

과제 1-1. Java 입문

제로베이스_대학직무부트캠프_백엔드(1)

황제현

1. 상속 (Inheritance)

1.1 개념

상속은 기존 클래스를 기반으로 새로운 클래스를 정의하는 방식으로, 코드 재사용성과 유지보수성을 높여준다. Java에서는 extends 키워드로 부모 클래스의 속성과 메서드를 자식 클래스가 물려받는다.

코드 예시

```
class Car {  
    void drive() {  
        System.out.println("차가 달립니다.");  
    }  
}  
  
class ElectricCar extends Car {  
    void charge() {  
        System.out.println("전기를 충전합니다.");  
    }  
}
```

1.2 주요 특징

- 코드 중복 제거
- 유지보수 및 확장 용이
- 자식 클래스는 부모 클래스의 모든 public 및 protected 멤버를 상속받는다
- Java는 **단일 상속**만 지원

2. 다형성 (Polymorphism)

2.1 개념

다형성은 부모 타입의 참조 변수로 자식 객체를 참조하고, 실제 객체의 메서드를 실행하는 개념이다.

코드 예시

```
class GasolineCar extends Car {  
    @Override  
    void drive() {  
        System.out.println("가솔린 차량이 달립니다.");  
    }  
}  
  
class ElectricCar extends Car {  
    @Override  
    void drive() {  
        System.out.println("전기차가 조용히 달립니다.");  
    }  
}  
  
public class Main {  
    public static void main(String[] args) {  
        Car car1 = new GasolineCar();  
        Car car2 = new ElectricCar();  
  
        car1.drive(); // 출력: 가솔린 차량이 달립니다.  
        car2.drive(); // 출력: 전기차가 조용히 달립니다.  
    }  
}
```

2.2 주요 개념

- **업캐스팅(Uncasting):** 자식 타입 객체는 부모 타입 참조변수에 저장할 수 있다.
 - **Ex.** Car car = new ElectricCar();
- **오버라이딩(Overriding):** 자식 클래스에서 메서드를 재정의.
- **동적 바인딩:** 실행 시점에 실제 객체의 메서드가 호출됨

2.3 장점

- 공통 인터페이스로 다양한 구현을 처리 가능
- 확장에 유리한 구조

3. 추상 클래스 (Abstract Class)

3.1 개념

추상 클래스를 사용하면 공통 기능을 정의하고, 특정 기능은 자식 클래스에서 반드시 구현하도록 강제할 수 있다.

코드 예시

```
abstract class Car {  
    abstract void drive(); // 추상 메서드  
  
    void stop() {  
        System.out.println("차가 멈춥니다.");  
    }  
}  
  
class ElectricCar extends Car {  
    @Override  
    void drive() {  
        System.out.println("전기차가 조용히 주행합니다.");  
    }  
}  
  
class GasolineCar extends Car {  
    @Override  
    void drive() {  
        System.out.println("가솔린 차가 엔진 소리를 내며 주행합니다.");  
    }  
}
```

3.2 주요 특징

- 객체 생성 불가: `new Car()`는 오류
- 추상 메서드가 하나라도 있으면 클래스도 추상 클래스여야 함
- 공통 로직은 구현, 개별 로직은 추상 메서드로 위임