



스터디 5주차

2021.09.12

섹션 5: 싱글톤 컨테이너

섹션 6: 컴포넌트 스캔

섹션 7: 의존관계 자동 주입

ThreadLocal

thread-local 변수를 제공하는 클래스이다.

thread-local 변수란 말그대로 thread 내부에서 사용되는 지역변수를 의미

get, set 메소드를 통해 접근하는 각 스레드가 독립적으로 초기화된 변수의 사본을 갖는다는 것이 일반변수와의 차이

ThreadLocal 인스턴스들은 일반적으로 상태를 스레드와 연결하려고 하는 클래스들의 private static 필드

ex. 유저 ID 또는 트랜잭션 ID

출처 : 오라클 docs



멀티 스레드 환경에서 각 스레드마다 독립적인 변수를 가지고,
get(), set() 메소드를 통해 값에 대해 접근할 수 있다.

ThreadLocal 이용한 싱글톤

각 스레드가 자체 버전의 객체를 필요로 하는 애플리케이션에서 유용

```
public class SingletonThreadLocalExample {  
  
    private SingletonThreadLocalExample () { }  
  
    private static ThreadLocal<SingletonThreadLocalExample> _instance = new ThreadLocal<SingletonThreadLocalExample>();  
    public int _count;  
  
    public static SingletonThreadLocalExample getInstance() {  
        if (_instance.get() == null) {  
            _instance.set(new SingletonThreadLocalExample());  
        }  
        return _instance.get();  
    }  
}
```

ThreadLocal 이용한 싱글톤

```
public class ThreadLocalTestMain implements Runnable {

    public static void main(String[] args) {

        ThreadLocalTestMain r = new ThreadLocalTestMain();

        Thread thread1 = new Thread(r);
        Thread thread2 = new Thread(r);

        thread1.start();
        thread2.start();

    }

    public void run() {

        SingletonThreadLocalExample instance = SingletonThreadLocalExample.getInstance();

        while (true) {
            instance._count++;
            System.out.println(String.format("%d: %d", Thread.currentThread().getId(), instance._count));
            try {
                Thread.sleep(100);
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
        }
    }
}
```

```
13: 1
14: 1
14: 2
13: 2
14: 3
13: 3
14: 4
```

ThreadLocal 이용한 싱글톤

Runnable 인스턴스를 1개 생성하여 두 개의 다른 스레드에게 전달한다.

각 스레드는 run() 메소드를 호출하고, ThreadLocal 인스턴스에 각자 다른 값을 세팅한다.

만약 여기서 ThreadLocal 객체가 없고 set() 메소드 호출부가 동기화(synchronized)되어 있다면, 두 번째 스레드의 값이 첫 번째 스레드가 세팅한 값을 덮어쓸 것이다.