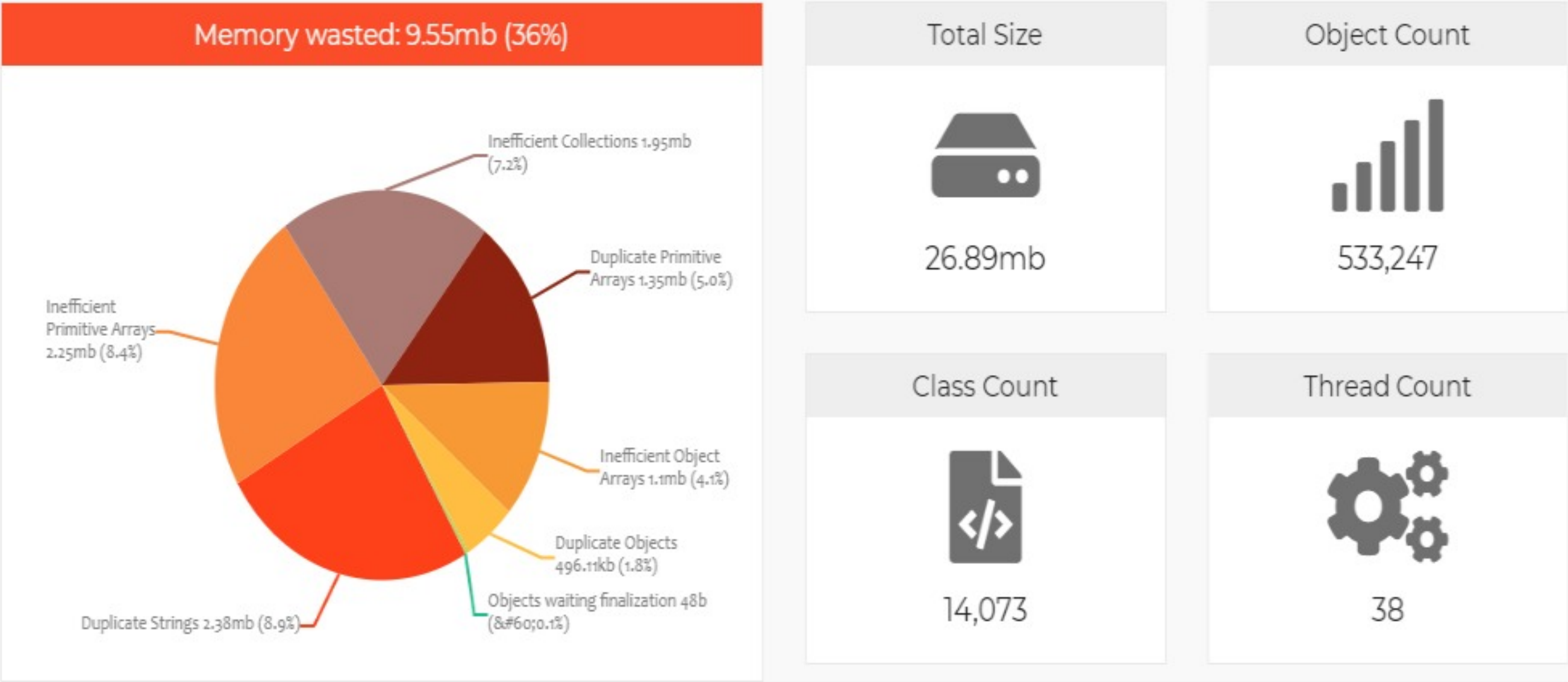


1. Heap Statistics

Learn more about [Heap Statistics](#)



2. What's in your Memory (by class)?

Learn more about [What's in Memory](#)

Class	Percentage	Size	Count
String	34.1% <div></div>	9.16mb	73,275

char[]	<div><div></div></div> 9.5%	2.56mb	74,390
java.lang.reflect.Method	<div><div></div></div> 7.3%	1.96mb	23,390
java.util.concurrent.ConcurrentHashMap	<div><div></div></div> 7.2%	1.93mb	695
j.u.l.LinkedHashMap	<div><div></div></div> 3.7%	1.01mb	9,599
byte[]	<div><div></div></div> 2.8%	781.52kb	3,857

