

[Java] Java 8 정리

Java 8은 가장 큰 변화가 있던 버전이다.
자바로 구현하기 힘들었던 병렬 프로세싱을 활용할 수 있게 된 버전이기 때문

시대가 발전하면서 이제 PC에서 멀티 코어 이상은 대중화되었다. 이제 수많은 데이터를 효율적으로 처리하기 위해서 '병렬' 처리는 필수적이다.

자바 프로그래밍은 다른 언어에 비해 병렬 처리가 쉽지 않다. 물론, 스레드를 사용하면 놓고 있는 유휴 코어를 활용할 수 있다. (대표적으로 스레드 풀) 하지만 개발자가 관리하기 어렵고, 사용하면서 많은 에러가 발생할 수 있는 단점이 존재한다.

이를 해결하기 위해 8버전에서는 좀 더 개발자들이 병렬 처리를 쉽고 간편하게 할 수 있도록 기능들이 추가되었다.

크게 3가지 기능이 8버전에서 추가되었다.

- Stream API
- Method Reference & Lambda
- Default Method

Stream API는 병렬 연산을 지원하는 API다. 이제 기존에 병렬 처리를 위해 사용하던 `synchronized`를 사용하지 않아도 된다. `synchronized`는 에러를 유발할 가능성과 비용 측면에서 문제점이 많은 단점이 있었다.

Stream API는 주어진 항목들을 연속으로 제공하는 기능이다. 파이프라인을 구축하여, 진행되는 순서는 정해져 있지만 동시에 작업을 처리하는 것이 가능하다.

스트림 파이프라인이 작업을 처리할 때 여러 CPU 코어에 할당 작업을 진행한다. 이를 통해서 하나의 큰 항목을 처리할 때 효율적으로 작업할 수 있는 것이다. 즉, 스레드를 사용하지 않아도 병렬 처리를 간편히 할 수 있게 되었다.

또한, 메소드 레퍼런스와 람다를 자바에서도 활용할 수 있게 되면서, 동작 파라미터를 구현할 수 있게 되었다. 기존에도 익명 클래스로 구현은 가능했지만, 코드가 복잡해지고 재사용이 힘든 단점을 해결할 수 있게 되었다.

[참고 자료]

- [링크](#)