

[Home](#) › [Interview](#) › [Testing & Profiling/Sampling Java Apps Q&A](#) › [jvisualvm profiling](#) › 01: jvisualvm to sample Java heap memory

01: jvisualvm to sample Java heap memory

Posted on [March 15, 2016](#) by [Arulkumaran Kumaraswamipillai](#)

Java does not have a **sizeof** operator like C++ does. Java uses automatic memory management known as the Garbage Collection, hence it is not that important to evaluate size of various objects. But, for the purpose of learning & fixing any potential memory issues, I have used “jvisualvm”, which is a very handy & free profiling tool that gets shipped with the JDK. This compliments [Java primitives & objects – memory consumption interview Q&A](#)

Step 1: Java code to “sample with jvisualvm”

Never ending while loop is used so that the application stay alive to sample the Java memory to see how much memory does “MyWrapper” object occupy.

```
1
2 import java.util.concurrent.TimeUnit;
3
```

600+ Full Stack Java/JEE Interview Q&As ♥Free ♦FAQs

[open all](#) | [close all](#)

- [Ice Breaker Interview](#)
- [Core Java Interview C](#)
- [JEE Interview Q&A \(3](#)
- [Pressed for time? Jav](#)
- [SQL, XML, UML, JSC](#)
- [Hadoop & BigData Int](#)
- [Java Architecture Inte](#)
- [Scala Interview Q&As](#)
- [Spring, Hibernate, & I](#)
- [Testing & Profiling/Sa](#)
- [Automation Testing](#)
- [Code Coverage \(2\)](#)
- [Code Quality \(2\)](#)
- [jvisualvm profiling \(](#)
- [01: jvisualvm to :](#)
- [02: jvisualvm to :](#)
- [03: jvisualvm to :](#)
- [Performance Testir](#)
- [Unit Testing Q&A \(2](#)
- [Other Interview Q&A 1](#)
- [Free Java Interview](#)

```

4 public class ObjectSize {
5
6     public static void main(String[] args) throw
7
8         MyWrapper five = new MyWrapper(5);
9
10        while(true) {
11            TimeUnit.SECONDS.sleep(10);
12            System.out.println(five);
13        }
14    }
15
16    //inner class
17    static class MyWrapper {
18        int number ; // 4 bytes each
19
20        public MyWrapper(int number) {
21            this.number = number;
22        }
23    }
24 }
25
26

```

Step 2: Start jvisualvm

Run the above code as a stand-alone Java application:

- 1) **jps** will give the process id of the
- 2) **jvisualvm** will start the profiler that is shipped with JDK.

```

1
2 $ jps
3 247
4 1208 ObjectSize
5 1209 Jps
6

```

1208 is the pid (i.e process id) of the JVM in which "ObjectSize" is running.

```

1
2 $ jvisualvm
3

```

Step 3: jvisualvm GUI opens up as shown below

16 Technical Key Areas

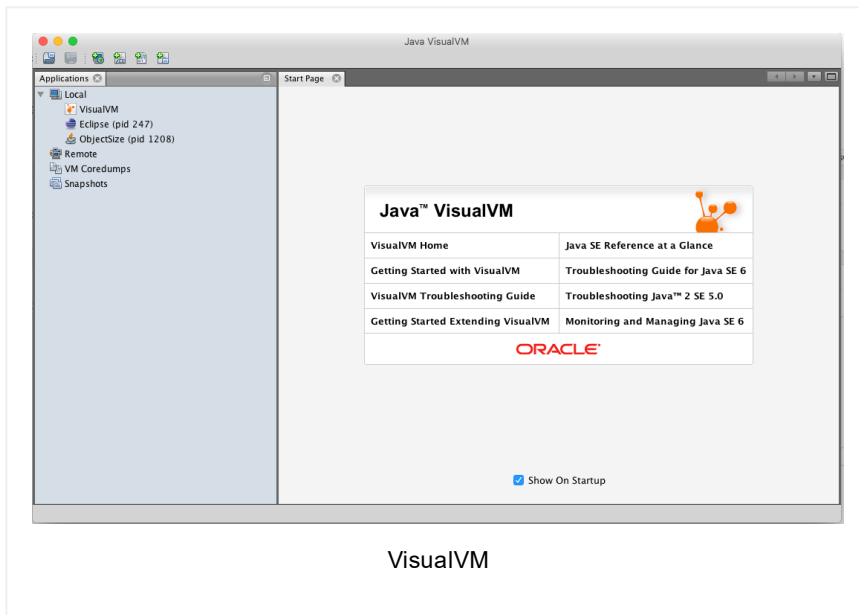
[open all](#) | [close all](#)