[Show all records >>](#)

3. Large objects

Learn more about [Large Objects](#)

Name	Percentage	Size
.Java.Static.java.lang.ApplicationShutdownHooks.hooks	<div><div></div></div> 19.3%	5.18mb
Unreachable (garbage) objects	<div><div></div></div> 18.9%	5.07mb
.Java.Static.org.apache.catalina.core.StandardHostValve.MY_CLASSLOADER	<div><div></div></div> 11.0%	2.95mb
.Java.Static.org.apache.tomcat.util.modeler.Registry.registry	<div><div></div></div> 2.8%	782.48kb
.Java.Static.org.springframework.data.jpa.util.JpaMetamodel.CACHE	<div><div></div></div> 2.7%	743.7kb
... and 11293 more objects retaining 3.63mb (13.5%)		

4. Object Headers

Learn more about [Object Headers](#)

Object Header Size



12b

Total size of all headers



6.1mb (22.7%)

💡 Top Object Headers

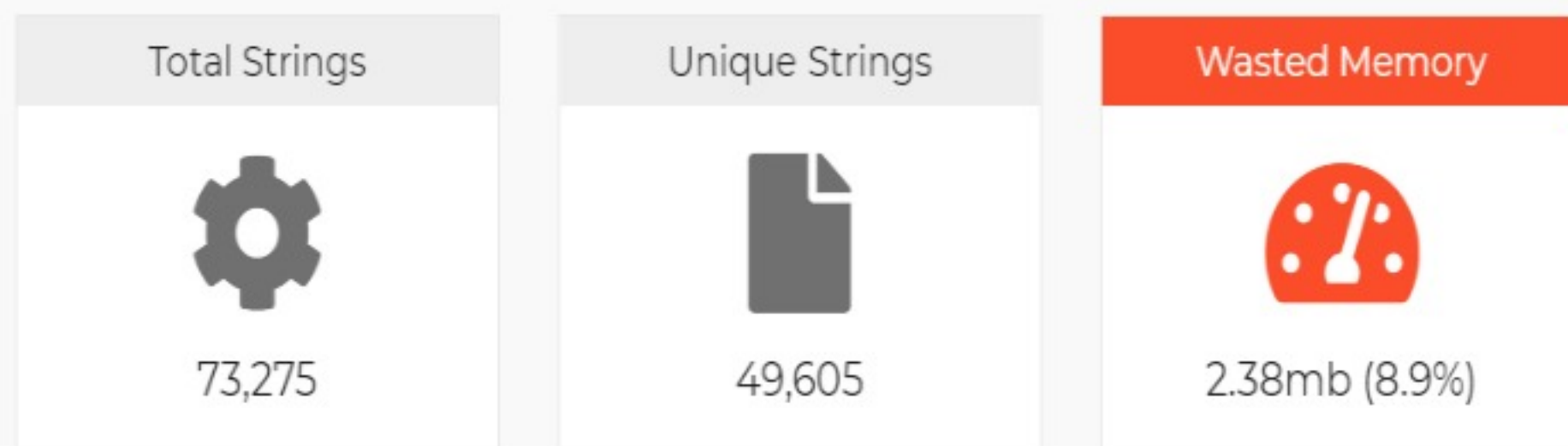
Class	Percentage	Total header size	Avg obj size	Count
char[]	3.2%	871.76kb	141	74,390
String	3.1%	858.69kb	24	73,275
java.util.concurrent.ConcurrentHashMap\$Node	2.1%	573.66kb	32	48,952
Object	1.4%	372.09kb	16	31,752
j.l.Class[]	1.0%	285.34kb	22	24,349

How to fix excessive Object headers?

Please refer to our [recommendations](#).

5. Duplicate Strings

Learn more about [Duplicate Strings](#)



Top Duplicate Strings

Duplicate String	Percentage	Wasted	Count
"java.lang.String"	0.2%	42.96kb	612
"java.lang.Object"	0.1%	35.86kb	511
"org.springframework.context.annotation.Bean"	<0.1%	26.25kb	211
"	<0.1%	25.77kb	669
"org.springframework.boot.autoconfigure.condition.ConditionalOnMissingBean"	<0.1%	21.75kb	117

[Show all records >>](#)

? Who is holding Duplicate Strings?

