Last week we talked about changes to Java 9 and this week will focus on some of the changes that happened in Java 10. Researching the differences between changes in Java 9 and Java 10 has helped give me some insight into the kinds of things that change as programming languages evolve. It also enabled do come across new resources. Howtodoinjava.com is another great blog that focuses on articles about Java.

A “garbage collector” is a component that is present in many programming languages including Python and Java. It assists in reclaiming memory that is no longer being used by an application. This optimization results in much cleaner performance and resource allocation. A garbage collector will monitor memory in use and identify data that is no longer being used.

In Java 10 they introduced the concept of “garbage collector interfaces”. Rather than having garbage collectors spread throughout the code base, Java 10 allowed for the implementation of alternative garbage collector plugins. This also aided in siloing the code base so that it would be separate from other garbage collection components.

While Java 9 introduced a garbage collector called G1, also known as “Garbage First”, it was considered lacking in performance. The improvements made to Java 10’s garbage collection resulted in a streamlined experience.

It seems that Java 10 didn’t really have a lot of big conceptual changes. To me, it seems that the Java 10 release focused more on small “quality of life” fixes and improvements. It just goes to show that languages need to continue to evolve to stay relevant, even if in small ways.

<https://www.oracle.com/java/technologies/javase/10-relnote-issues.html>

<https://howtodoinjava.com/java10/java10-features/>

<https://www.tutorialspoint.com/java10/java10_enhanced_garbage_collection.htm>