- [Best Practice \(6\)](#)
- [Coding \(26\)](#)
- [Concurrency \(6\)](#)
- [Design Concepts \(7\)](#)
- [Design Patterns \(11\)](#)
- [Exception Handling \(3\)](#)
- [Java Debugging \(21\)](#)
- [Judging Experience \(1\)](#)
- [Low Latency \(7\)](#)
- [Memory Management \(1\)](#)
- [Performance \(13\)](#)
- [QoS \(8\)](#)
- [Scalability \(4\)](#)
- [SDLC \(6\)](#)
- [Security \(13\)](#)
- [Transaction Management \(1\)](#)

80+ step by step Java Tutorials

[open all](#) | [close all](#)

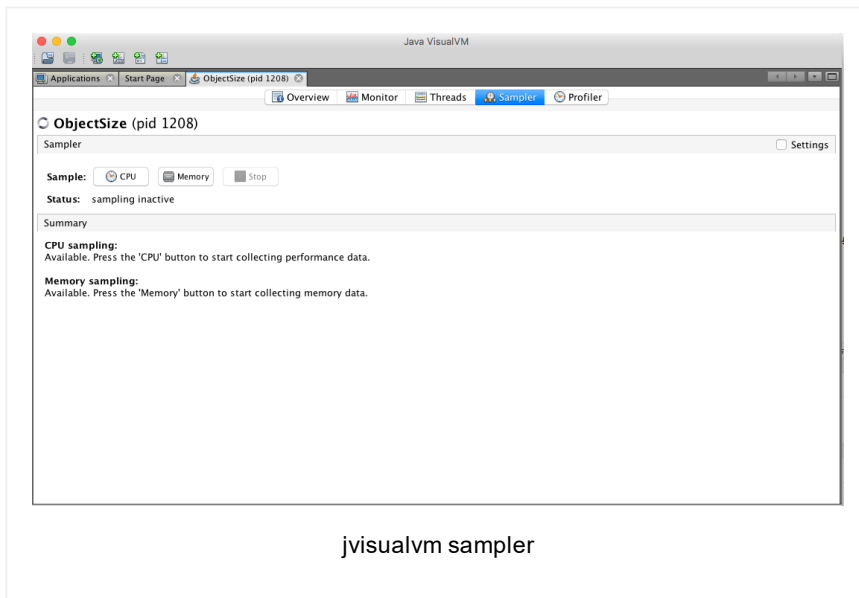
- [Setting up Tutorial \(6\)](#)
- [Tutorial - Diagnosis \(2\)](#)
- [Akka Tutorial \(9\)](#)
- [Core Java Tutorials \(2\)](#)
- [Hadoop & Spark Tutorials \(1\)](#)
- [JEE Tutorials \(19\)](#)
- [Scala Tutorials \(1\)](#)
- [Spring & Hibernate Tutorials \(1\)](#)
- [Tools Tutorials \(19\)](#)
- [Other Tutorials \(45\)](#)



You can see **ObjectSize** with pid 1208.

Step 4: jvisualvm tabs

Double click on “**ObjectSize** with pid 1208.” You will get the following screen, and select the “Sampler” tab.



Step 5: jvisualvm memory sampling

Click on the “memory” button,

100+ Java pre-interview coding tests

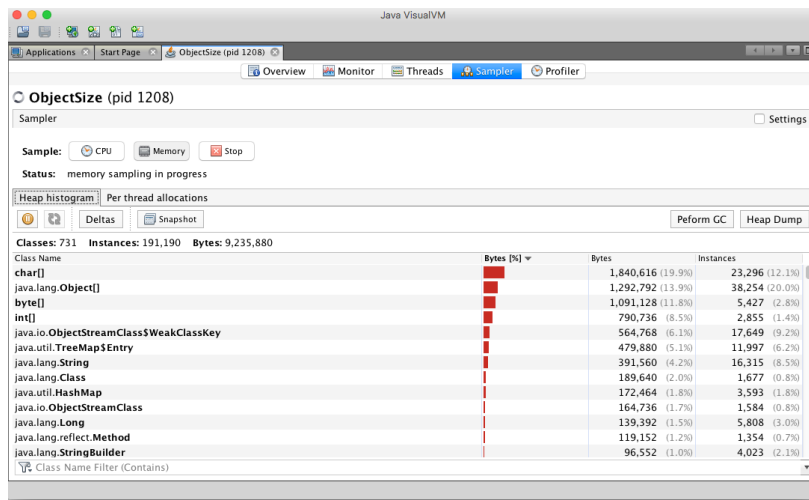
[open all](#) | [close all](#)