Object Tree	Percentage	size
java.lang.reflect.MethodSignature	0.4%	112K
java.lang.reflect.MethodSignature	0.4%	110K
sun.reflect.generics.tree.SimpleClassTypeSignature.name	0.2%	47K
{j.u.l.LinkedHashMap}keys	0.2%	42K
org.apache.tomcat.util.modeler.AttributeInfo.type	0.1%	41K

[Show all records >>](#)

🔧 How to fix Duplicate Strings?

Please refer to our [recommendations](#).

6. Inefficient collections

Learn more about [Inefficient Collections](#)

Total Collections



Inefficient collections



Wasted Memory





25,833



20,283



1.95mb (7.2%)

💡 Top inefficient collections

Problem	Percentage	Wasted
22% of <code>ju.LinkedHashMap</code> contains 1 element only	1.2%	342.04kb
11% of <code>ju.LinkedHashMap</code> contains 2 - 4 elements only	0.8%	207.58kb
28% of <code>ju.LinkedHashSet</code> contains 1 element only	0.6%	167.2kb
45% of <code>ju.ArrayList</code> contains no elements	0.6%	160.22kb
44% of <code>ju.ArrayList</code> contains 1 element only	0.6%	158.35kb

[Show all records >>](#)

❓ Who is holding Inefficient Collections?

Object Tree	Percentage	size
<code>org.springframework.boot.autoconfigure.condition.ConditionEvaluationReport\$ConditionAndOutcomes.outcomes</code>	0.3%	75K
<code>jl.Class\$AnnotationData.annotations</code>	0.3%	68K
<code>java.lang.reflect.Method.declaredAnnotations</code>	0.2%	48K
<code>java.lang.reflect.Field.declaredAnnotations</code>	0.1%	33K
<code>Unreachable (garbage) objects</code>	0.1%	31K

How to fix Inefficient Collections?

Please refer to our [recommendations](#).

7. Inefficient Object Arrays

Learn more about [Inefficient Object Arrays](#)

Total Object Arrays



68,607

Inefficient Object Arrays



41,656

Wasted Memory



1.1mb (4.1%)

Top inefficient Object Arrays

Problem	Percentage	Wasted
47% of <code>j.l.Class[]</code> declared with 1 length	1.0%	270.73kb
33% of <code>j.l.Class[]</code> declared with 0 length	0.5%	127.45kb
13% of <code>Object[]</code> contains no elements	0.4%	97.83kb
9% of <code>j.u.HashMap\$Node[]</code> contains 1 element only	0.2%	60.91kb

10% of Object[] contains 1 element only	0.2%	52.64kb
---	------	---------

[Show all records >>](#)

? Who is holding Inefficient Object Arrays?

Object Tree	Percentage	size
Unreachable (garbage) objects	0.7%	189K
java.lang.reflect.Method parameterTypes	0.3%	81K
java.lang.reflect.Method parameterTypes	0.2%	51K
java.lang.reflect.Method parameterTypes	0.2%	48K
j.l.Class\$ReflectionData declaredConstructors	0.1%	31K

🔧 How to fix Inefficient Object Arrays?

Please refer to our [recommendations](#).

8. Inefficient Primitive Arrays

Learn more about [Inefficient Primitive Arrays](#)

Total Primitive Arrays

Inefficient Primi Arrays

Wasted Memory



85,596



8,540



2.25mb (8.4%)

💡 Top inefficient Primitive Arrays

Problem	Percentage	Wasted
< 0.1% of char[] contains no elements	5.6%	1.5mb
46% of byte[] contains lot of 0s	1.3%	353.29kb
6% of int[] contains no elements	0.4%	101.8kb
16% of int[] contains lot of 0s	0.3%	73.01kb
59% of int[] declared with 0 length	0.2%	63.45kb

[Show all records >>](#)

❓ Who is holding Inefficient Primitive Arrays?

