style communication between components without requiring the components to explicitly register with one another (and thus be aware of each other).
com.google.common.graph	An API for representing graph (node and edge) data.
com.google.common.hash	Hash functions and related structures.
com.google.common.html	Escapers for HTML.
com.google.common.io	This package contains utility methods and classes for working with Java I/O; for example input streams, output streams, readers, writers, and files.
com.google.common.math	Arithmetic functions operating on primitive values and BigInteger instances.
com.google.common.net	This package contains utility methods and classes for working with net addresses (numeric IP and domain names).
com.google.common.primitives	Static utilities for working with the eight primitive types and <code>void</code> , and value types for treating them as unsigned.
com.google.common.reflect	This package contains utilities to work with Java reflection.
com.google.common.util.concurrent	Concurrency utilities.
com.google.common.xml	Escapers for XML.

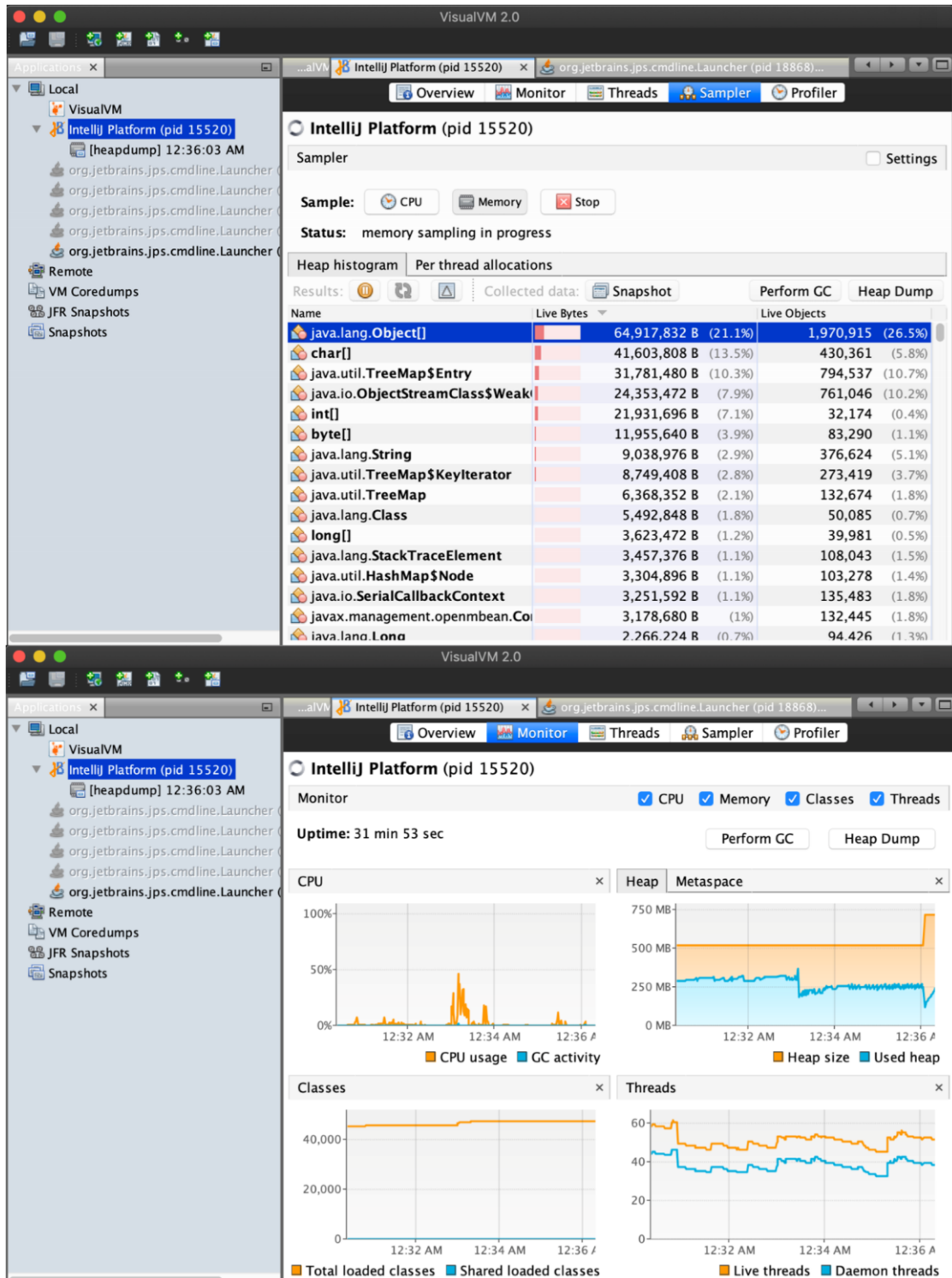
Apache Commons libraries :

