8张图理解Java

2014/06/12 | 分类： [基础技术](http://www.importnew.com/cat/basic) | [14 条评论](http://www.importnew.com/11725.html#comments) | 标签： [JAVA](http://www.importnew.com/tag/java)

分享到：**349**

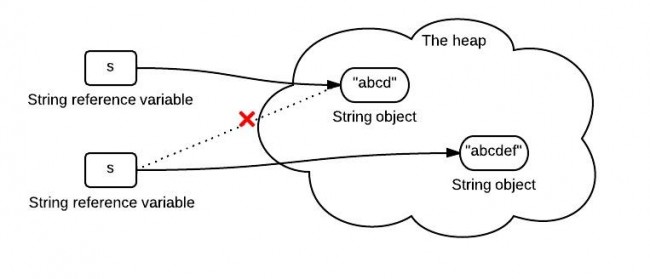
本文由 [ImportNew](http://www.importnew.com/) - [era\_misa](http://www.importnew.com/author/era_misa) 翻译自 [programcreek](http://www.programcreek.com/2013/09/top-8-diagrams-for-understanding-java/" \t "_blank)。欢迎加入[翻译小组](http://group.jobbole.com/category/feedback/trans-team/" \t "_blank)。转载请见文末要求。

一图胜千言，下面图解均来自Program Creek 网站的[Java教程](http://www.programcreek.com/java-tutorials/" \t "_blank)，目前它们拥有最多的票选。如果图解没有阐明问题，那么你可以借助它的标题来一窥究竟。

1、[字符串不变性](http://www.programcreek.com/2009/02/diagram-to-show-java-strings-immutability/" \t "_blank)

下面这张图展示了这段代码做了什么

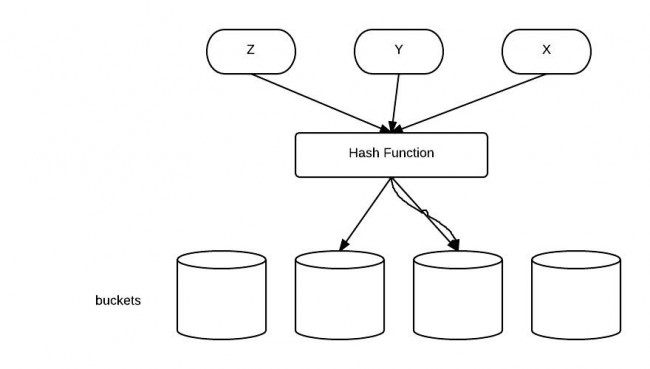
|  |  |
| --- | --- |
| 1  2 | String s = "abcd";  s = s.concat("ef"); |

[](http://www.importnew.com/11725.html/string-immutability-650x279)

2、[equals()方法、hashCode()方法的区别](http://www.programcreek.com/2011/07/java-equals-and-hashcode-contract/" \t "_blank)

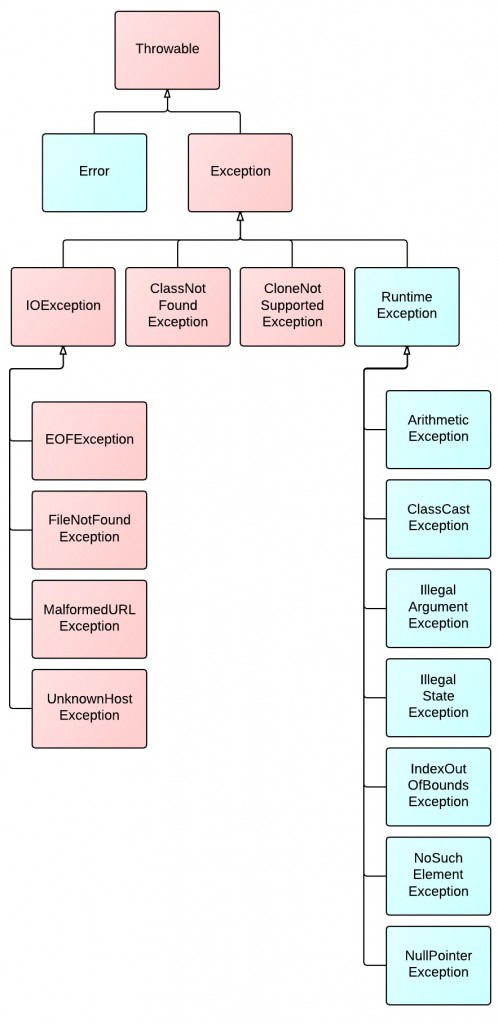
HashCode被设计用来提高性能。equals()方法与hashCode()方法的区别在于：

