

HashMap 동시성 문제

JAVA

HashMap은 다중 스레드 환경에서 동시에 HashMap에 접근하여 수정하면 문제가 발생할 수 있다.

이를 HashMap은 스레드 안전(thread-safe)하지 않다고 표현한다.

동시성 문제

- Race condition
 - 여러 스레드가 동시에 같은 HashMap의 같은 위치에 동시 접근하는 경우 일부 데이터가 누락되거나 덮어 씌워질 수 있다.
- Deadlock
 - 두 개 이상의 스레드가 HashMap을 서로 대기하는 경우 발생할 수 있다.

해결법

이같은 문제를 해결하기 위해서 여러 스레드가 동시에 HashMap에 접근하는것을 막거나 스레드에 안전한 ConcurrentHashMap을 사용한다.

ConcurrentHashMap은 내부에서 세그먼트(segment)를 사용하여 여러 스레드가 동시에 HashMap의 서로 다른 부분을 수정할 수 있도록하고, 안전한 연산을 제공해 스레드 안전하게 설계되어있다.

사용법

```
import java.util.Map;
import java.util.concurrent.ConcurrentHashMap;

Map<String, String> concurrentHashMap = new ConcurrentHashMap<>();
concurrentHashMap.put("key", "value");
```