Apache Commons Lang 3.9 API

Packages

Package	Description
<code>org.apache.commons.lang3</code>	Provides highly reusable static utility methods, chiefly concerned with adding value to the <code>java.lang</code> classes.
<code>org.apache.commons.lang3.arch</code>	Provides classes to work with the values of the <code>os.arch</code> system property.
<code>org.apache.commons.lang3.builder</code>	Assists in creating consistent <code>equals(Object)</code> , <code>toString()</code> , <code>hashCode()</code> , and <code>compareTo(Object)</code> methods.
<code>org.apache.commons.lang3.concurrent</code>	Provides support classes for multi-threaded programming.
<code>org.apache.commons.lang3.event</code>	Provides some useful event-based utilities.
<code>org.apache.commons.lang3.exception</code>	Provides functionality for Exceptions.
<code>org.apache.commons.lang3.math</code>	Extends <code>java.math</code> for business mathematical classes.
<code>org.apache.commons.lang3.mutable</code>	Provides typed mutable wrappers to primitive values and <code>Object</code> .
<code>org.apache.commons.lang3.reflect</code>	Accumulates common high-level uses of the <code>java.lang.reflect</code> APIs.
<code>org.apache.commons.lang3.text</code>	Provides classes for handling and manipulating text, partly as an extension to <code>java.text</code> .
<code>org.apache.commons.lang3.text.translate</code>	An API for creating text translation routines from a set of smaller building blocks.
<code>org.apache.commons.lang3.time</code>	Provides classes and methods to work with dates and durations.
<code>org.apache.commons.lang3.tuple</code>	Tuple classes, starting with a <code>Pair</code> class in version 3.0.

VisualVM screenshots:



String class methods:

Java String class methods

The java.lang.String class provides many useful methods to perform operations on sequence of char values.

No.	Method	Description
1	<code>char charAt(int index)</code>	returns char value for the particular index
2	<code>int length()</code>	returns string length
3	<code>static String format(String format, Object... args)</code>	returns a formatted string.
4	<code>static String format(Locale l, String format, Object... args)</code>	returns formatted string with given locale.
5	<code>String substring(int beginIndex)</code>	returns substring for given begin index.
6	<code>String substring(int beginIndex, int endIndex)</code>	returns substring for given begin index and end index.
7	<code>boolean contains(CharSequence s)</code>	returns true or false after matching the sequence of char value.
8	<code>static String join(CharSequence delimiter, CharSequence... elements)</code>	returns a joined string.
9	<code>static String join(CharSequence delimiter, Iterable<? extends CharSequence> elements)</code>	returns a joined string.
10	<code>boolean equals(Object another)</code>	checks the equality of string with the given object.
11	<code>boolean isEmpty()</code>	checks if string is empty.
12	<code>String concat(String str)</code>	concatenates the specified string.
13	<code>String replace(char old, char new)</code>	replaces all occurrences of the specified char value.
14	<code>String replace(CharSequence old, CharSequence new)</code>	replaces all occurrences of the specified CharSequence.

15	<code>static String equalsIgnoreCase(String another)</code>	compares another string. It doesn't check case.
16	<code>String[] split(String regex)</code>	returns a split string matching regex.
17	<code>String[] split(String regex, int limit)</code>	returns a split string matching regex and limit.
18	<code>String intern()</code>	returns an interned string.
19	<code>int indexOf(int ch)</code>	returns the specified char value index.
20	<code>int indexOf(int ch, int fromIndex)</code>	returns the specified char value index starting with given index.
21	<code>int indexOf(String substring)</code>	returns the specified substring index.
22	<code>int indexOf(String substring, int fromIndex)</code>	returns the specified substring index starting with given index.
23	<code>String toLowerCase()</code>	returns a string in lowercase.
24	<code>String toLowerCase(Locale l)</code>	returns a string in lowercase using specified locale.
25	<code>String toUpperCase()</code>	returns a string in uppercase.
26	<code>String toUpperCase(Locale l)</code>	returns a string in uppercase using specified locale.
27	<code>String trim()</code>	removes beginning and ending spaces of this string.
28	<code>static String valueOf(int value)</code>	converts given type into string. It is an overloaded method.