1. 如果两个对象相等(equal)，那么他们一定有相同的哈希值。
2. 如果两个对象的哈希值相同，但他们未必相等(equal)。

[](http://www.importnew.com/11725.html/java-hashcode-650x369)

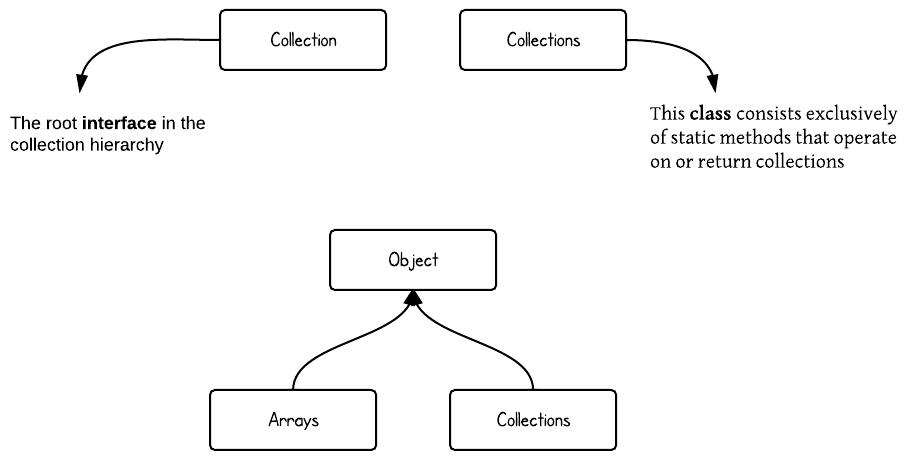
3、[Java异常类的层次结构](http://www.programcreek.com/2009/02/diagram-for-hierarchy-of-exception-classes/)

图中红色部分为受检查异常。它们必须被捕获，或者在函数中声明为抛出该异常。

[](http://www.importnew.com/11725.html/exception-hierarchy-diagram)

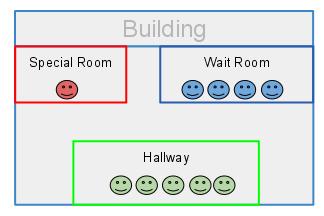
4、[集合类的层次结构](http://www.programcreek.com/2009/02/the-interface-and-class-hierarchy-for-collections/)

注意Collections和Collection的区别。（Collections包含有各种有关集合操作的静态多态方法）

[](http://www.importnew.com/11725.html/collectionvscollections-2)

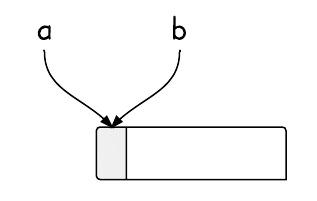
5、[Java同步](http://www.programcreek.com/2011/12/monitors-java-synchronization-mechanism/)

Java同步机制可通过类比建筑物来阐明。

[](http://www.importnew.com/11725.html/java-monitor)

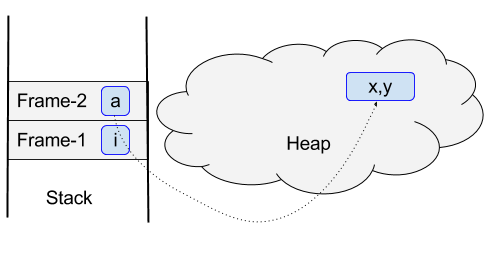
6、[别名](http://www.programcreek.com/2012/12/how-does-java-handle-aliasing/)

别名意味着有多个变量指向同一可被更新的内存块，这些别名分别是不同的对象类型。

[](http://www.importnew.com/11725.html/javaaliasing)

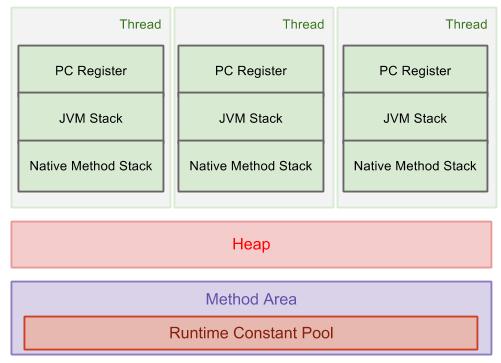
7、[堆和栈](http://www.programcreek.com/2013/04/what-does-a-java-array-look-like-in-memory/)

图解表明了方法和对象在运行时内存中的位置。

[](http://www.importnew.com/11725.html/java-array-in-memory)

8、[Java虚拟机运行时数据区域](http://www.programcreek.com/2013/04/jvm-run-time-data-areas/)

图解展示了整个虚拟机运行时数据区域的情况。

[](http://www.importnew.com/11725.html/jvm-runtime-data-area)