KB금융그룹 | 국민의 평생 금융파트너

2025년 상반기 K-디지털 트레이닝

# 멀티스레드 - 동기화

쓰레드 간에 데이터를 공유할때 생기는 문제에 대하여

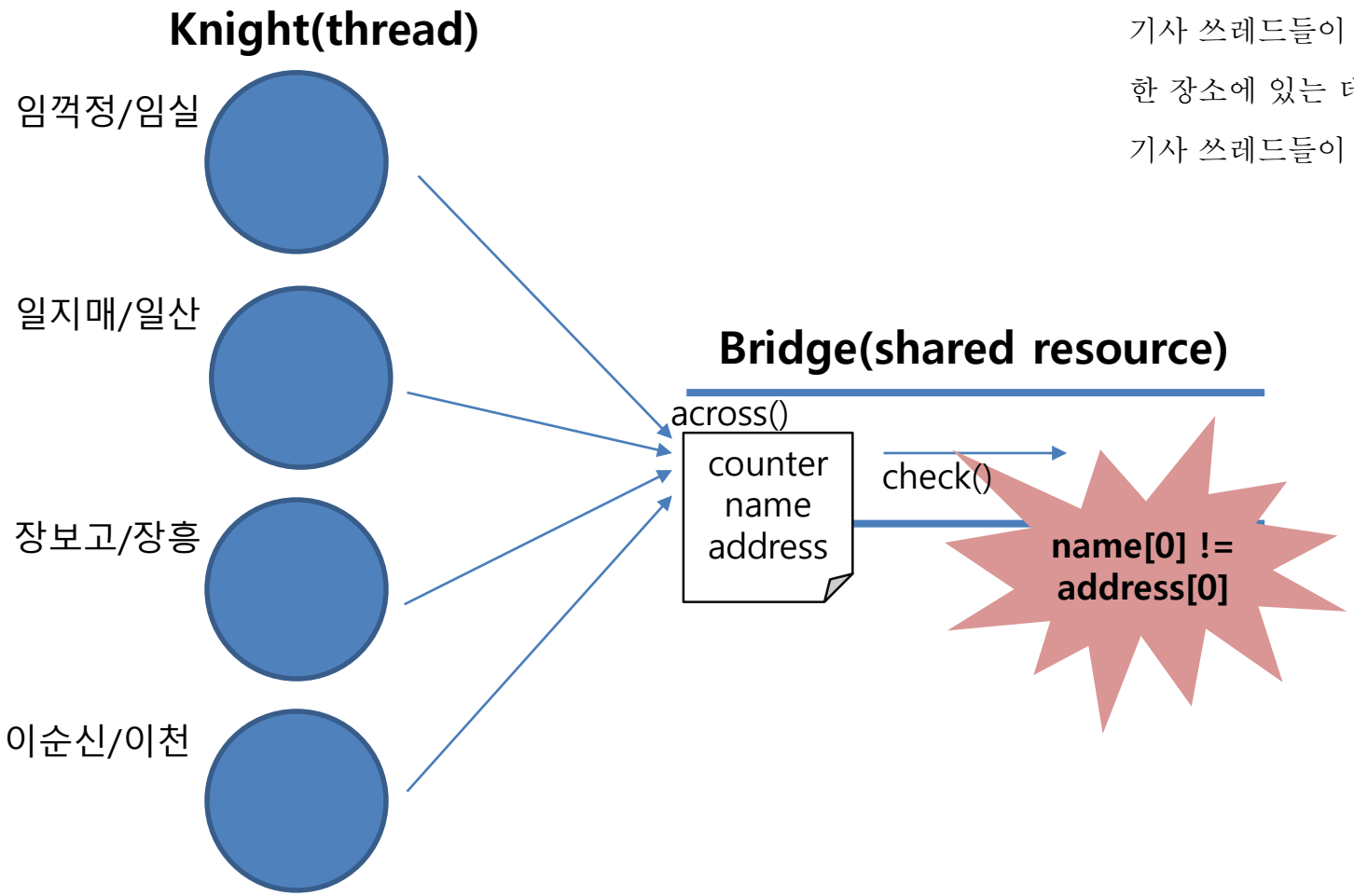
## [KB] IT's Your Life

KB국민은행

# 멀티스레드 - 동기화

## 상황

시나리오




기사 쓰레드들이

한 장소에 있는 데이터 들에 접근.