- [Can you write code? \(1\)](#)
- [Complete the given code \(1\)](#)
- [Converting from A to B \(1\)](#)
- [Designing your classes \(1\)](#)
- [Java Data Structures \(1\)](#)
- [Passing the unit tests \(1\)](#)
- [What is wrong with this code? \(1\)](#)
- [Writing Code Home A \(1\)](#)
- [Written Test Core Java \(1\)](#)
- [Written Test JEE \(1\)](#)

How good are your?

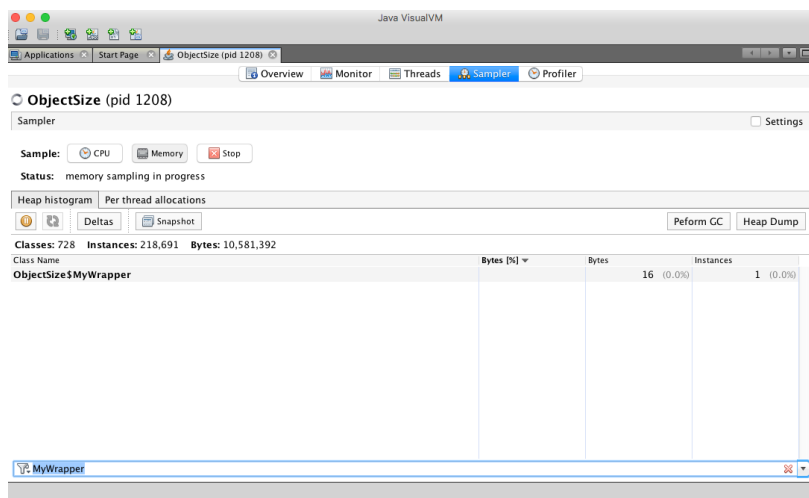
[open all](#) | [close all](#)

- [Career Making Knowledge \(1\)](#)
- [Job Hunting & Resumes \(1\)](#)



jvisualvm histogram

Filter “MyWrapper” by typing it at the bottom



MyWrapper Object on the JVM heap

Step 6: Why 16 bytes when primitive int data is only 4 bytes?

The Object metadata (aka header information) consumes memory in the heap as described below

- 1) Class** information: 32 bits = **4 bytes**.
- 2) Flags**: array or not, hashCode, etc : 32 bits = **4 bytes**.
- 3) Lock** information: synchronization 32 bits = **4 bytes**.

int number = 32 bits = **4 bytes**.

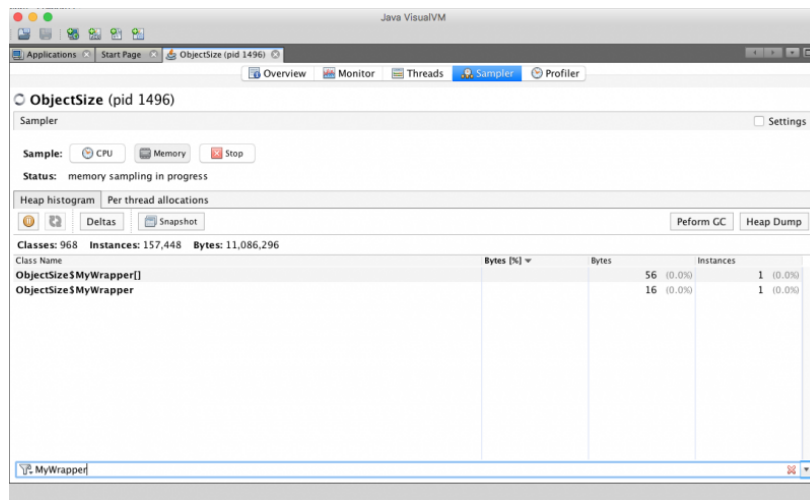
So, total 12 bytes of meta data + 4 bytes of data = **16 bytes**.

How about an array that can hold 10 MyWrapper objects

```
1
2 import java.util.concurrent.TimeUnit;
3
4 public class ObjectSize {
5
6     public static void main(String[] args) throw
7
8         MyWrapper[] five = new MyWrapper[10];
9         five[0] = new MyWrapper(0);
10
11         while(true) {
12             TimeUnit.SECONDS.sleep(10);
13             System.out.println(five);
14         }
15     }
16
17     static class MyWrapper {
18         int number ;
19
20         public MyWrapper(int number) {
21             this.number = number;
22         }
23     }
24 }
25
```

How much memory does the above MyWrapper[] take?

