

## 1. 다음에서 클로저를 분석하시오

위 코드에서 `outerFunction`은 내부에 `innerFunction` 함수를 정의하고, `innerFunction`을 반환합니다. 반환된 `innerFunction`은 변수 `innerFunc`에 저장됩니다.

`innerFunction`은 `outerVar`와 `innerVar` 두 개의 변수를 사용합니다. `outerVar`는 `outerFunction`에서 정의된 변수이며, `innerFunction`에서 접근할 수 있습니다. `innerVar`는 `innerFunction`에서 정의된 변수이며, `innerFunction` 내부에서만 사용할 수 있습니다.

여기서 중요한 점은 `innerFunction`이 `outerFunction`의 스코프에 접근할 수 있다는 것입니다. `innerFunction`은 자신의 내부 스코프에 `outerVar`에 접근하면서, `outerFunction`의 실행이 종료된 이후에도 `outerVar`에 접근할 수 있게 됩니다. 이러한 현상을 클로저(Closure)라고 부릅니다.

따라서 `innerFunc()`을 호출하면, `outerVar`와 `innerVar`의 합인 30이 콘솔에 출력됩니다. 이는 `innerFunction`이 `outerVar`에 대한 클로저를 생성하여 `outerFunction`의 실행이 종료된 이후에도 `outerVar`에 접근할 수 있기 때문입니다.