기사 쓰레드들이 브릿지 객체의 across메소드를 호출하며 접근

## Bridge.java

```
package ch14.sec00;

public class Bridge {
  int counter;  // 전체 통과 회수
  String name;  // 현재 통과 중인 기사의 이름
  String address; // 현태 통과 중인 기사의 주소

  public Bridge() {
    counter = 0;
    name = "아무개";
    address = "모름";
  }

  // 다리를 지나는 기사의 이름과 주소, counter 값 기록
  public void across(String name, String address) {
    counter++;
    this.name = name;
    this.address = address;
    check();
  }
```

동시 접근시 동기화ㅏ 제어를 안하면
t1에서눈 name을
t2에선 address를
혹은같은걸 동시에
난리난다 막

## Bridge.java

```
  public void check() {
    if(name.charAt(0) != address.charAt(0)) {
      System.out.println("문제 발생!!! " + this );
    }
  }

  @Override
  public String toString() {
    return "Bridge{" +
        "counter=" + counter +
        ", name='" + name + '\'' +
        ", address='" + address + '\'' +
        '}';
  }
}
```

롬복 @ToString해도 된다

## Knight.java

```
package ch14.sec00;

public class Knight extends Thread {
  Bridge bridge;  //  공유자원
  String name;    // 기사 이름
  String address; // 기사 주소

  public Knight(Bridge bridge, String name, String address) {
    this.bridge = bridge;
    this.name = name;
    this.address = address;
  }

  @Override
  public void run() {
    System.out.println(name + " 기사가 도전한다.");
    while(true) {
      bridge.across(name, address);
    }
  }
}
```

상속으로 쓰레드 클래스 생성!

외부에서 주입될거임. 그러니까 멤버로 갖고 있어도 외부에 잇는 것을 참조하니까 공유맞제

롬복 @AllArgConstructor

# 멀티스레드 - 동기화

## Main.java

```
package ch14.sec00;

public class Main {
  public static void main(String[] args) {
    System.out.println("시뮬레이션을 시작한다.");

    Bridge bridge = new Bridge();
    new Knight(bridge, "홍길동", "홍천").start();
    new Knight(bridge, "임꺽정", "임실").start();
    new Knight(bridge, "일지매", "일산").start();
    new Knight(bridge, "장보고", "장흥").start();
    new Knight(bridge, "이순신", "이천").start();
  }
}
```

결과가 이상해요!!!

```
시뮬레이션을 시작한다.
홍길동 기사가 도전한다.
일지매 기사가 도전한다.
장보고 기사가 도전한다.
임꺽정 기사가 도전한다.
이순신 기사가 도전한다.
문제 발생!!! Bridge{counter=1506, name='장보고', address='이천'}
문제 발생!!! Bridge{counter=1506, name='장보고', address='이천'}
문제 발생!!! Bridge{counter=8614, name='장보고', address='장흥'}
문제 발생!!! Bridge{counter=9723, name='장보고', address='장흥'}
문제 발생!!! Bridge{counter=10559, name='장보고', address='장흥'}
문제 발생!!! Bridge{counter=11264, name='임꺽정', address='임실'}
문제 발생!!! Bridge{counter=1506, name='장보고', address='이천'}
문제 발생!!! Bridge{counter=12270, name='임꺽정', address='임실'}
문제 발생!!! Bridge{counter=12730, name='일지매', address='일산'}
문제 발생!!! Bridge{counter=13511, name='일지매', address='일산'}
문제 발생!!! Bridge{counter=1506, name='장보고', address='이천'}
문제 발생!!! Bridge{counter=1506, name='장보고', address='이천'}
문제 발생!!! Bridge{counter=17161, name='홍길동', address='홍천'}
문제 발생!!! Bridge{counter=16330, name='홍길동', address='홍천'}
문제 발생!!! Bridge{counter=15240, name='일지매', address='일산'}
문제 발생!!! Bridge{counter=14365, name='일지매', address='일산'}
문제 발생!!! Bridge{counter=19091, name='홍길동', address='홍천'}
문제 발생!!! Bridge{counter=18347, name='홍길동', address='홍천'}
문제 발생!!! Bridge{counter=18368, name='홍길동', address='홍천'}
문제 발생!!! Bridge{counter=20521, name='장보고', address='장흥'}
```

# 멀티스레드 - 동기화

## 동기화

- 여러 스레드가 공유 자원을 사용할 때 경쟁 상태가 발생
- Critical Section(CS)
  - 경쟁 상태가 발생하는 메서드 또는 코드 블럭
  - 주로 공유 자원의 쓰기 코드가 CS가 됨

- synchronized 키워드 자동으로 락 관리를 해줌!!
  - 한 스레드가 CS에 진입할 때 lock
  - 한 번에 한 스레드만 진입하도록 보장
  - 경쟁하는 나머지 스레드는 대기 상태로 진입
  - CS를 벗어 날 때 unlock,
  - 대기 상태인 스레드 깨어나 다시 경쟁하여 한 스레드만 진입

## Bridge.java

```
package ch14.sec00;

public class Bridge {
  ...

  // 다리를 지나는 기사의 이름과 주소, counter 값 기록
  synchronized public void across(String name, String address) {
    counter++;
    this.name = name;
    this.address = address;
    check();
  }

  ...
}
```

오!!

멀티 쓰레드 패턴

producer consumer 패턴

vod 앱을 예로

요인이 많다

클라쪽에서 출력쓰레드에서
출력이 느려서 큐에 쌓일 수도

수신 쓰레드에서 수신 받는게 혹은 수신 받은걸 쓰는게 느릴 수도

혹은 송신측에서 보내는 과정이 느려질 수도.

이때 동기화가 필요.