

5957 객체지향 프로그래밍

60222117 이서현

목차

1. Remote method Invocation Stub과 Skeleton
2. 프로그램 설계 원칙 - SOLID

주제 : RMI(remote method invocation)

1. Remote method Invocation Stub과 Skeleton

“RMI는 자바 라이브러리인데, 분산 프로그래밍을 지원하는 라이브러리이다.”[참고문헌 -1번 : 인용] 따라서 통신이 필요하고, 소켓 프로그래밍이 내장되어 있다. Stub과 skeleton을 따로 만들거나 가져오지 않고, client와 server 각각의 프로젝트에서 class를 작성할 때, extends로 Remote를 상속받는데 이안에 이미 존재한다. Socket programming에서 작성한 맵 대신 인터페이스를 공유해 어떠한 함수이름이고, 파라미터인지를 파악한다. 또한, 객체이름은 registry를 이용해 서로 공유하고 있는 새로운 공간을 만든다. 이에 맵핑테이블 처럼 키-값 형태로 저장하고, bind를 통해 키값을 넣고 연결할 수 있다.

포트넘버와 IP주소는 따로 전달하지 않아도 된다. 우선 포트넘버는 따로 지정하지 않으면 1099 포트 넘버를 디폴트 값으로 사용한다. IP주소는 registry에 객체를 등록할 때, 서버의 IP를 자동으로 저장하기 때문이다.

2. 프로그램 설계 원칙 - SOLID

프로그램을 설계할 때에는, 관심사를 분리해야 한다. Domain과 technical로 나누는데, Domain은 비즈니스 로직이나 핵심 기능, 기술 등

을 처리하는 서비스이다. Technical은 기술적인 측면에서 도메인 즉, 기능과 관련이 없는 통신, 보안, 로깅, 캐싱, DB 연결 등 기술적인 측면, 수단으로 이용할 서비스를 의미한다.

도메인 서비스를 구현할 때, 가장 주의해야 하는 부분은 SOLID 원칙에 맞추어 코드를 짜는 것이다. S는 SRP, 단일 책임의 원칙을 의미한다. 하나의 클래스는 하나의 책임만 가져야 한다는 뜻이다. O는 OCP, 개방폐쇄원칙을 의미한다. 소프트웨어 엔티티는 확장에는 열려있어야하고, 변경에는 닫혀있어야한다. 즉, 새로운 기능을 추가할 때에, 기존의 코드를 변경하지 않고 확장해야한다. L은 LSP, 리스코프 치환 원칙을 의미한다. 자식 클래스는 항상 부모클래스를 사용할 수 있어야한다. 즉, 부모 대신 자식이 들어가도 되어야 한다. I는 ISP, 인터페이스의 분리 원칙을 의미한다. 한 인터페이스가 너무 많은 메서드를 포함하지 않아야 한다. DIP는 의존 역전 원칙으로, 고수준 모듈이든 저수준 모듈이든 추상화에 의존해야지 구체적인 구현에 의존하면 안된다.

참고문헌

1)" rmi 통신과 socket", chatgpt, 인용 : 1번째 줄