Object Tree	Percentage	size
java.io.BufferedOutputStreambuf	0.5%	150K
com.mysql.cj.protocol.ReadAheadInputStreambuf	0.5%	147K
static.sun.nio.cs.ext.IBM1364.c2b	0.3%	78K
static.sun.nio.cs.ext.IBM933.c2b	0.3%	77K
static.sun.nio.cs.ext.MS949.c2b	0.3%	73K

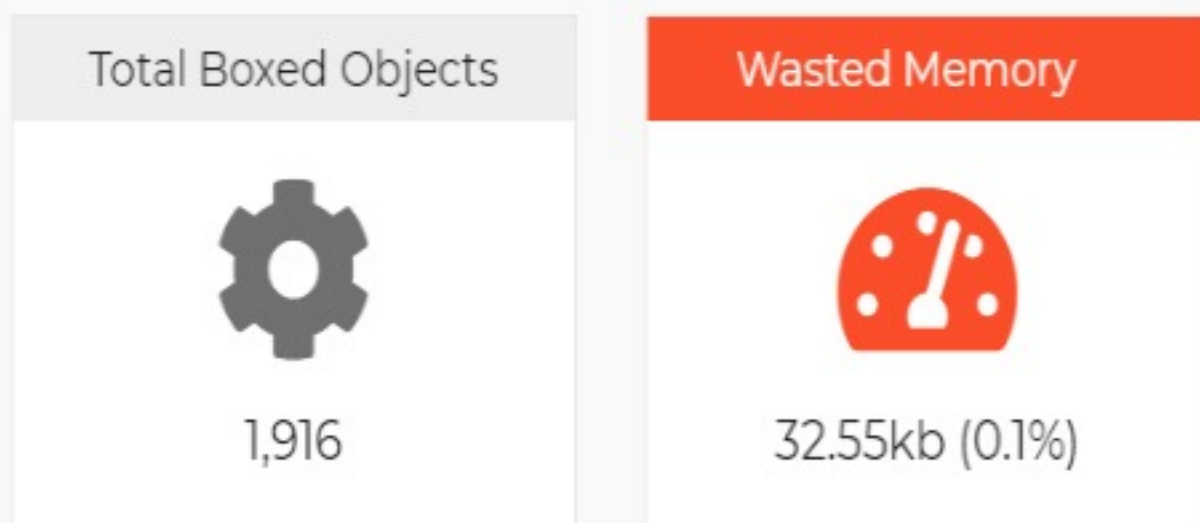
[Show all records >>](#)

How to fix Inefficient Primitive Arrays?




Please refer to our [recommendations](#).

9. Boxed Numbers

Learn more about [Boxed Numbers](#)



Who is holding Boxed Numbers?

Object Tree	Percentage	size
<code>j.l.Byte[]</code> 	<0.1% <div></div>	4K
<code>j.l.Long[]</code> 	<0.1% <div></div>	4K
<code>j.l.Short[]</code> 	<0.1% <div></div>	4K

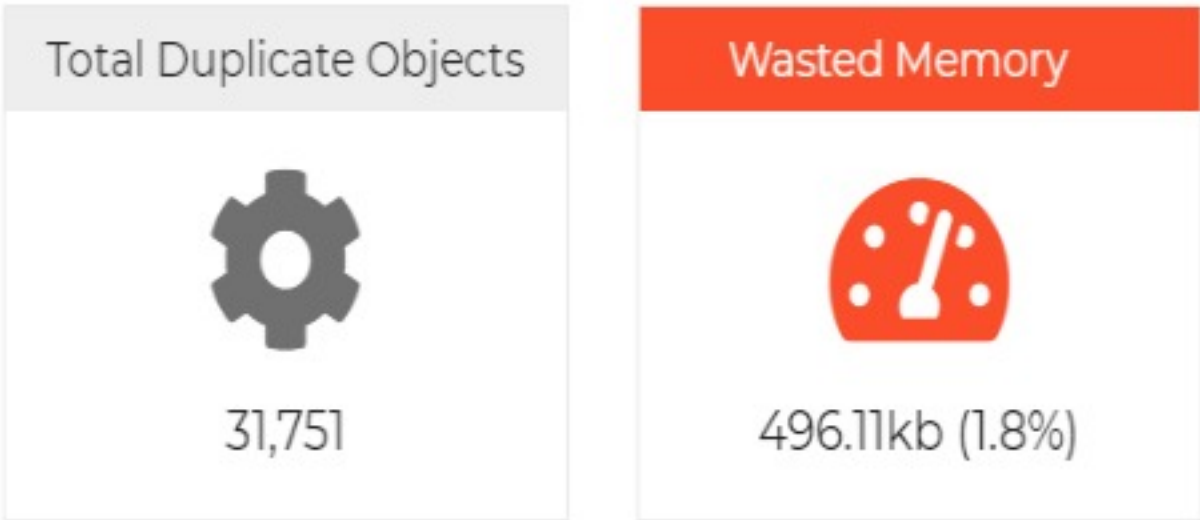


How to fix Boxed Numbers?

