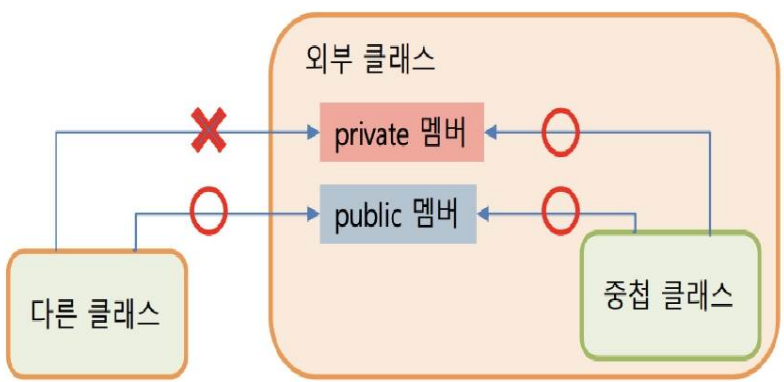
중첩 클래스

▣ 중첩 클래스

- 클래스가 여러 클래스와 관계를 맺는 경우에는 독립적으로 선언하는 것이 좋으나,
  특정 클래스만 관계를 맺을 경우에는 중첩 클래스로 선언하는 것이 유지보수에 도움이 되는 경우가 많다.

- 중첩 클래스(Nested Class)란 클래스 내부에 선언한 클래스를 말하는데,
  중첩 클래스를 사용하면 클래스의 멤버를 쉽게 사용할 수 있고, 외부에는 중첩 관계 클래스를 감춤으로써
  코드의 복잡성을 줄일 수 있다는 장점이 있다.



중첩 클래스는 외부 클래스를
상속할 필요 없이
외부 클래스의 private 멤버까지
접근할 수 있다.

A $ B .class
바깥 클래스 멤버 클래스

A $1 B .class
바깥 클래스 로컬 클래스

선언 위치에 따른 분류		선언 위치	객체 생성 조건

멤버 클래스	인스턴스 멤버 클래스	class A {
		  class B { … }
		}
			A 객체를 생성해야만
			B 객체를 생성할 수 있음

	정적 멤버 클래스	class A {
		  static class B { … }
		}
			A 객체를 생성하지 않아도
			B 객체를 생성할 수 있음

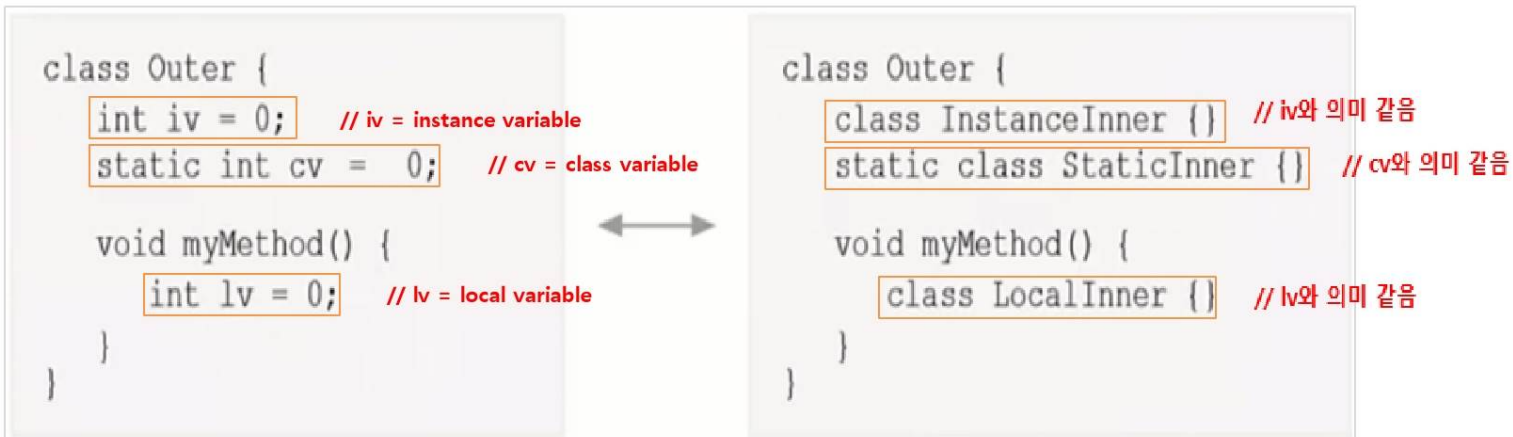
로컬 클래스		class A {
		  void method() {
		  class B { … }
		  }
		}
			method가 실행할 때만
			B 객체를 생성할 수 있음

