객체지향 프로그래밍 in 자바

자바는 객체지향 프로그래밍 언어이다. 객체지향 프로그래밍은 프로그래밍 패러다임의 하나이다. 패러다임의 의미는 사전에서 찾아보면 ‘어떤 한 시대 사람들의 견해나 사고를 근본적으로 규정하고 있는 테두리로서의 인식의 체계’라고 나와 있다. 세상을 보는 관점 같은 것이다. 프로그래밍 패러다임은 프로그램을 설계하는 관점을 나타낸다. 같은 프로그램을 만들더라도 패러다임에 따라 만드는 방식과 접근 방법이 달라진다.

객체지향 프로그래밍은 데이터와 데이터를 조작하는 동작을 객체라는 단위로 묶어 프로그램을 구성하는 방식이다. 객체 간에는 서로 메시지를 전달해 상호작용하며 흐름을 제어한다. 이 방식으로 프로그램을 작성하면 반복 코드가 줄어 재사용성이 높아지고 간결해진다. 또한 프로그램이 유연하고 확장하기 쉬우며 유지 보수가 쉽다.

객체지향 프로그래밍의 특징은 캡슐화, 상속성, 다형성, 추상화이다.

1. 캡슐화 : 프로그램에 필요한 자원을 객체로 묶어 감추는 것. 객체의 데이터를 보호하고 외부의 잘못된 접근을 막을 수 있다.
2. 상속성 : 한 클래스의 데이터와 기능을 다른 클래스가 이어받는 것을 말한다. 상속을 통해 클래스끼리 부모 자식 관계라 생긴다.
3. 다형성 : 동작이나 형태가 다른 기능을 하나의 이름으로 사용할 수 있게 하는 것을 다형성이라 한다. 자바에서는 오버라이딩과 오버로딩 등을 통해 다형성을 지원한다.
4. 추상화 : 객체마다 데이터와 동작을 구체적으로 정의하지 않고 객체들의 공통적인 데이터와 동작만 추출해 선언해두는 것이다. 이렇게 선언한 클래스를 추상화 클래스라고 한다.