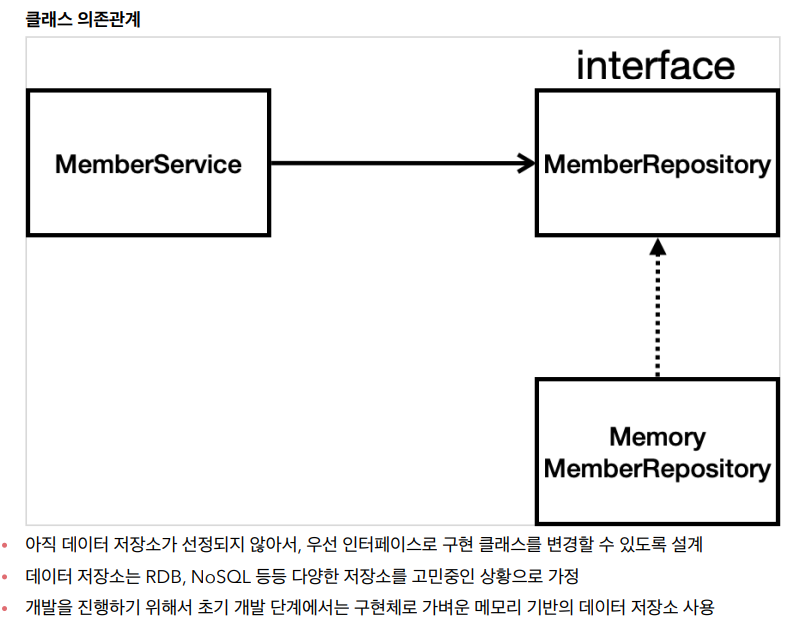
**InterFace**



인터페이스 : 상속 관계 X , 클래스 간 유사성을 인코딩하기 위해 사용

예를 들어, 자동차와 사람 둘 다 달릴 수 있지만 이 둘을 같은 Runner 클래스의 자식으로 두기엔 유사성이 맞지 않을 것. 이런 경우, Runnable이라는 인터페이스를 만들고 이 클래스를 양쪽에서 각자 구현하도록 하는 것이 인터페이스의 역할.

텍스트이(가) 표시된 사진

자동 생성된 설명텍스트이(가) 표시된 사진

자동 생성된 설명

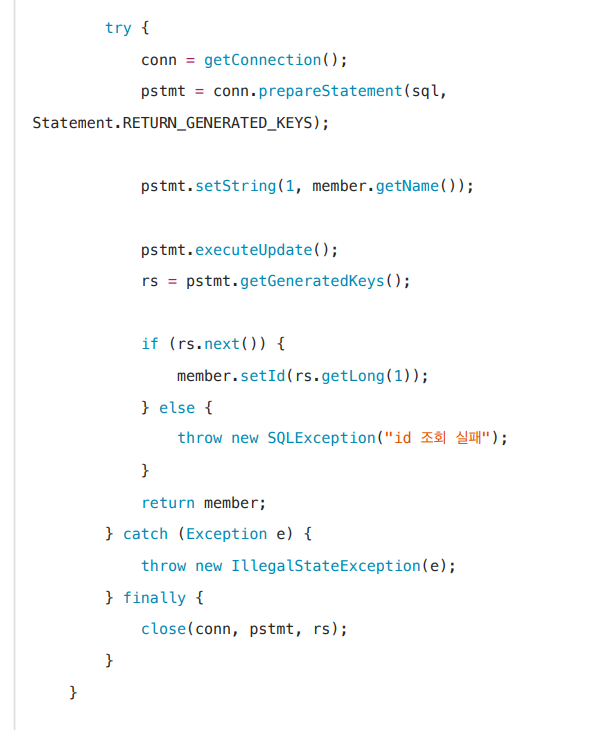
**추상클래스**

관련된 클래스 사이에서 코드를 공유하고 싶다면 추상 클래스 사용. 공통적 필드가 많거나 public을 사용해야하는 경우, 정적(Static)이 아닌 필드나 상수가 아닌 필드를 선언하기 원할 때.

**인터페이스**

관련없는 클래스들이 같은 동작을 실행시켜 구현하려 할 때, 특정 자료형의 동작을 지정하고 싶지만 누가 구현하던 신경쓸 필요가 없을 때, 다중 상속이 필요할 때.(왜?)

**Try - Catch**



텍스트이(가) 표시된 사진

자동 생성된 설명

예외발생은 코드에 에러가 발생하게 될 때, 사용자에게 미리 알려 데이터를 저장하게 한 뒤 사용자가 우아하게 종료하기 위함