



24년 03월 2주차 회고록

- 회의 날짜: 2024.03.12 (화)
- 회의 장소: 학교 강의실
- 참석자: 김명준, 이동현, 전경섭, 장현서

🧐 [이번 스터디 공부한 내용]

Java 변수와 타입, 메소드의 종류 및 구분
Object 클래스의 메소드 종류
상속할 때 유의사항
인터페이스와 추상 클래스, 클래스의 차이

🧐 [기억나는 부분]

- 🧐 김명준

Java 변수와 타입

기본형: 실제 값을 저장 (숫자형, 문자형, 불린형)
참조형: 메모리 주소를 저장 (기본형 제외 나머지)

인스턴스 변수: 클래스 안, 메소드 밖에 선언
-> 인스턴스마다 각각 다른 값을 가짐
클래스 변수: 인스턴스 변수 앞에 static 키워드
-> 모든 인스턴스가 같은 값을 가짐
지역 변수: 메소드 안에서 선언
매개 변수: 메소드 소괄호 안에 선언

메소드의 종류 및 구분

인스턴스 메소드: 객체를 생성해야 사용 가능
클래스 메소드: 객체 생성 없이 사용 가능
-> 인스턴스 멤버 사용 불가

Object 클래스의 메소드 종류

toString(): 객체의 정보를 출력
equals(): 객체의 주소를 비교
hashCode(): 객체의 hashCode를 반환
※ equals()를 오버라이딩했으면 hashCode()도 오버라이딩

상속할 때 유의사항

부모에 기본 생성자 만들기
-> 모든 생성자의 첫 줄에는 super()가 생략되어있음
접근 제어자의 범위가 좁아지면 안됨
-> 부모가 public이면 public으로 선언해야 함

인터페이스와 추상 클래스, 클래스의 차이

인터페이스:

1. 상수와 추상 메소드만 가질 수 있음(default 메소드도 추가)
2. 다중 상속이 가능하며, 인터페이스를 구현하는 클래스가 다중 구현 가능

추상 클래스:

1. 추상 메소드가 하나 이상이면 됨
2. 단일 상속만 가능

클래스:

1. 추상 메소드가 하나라도 있으면 안됨
2. 단일 상속만 가능

• 🧑 이동현

java변수와 타입

기본형: 실제 값을 저장하는 곳(int, double, char, boolean 등등)

참조형: 값의 주소를 저장(String, 배열 등등)

메소드의 종류 및 구분

인스턴스 메소드 클래스 메소드의 구분하려면,

static이 붙어 있으면 클래스 메소드이다.

또한, 선언 방식도 다른데 인스턴스는 new 객체()를 통해

선언 클래스는 직접 클래스명.메소드로 접근한다.

추상메소드를 할 때는 접근 제어자 "abstract" 자료형 메소드명을 해준다

Object 클래스의 메소드 종류

toString: 객체를 문자열로 형 변환하는데 사용하는데

오버라이딩을 하지 않고 사용하면 해시코드를 반환하고 하면 문자열로 반환

hashCode: 객체의 해시코드 리턴

equals(Object obj): 문자열이 같은지 비교하는 메소드(참조를 비교)

상속할 때 주의할 점

부모의 접근 제어자가 자식의 접근 제어자보다 넓으면 안됨(부모 < 자식)

super()를 사용할 시 맨 앞에 사용

부모 클래스에 선언되어 있는 메소드와 자식 클래스에서 선언된 메소드가 동일하다면

자식 클래스의 메소드만 실행(메소드 overriding) 덮어 씌움

오버라이딩을 할 시 동일한 리턴 타입

• 🧑 전경섭

Java 자료형

- 기본형: 기본적으로 있는 타입(int, double, boolean), 나누는 기준은 연산자
- 참조형: 주소값을 저장하는 타입 (Object를 상속함)

Method의 종류 및 구분

- 모든 클래스의 부모 클래스 Object
- 클래스 메소드: static이 붙어있는 메소드 (객체 생성 없이 호출 가능)
- 인스턴스 메소드: static이 없는 메소드

ObjectClass의 Method 종류

- 클래스는 기본적으로 아무런 상속을 받지 않으면 `java.lang.Object` 클래스를 상속 받는다.
- `toString()`: 해시코드 출력,
오버라이드시 `String`객체의 경우 저장하고 있는 문자열 출력
- `hashCode()`: 객체에 대한 해시 코드(hash code) 값을 리턴한다.
오버라이드시 같은 해시코드 반환
- `equals()`: 연산자 `==` 와 같은 기능
오버라이드시 해시코드값을 비교

상속할 때 유의사항

- 자식 클래스의 생성자가 호출되면,
자동으로 부모 클래스의 매개 변수 없는 생성자가 실행된다.
(이게 `super()`인가..?)
- 자식 클래스에서는 `public`, `protected`로 선언된 인스턴스 및 클래스변수와 메소드를 사용할 수 있다.

인터페이스와 추상 클래스, 클래스의 차이

- 인터페이스: `default`메소드와 `abstract`, `static final`만 사용 가능
다중구현, 다중상속 가능
- 추상 클래스: 단일상속만 가능
클래스간의 연관 관계를 구축
추상 메소드가 있으면 추상클래스
- 클래스: 단일상속만 가능

• 🤔 장현서

Java 변수와 타입

- 기본 (정수 타입) `byte`, `int`, `char`, `short`, `long`
(실수 타입) `float`, `double`
(논리 타입) `boolean`
- 참조 -> 그 외

메소드의 종류 및 구분

- 인스턴스 메소드(instance method)
인스턴스를 생성해야 호출 가능
- 클래스 메소드(static method)
앞에 `static`이 붙는다
인스턴스를 생성하지 않아도 호출 가능
인스턴스 변수를 사용할 수 없다

Object 클래스의 메소드 종류

- `toString()` : 오버라이드 `x` -> 해시코드 출력
오버라이드 `0` -> `String`객체의 경우, 저장하고 있는 문자열 출력
- `hashCode()` : 생성자가 동일한 객체를 선언 후 `equals()` 사용
오버라이드 `x` -> 서로 다른 해시코드 반환
오버라이드 `0` -> 같은 해시코드 반환
- `equals()` : 기본동작 -> `'=='` 연산 결과 출력
오버라이드 `0`
-> 물리적으로 다른 메모리에 위치해도, 논리적으로 같음을 구현

상속할 때 유의사항

- 부모 클래스의 생성자, 초기화 블록은 상속받지 못 한다
(필드, 메서드만 상속 가능)
- 상속관계는 두 클래스 간의 영향력이 커지므로,

변경할 때 같이 영향받는 경우가 있어,
부모 클래스를 자식 클래스에서 구체화할 때 상속을 사용하지 않는 경우에 상속 지양

인터페이스와 추상 클래스, 클래스 차이

- 인터페이스 : 'interface' 키워드 사용 -> 상수와 추상 메서드의 집합
 - 추상 클래스보다 추상화 정도가 높다
 - 멤버 변수, 멤버 메서드 선언 가능 / 접근 제한자 설정 불가능
 - 다중 상속 가능
- 추상클래스 : 'abstract' 키워드 사용 -> 하나 이상의 추상 메소드를 갖는다
 - 상속을 위한 클래스이기에 new 키워드를 통해 객체 생성 불가능
 - 다중 상속 불가능
- 클래스 : 별도 인스턴스를 만들지 않고, static 함수를 사용해 인스턴스 생성 가능