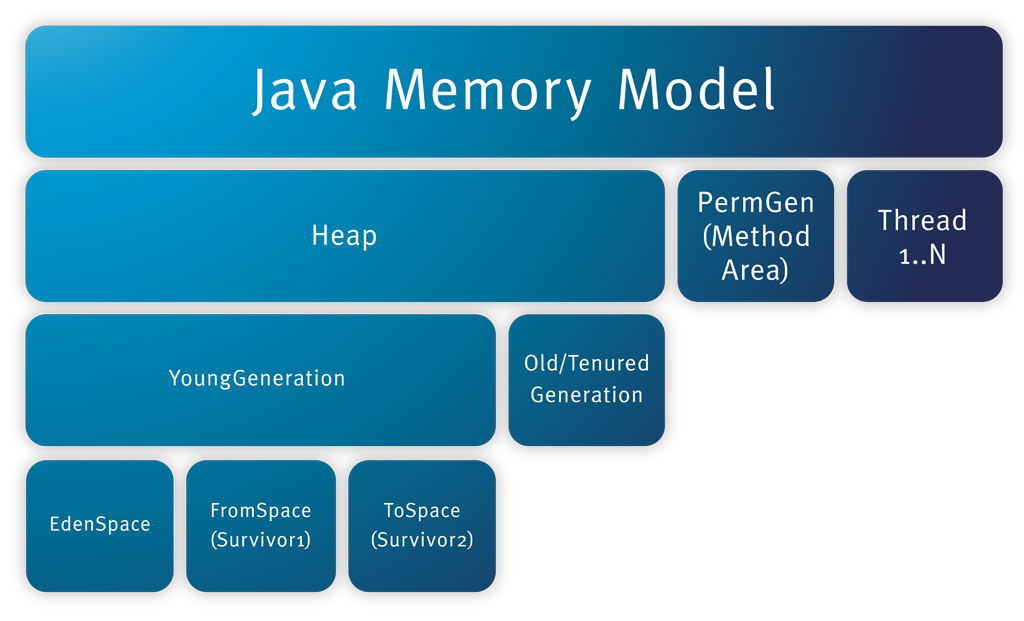
**1. Garbage Collection이란?**

쓰레기 수집(garbage collection, GC)은 메모리 관리 기법 중의 하나로, 프로그램이 동적으로 할당했던 메모리 영역 중에서 필요없게 된 영역을 해제하는 기능입니다. 영어를 그대로 읽어 가비지 컬렉션이라 부르기도 하며, 1959년 무렵 리스프의 문제를 해결하기 위해 존 매카시가 개발하였습니다.

자바는 가비지 컬렉션을 지원하는 대표적인 언어입니다(정확하게 말하면 자바가 아닌 JVM이 지원). C언어나 C++에서는 메모리를 동적으로 할당한(힙영역에 할당) 후 사용을 완료하면 free나 delete등의 키워드를 사용하여 해제하여야 합니다. 이 때 사용한 메모리를 제대로 해제 하지 않으면 메모리 누수나 여러 가지 문제가 발생합니다.

가비지 컬렉션이란 이러한 메모리 해제작업 해주는 작업을 말합니다. 개발자가 특정 키워드를 이용하여 해제할 메모리영역을 명시하지 않고 현재 애플리케이션에서 할당된 메모리 중 사용되지 않는 메모리를 해제하여 다시 사용가능 상태로 만드는 것입니다. 이것은 전적으로 JVM이 작업을 합니다. 그렇다고 해서 개발자는 가비지 컬렉션만 믿고 마음껏 메모리를 쓰고 메모리 해제에 대해 신경 쓰지 않아도 되는 것은 아닙니다. 이유는 가비지 컬렉션이 자바 애플리케이션의 성능에 굉장히 많은 영향을 주는 요소이고 개발자(혹은 시스템 운용자)는 이것을 잘 튜닝(옵션)하여 현재 사용하고 있는 애플리케이션에 최적화된 가비지 컬렉션이 수행하도록 만들 책임이 있기 때문입니다.

**2. JVM 메모리 구조**



위는 자바(JVM)의 메모리 구조를 나타냅니다. 크게 Heap과 Perm, Thread로 나눌수 있고 Heap은 다시 Young(New)영역과 Old영역, Young영역은 다시 Eden, Survior1, Survior2로 나눌 수 있습니다.

Young영역은 객체가 처음생성되었을 때(new 연산자를 이용하여 객체생성) 저장되는 영역을 나타냅니다(정확하게 말하면 new연산자를 통해 생성되는 객체들은 EdenSpace에 저장).

Old영역은 Young영역에 존재하던 객체들이 오래되어 저장되는 영역입니다.

Perm영역은 Class 혹은 method코드가 저장되는 영역으로 JVM에 의해서 사용됩니다.

**3. Garbege Collection의 종류**

**Minor GC**

자바의 GC는 크게 두 가지로 Minor GC와 Full GC로 나눌수 있습니다. 그 중 Minor GC는 Young영역에 발생하는 GC를 나타냅니다. 동작방식은 아래와 같습니다.

1) 처음 Minor GC가 발생하기 전 할당되어있는 모든 객체들은 Eden영역에만 존재하게 됩니다. 이때 Minor GC가 발생하면 Eden영역 중 활성객체가 Survivor1영역으로 이동됩니다. 이 후 비활성 객체만 남은 Eden영역이 클리어 됩니다.

2) 다음 Minor GC가 발생하면 처음 Minor GC 이후 생성된 객체들이 존재하는 Eden영역과 처음 Minor GC이후 생존한 객체들이 저장된 Survivor1의 영역에서 존재하는 활성객체들이 Survivor2로 이동되고 Eden영역과 Survivor1영역이 클리어 됩니다.

3) 이 후 위 작업이 Survivor1과 Survivor2를 번갈아가면서 계속 수행되고 계속해서 활성객체로 생존하는 객체들은 시간이 지남에 따라 Old영역으로 이동됩니다.

**Full GC**

위에서 Minor GC란 Young영역의 GC라고 하였습니다. 이와 상응되게 Full GC는 Old영역에 발생하는 GC를 나타냅니다. Full GC는 특정 순서가 있는 작업을 수행한다기 보다 Old영역을 스캔하여 객체들의 참조상태를 표시하고 참조되지 않는 객체를 해제(Mark & Compact)하는 작업을 나타냅니다. 이러한 Full GC는 속도가 느리고 성능에 아주 큰 영향이 있습니다. 이유는 Full GC가 수행되는 동안 자바 애플리케이션의 멈춤 현상이 동행되기 때문입니다.

**4. 여러 가지 Garbege Collection 알고리즘**

- Default Collection

- Parallel GC

- Concurrent GC

- Incremental GC(Train GC)