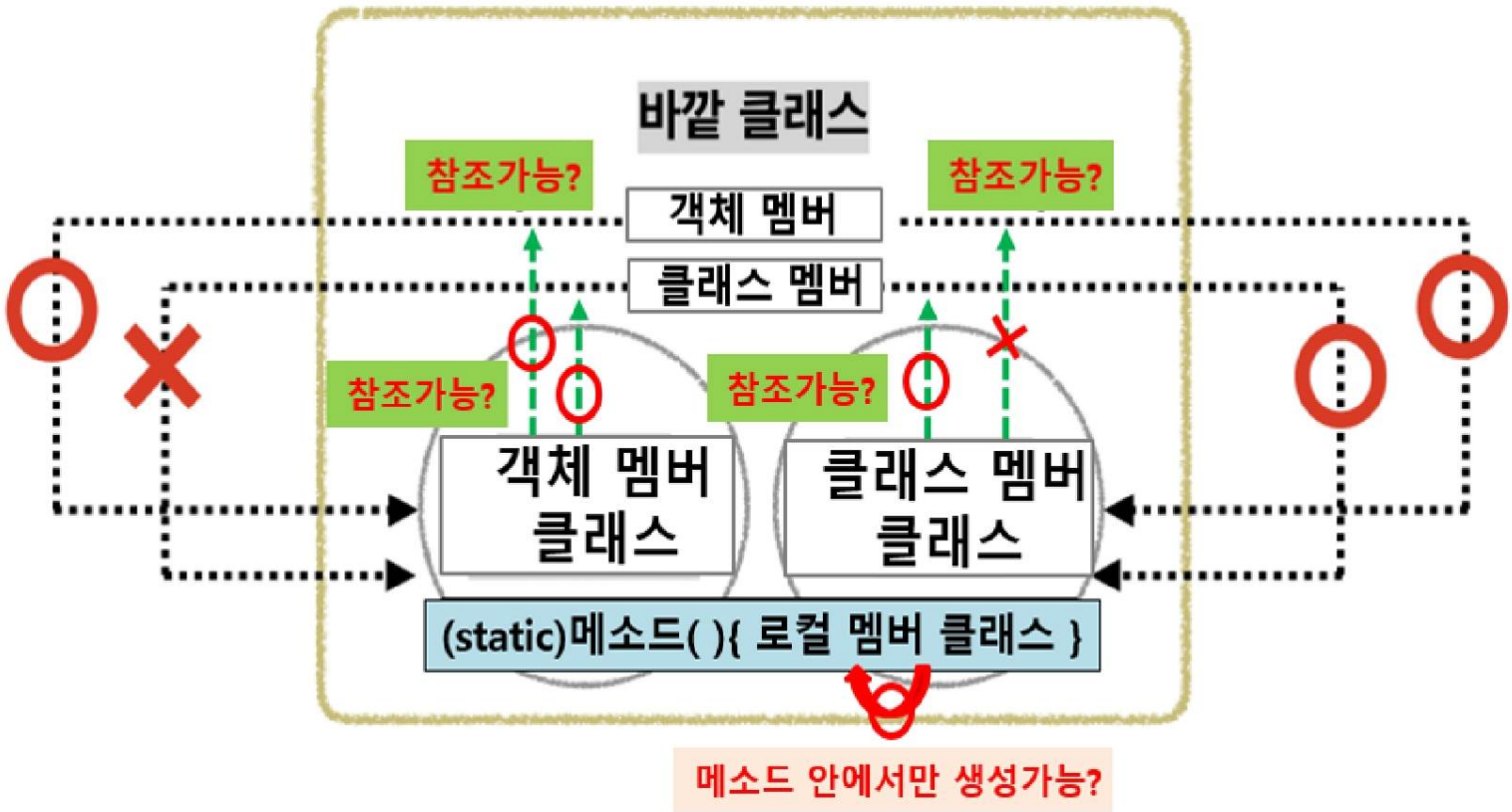
✓ 컴파일 후
class파일 확인

InnerTest.class
InnerTest$StaticInner.class
InnerTest$IntanceInner.class
InnerTest$1LocalInner.class

```
class InnerTest {
    static class StaticInner {
    }
    class IntanceInner {
    }
    void Method() {
        class LocalInner {
        }
    }
}
```

내부 클래스의 종류와 유효범위(scope)는 변수와 동일




바깥 클래스
참조가능?
객체 멤버
클래스 멤버
참조가능?
참조가능?
참조가능?
객체 멤버 클래스
클래스 멤버 클래스
(static)메소드( ){ 로컬 멤버 클래스 }
메소드 안에서만 생성가능?