**CHAPTER1**

**리팩토링 첫 단계**

아무리 체계적이 리팩토링 공식을 이용해 버그가 생길 수 있는 대부분의 원인을 방지하더라도, 인간인 이상 실수할 수 있기 때문에 신뢰도 높은 테스트 작성은 필수다.

번거로운데 굳이 변수명을 변경할 필요성이 있나?

🡺당연히 있다. 좋은 코드는 그것이 무슨 기능을 하는지 분명히 드러나야 하는데, 코드의 기능을 분명히 드러내는 열쇠가 바로 직관적인 변수명이다.

**컴퓨터가 인식 가능한 코드는 바보라도 작성할 수 있지만, 인간이 이해할 수 있는 코드는 실력 있는 프로그래머만 작성할 수 있다.**

상태 패턴을 적용하려면 세 가지 리팩토링 기법을 사용해야 한다.

1) 분류 부호를 상태/전략 패턴으로 전환 기법을 실시해서 분류 부호의 기능을 상태 패턴안으로 옮겨야 한다.

2) 메서드 이동 기법을 실시해서 switch 문을 State(interface) 클래스 안으로 옮겨야 한다.

3) 조건문을 재정의로 전환 기법을 실시해서 switch 문을 없애야 한다.