Please refer to our [recommendations](#).

10. Duplicate Objects

Learn more about [Duplicate Objects](#)



Types of Duplicate Objects

Object	Percentage	Wasted	Duplicate Count
Object	<div><div>1.8%</div></div>	496.11kb	31,751






Top Duplicate Objects

Duplicate Object	Percentage	Wasted	Count

Objec)	1.8%	496.11kb	31,752
--------	------	----------	--------

? Who is holding Duplicate Objects?

Object Tree	Percentage	size
{java.util.concurrent.ConcurrentHashMap}.values 	0.6%	155K
{java.util.concurrent.ConcurrentHashMap}.values 	0.6%	155K
{java.util.concurrent.ConcurrentHashMap}.values 	0.6%	154K

🔧 How to fix Duplicate Objects?

Please refer to our [recommendations](#).

11. Duplicate Primitive Arrays

Learn more about [Duplicate Primitive Arrays](#)

Total Duplicate Arrays



8,887

Wasted Memory



1.35mb (5.0%)

Array Type	Percentage	Wasted	Duplicate Count
char[]	<div><div></div>2.4%</div>	674.09kb	231
byte[]	<div><div></div>1.6%</div>	446.3kb	2,521
int[]	<div><div></div>0.9%</div>	242.67kb	5,751
boolean[]	<div><div></div><0.1%</div>	20.05kb	353
long[]	<div><div></div><0.1%</div>	2.69kb	12

[Show all records >>](#)

💡 Top Duplicate Arrays

Duplicate Array	Percentage	Wasted	Count
byte[16384](1, 44, 0, 0, 2, 3, 100, 101, 102, 0, 0, 0, 22, 64, 64, 115, 101, 115, 115, 105, ...)	0.5%	128.12kb	9
byte[16384](30, 0, 0, 0, 3, 83, 69, 76, 69, 67, 84, 32, 64, 64, 115, 101, 115, 115, 105, 111, ...)	0.5%	128.12kb	9
char[32512](.....-)	0.5%	127.03kb	3
char[29696](.....-)	0.4%	116.03kb	3
char[27392](.....-)	0.4%	107.03kb	3

[Show all records >>](#)

② Who is holding Duplicate Arrays?

🔧 Who is holding Duplicate Arrays?

Object Tree	Percentage	size
com.mysql.cj.protocol.ReadAheadInputStream.buf	0.5%	128K
java.io.BufferedOutputStream.buf	0.5%	128K
Unreachable (garbage) objects	0.2%	63K
(java.util.concurrent.ConcurrentHashMap).values	0.2%	48K
byte[][]	0.2%	48K

[Show all records >>](#)

🔧 How to fix Duplicate Arrays?

Please refer to our [recommendations](#).

12. Objects waiting for Finalization

Learn more about [Objects waiting for Finalization](#)

Wasted Memory



48b (<0.1%)

What are the objects waiting for finalization?

To see objects waiting for finalization, [click here](#) 

How to fix objects waiting for finalization?

Please refer to our [recommendations](#).

13. Threads

Learn more about [Threads](#)



To view all threads stacktrace [click here](#) 

14. Heap settings

Learn more about [Heap Settings](#)

No major recommendations.

15. System Properties

15. System Properties

Learn more about [System Properties](#)

Key	Value
PID	2700
awt.toolkit	sun.awt.windows.WToolkit
catalina.base	C:\Users\Caree2\AppData\Local\Temp\tomcat.1006059830920657220.9095
catalina.home	C:\Users\Caree2\AppData\Local\Temp\tomcat.3857286744465553031.6094
catalina.useNaming	false

[Show all records >>](#)