**왜 인터페이스, 추상 클래스, 상속 등을 쓰나요?**

자바는 객체지향 언어로써 프로그래머가 클래스를 설계하고 이를 실제로 구현하면서 객체와 객체를 연결하면서 프로그램을 구현합니다. 그 중에서 다형성의 활용을 통해서 프로그래머는 자신의 프로그램을 더 손쉽게 관리할 수 있고 유지보수성을 높일 수 있습니다. 이러한 이유로 인터페이스, 추상클래스, 상속 등을 통해 다형성을 활용합니다.

**추상클래스란?**

여러 가지의 클래스의 비슷한 필드나 메소드를 공통적으로 추출한 클래스

**왜 사용하나요?**

1. 공통적인 필드와 메소드를 통일함
2. 실체클래스 구현 시 시간이 절약된다.
3. 규격에 맞게 실체클래스를 구현할 수 있다.

**사용방법?**

public abstract class 클래스명{

// 필드

// 생성자

// 메소드

// 추상메소드

}

추상메소드도 동일한 방식

추상클래스에 추상메소드를 쓰면 상속받은 실체클래스들은 무조건 추상메소드를 재정의 해야한다.

**인터페이스란?**

동일한 목적하에 동일한 기능을 보장하게 하기 위한 방법으로 하위 클래스에 특정한 메소드가 있도록 강제하는 기능.

**왜 사용하나요?**

클래스를 구현할 때 동일한 규격으로 구현할 수 있게 하기 위함.

**사용방법?**

public **interface** 인터페이스명{

// 상수

// 추상 메소드

// 디폴트 메소드

// 정적 메소드

}

상수 : 인터페이스에서 값을 정해줘서 바꾸지 말고 제공하는 값만 참조하게 함

추상 메소드 : 메소드의 형식만 주고 오버라이딩해서 재구현하게 함

디폴트 메소드 : 인터페이스에서 제공하지만 자유롭게 구현할 수 있음

정적 메소드 : 인터페이스에서 제공해주는 것으로 무조건 사용

만약 추상메소드를 인터페이스에서 추가한다면, 이를 implements한 모든 클래스에서 강제적으로 추상메소드를 구현해야하고 구현하지 않을시 전부 에러가 난다.

디폴트 메소드나 정적 메소드를 실제로 구현하지 않아도 사용할 수 있음

정적 메소드는 오버라이딩이 안됨.

**두 개를 어떨 때 사용해야하나요?**

추상 클래스는 상속을 사용하기 때문에 한 클래스에서 이미 상속을 받고있다면 사용할 수 없다. 그래서 이럴때는 인터페이스를 사용해야한다. 인터페이스는 다중상속이 가능하기 때문이다.

**두 개의 차이점은 무엇인가요?**

둘다 추상 메소드를 사용하는 점에서 헷갈릴 수 있으나 추상 클래스는 상속을 이용하는데 상속은 슈퍼클래스의 기능을 이용하거나 확장하기 위해 사용되는 반면 인터페이스는 인터페이스를 구현한 객체들에 대해서 동일한 동작을 약속하기 위해서 존재하는 점이 다르다.