Follow the same steps as above.



MyWrapper[] heap histogram

- 1) The “MyWrapper” object takes “16 bytes” as before
- 2) The array MyWrapper [] takes 4 bytes * 10 = **40 bytes** for 10 elements.

The remaining 16 bytes are for the Object meta data (aka array header information).

- 1) Class information: 32 bits = **4 bytes**.
- 2) Flags: array or not, hashCode, etc: 32 bits = **4 bytes**.
- 3) Lock information: synchronization: 32 bits = **4 bytes**.
- 4) **Size of the array** 32 bits = **4 bytes**.

Popular Member Posts

♦ [11 Spring boot interview questions & answers](#)

905 views

♦ [Q11-Q23: Top 50+ Core on Java OOP Interview Questions & Answers](#)

816 views

001A: ♦ [7+ Java integration styles & patterns interview questions & answers](#)

427 views

[18 Java scenarios based interview Questions and Answers](#)

409 views

♦ [7 Java debugging interview questions & answers](#)

324 views

01b: ♦ 13 Spring basics Q8 – Q13 interview questions & answers

311 views

01: ♦ 15 Ice breaker questions asked 90% of the time in Java job interviews with hints

304 views

♦ 10 ERD (Entity-Relationship Diagrams) Interview Questions and Answers

301 views

♦ Q24-Q36: Top 50+ Core on Java classes, interfaces and generics interview questions & answers

251 views

♦ Object equals Vs == and pass by reference Vs value

234 views

25

Like

Share

Tweet

submit

reddit

3

G+1

4

Share

Bio

Latest Posts



Arulkumaran Kumaraswamipillai

Mechanical Eng to freelance Java developer in 3 yrs. Contracting since 2003, and attended 150+ Java job interviews, and often got 4 - 7 job offers to choose from. It pays to prepare. So, published Java interview Q&A books via Amazon.com in 2005, and sold 35,000+ copies. Books are outdated and replaced with this subscription based site. **945+** paid members. [join my LinkedIn Group](#). [Reviews](#)



About [Arulkumaran Kumaraswamipillai](#)



Mechanical Eng to freelance Java developer in 3 yrs. Contracting since 2003, and attended 150+ Java job interviews, and often got 4 - 7 job offers to choose from. It pays to prepare. So, published Java interview Q&A books via [Amazon.com](https://www.amazon.com) in 2005, and sold 35,000+ copies. Books are outdated and replaced with this subscription based site. **945+** paid members. [join my LinkedIn Group](#). [Reviews](#)

◀ 05: Java primitives & objects – memory consumption interview Q&A

02: jvisualvm to detect memory leak – a quick tutorial style Java demo ▶

Posted in jvisualvm profiling, JVM, Memory Management

Empowers you to open more doors, and fast-track

Technical Know Hows

☀ [Java generics in no time](#) ☀ [Top 6 tips to transforming your thinking from OOP to FP](#) ☀ [How does a HashMap internally work? What is a hashing function?](#)
☀ [10+ Java String class interview Q&As](#) ☀ [Java auto un/boxing benefits & caveats](#) ☀ [Top 11 slacknesses that can come back and bite you as an experienced Java developer or architect](#)

Non-Technical Know Hows

☀ [6 Aspects that can motivate you to fast-track your career & go places](#) ☀ [Are you reinventing yourself as a Java developer?](#) ☀ [8 tips to safeguard your Java career against offshoring](#) ☀ [My top 5 career mistakes](#)

Prepare to succeed

☀ [Turn readers of your Java CV go from “Blah blah” to “Wow”?](#) ☀ [How to prepare for Java job interviews?](#) ☀ [16 Technical Key Areas](#) ☀ [How to choose from multiple Java job offers?](#)

Select Category ▼

© Disclaimer

The contents in this Java-Success are copy righted. The author has the right to correct or enhance the current content without any prior notice.

These are general advice only, and one needs to take his/her own circumstances into consideration. The author will not be held liable for any damages caused or alleged to be caused either directly or indirectly by these materials and resources. Any trademarked names or labels used in this blog remain the property of their respective trademark owners. No guarantees are made regarding the accuracy or usefulness of content, though I do make an effort to be accurate. Links to external sites do not imply endorsement of the linked-to sites.