Garbage collection in Java is the automatic process of reclaiming memory that is no longer in use by objects in the program. It helps in managing memory efficiently and prevents memory leaks. The Java Virtual Machine (JVM) is responsible for performing garbage collection.

you can configure and customize the garbage collection process by using various command-line options when starting the JVM. These options allow you to control the behavior of the garbage collector and optimize memory management based on your application's requirements. Here are some commonly used garbage collection options:

1. **-Xms** and **-Xmx**: These options set the initial and maximum heap sizes, respectively. For example, **-Xms512m** sets the initial heap size to 512 megabytes, and **-Xmx2g** sets the maximum heap size to 2 gigabytes.
2. **-XX:NewRatio**: This option sets the ratio between the young and old generation sizes. For example, **-XX:NewRatio=2** sets the young generation size to one-third of the total heap size.
3. **-XX:SurvivorRatio**: This option sets the ratio between the two survivor spaces in the young generation. For example, **-XX:SurvivorRatio=8** sets the survivor spaces to 1/8th of the young generation size.
4. **-XX:MaxTenuringThreshold**: This option sets the maximum tenuring threshold for objects in the young generation before they are promoted to the old generation. For example, **-XX:MaxTenuringThreshold=15** sets the maximum tenuring threshold to 15.