**Web개발\_Day57\_노트정리**

**객체지향의 의미**

ㆍ정의 : 객체지향언어란 상속, 캡슐화, 다형성, 추상화등의 “코드 재사용성” 및 “유지보수”에 용이함

ㆍ특징 : 코드의 캡슐화, 상속, 다형성, 추상화를 유지한다.

ㆍ중요 포인트 : 코드의 의존성을 줄이고, 결합도를 낮추는 것

- 하나의 페이지에서 모든 기능, 변수, 개념등을 정의하지 않고 분류별로 나눠서 작성한다. = 해당 페이지의 의존성을 줄인다.

‘ㆍ장점 : 요구하는 기능을 수행하면서도 변경하지 쉬운 (=유지/보수/관리가 매우 용이한) 코드 작성이 가능하다.

1. 상속 = 재사용성 높임
2. 클래스를 객체화 하여 생상성+재사용성을 높임
3. 캡슐화 = 유지보수가 용이

**상속**

ㆍ부모클래스의 기능, 맴버변수등을 자식클래스가 물려받아, 그대로 선언 없이 사용하는 기법

ㆍ장점 : 객체간의 ‘공통적인 기능’을 관리하기 용이

- 코드간의 재사용성이 높으므로, 유지/보수에 용이

**캡슐화**

ㆍ데이터와 코드의 형태를 외부로부터 알 수 없게 하고, 데이터의 구조와 역할, 기능을 하나의 캡슐 형태로 만드는 것

**다형성**

ㆍ오버로드, Overload : 같은 이름의 메소드를 각각의 기능을 따로 구현 가능하다.

- 같은 이름의 메소드를 서로 다른 매개변수/인자를 통하여 구분 🡪 실행 한다.

ㆍ오버라이드, Override : 부모의 메소드를 자식클래스에서 자신의 입맛에 맞게 ‘재정의’하는 기능

- 상속을 받는가는 것을 전제 조건으로 한다.

- 부모의 메소드를 같은 이름으로, 다른 기능을 구현한다.

> 자식클래스 객체에서 메소드를 부른다면 부모클래스에 있는 메소드 접근은 힘들다.

- super 키워드로 부모 클래스의 메소드 또는 부모클래스 그 자체를 불러오는 구현이 가능하다.

ㆍ장점 : 서로 다른 기능을, 같은 이름으로 처리함으로써, 프로그램의 ‘일관성’을 유지한다.

ㆍ특이점 : @Override, 오버라이의 경우, 어노테이션을 작성해주면 오버라이드 조건을 맞춰주는 기능이 있다.

**추상화**

ㆍ추상화 : 메소드의 정의 단계를 상속시에 기능을 정의한다.

ㆍ추상화의 장점 : 재정의 과정의 오류가 없다.

- 오버라이드 처리시에 메소드를 재정의 하는 과정에서 개발자가 잘못 정의 하는 경우를 방지한다.

ㆍ인터페이스 : 추상메소드로만 이루어진 클래스를 interface, 인터페이스 라고 명시한다.

- 추상화와 다르게 추상메소드 외의 메소드를 허용하지 않는다.

ㆍ인터페이스의 특징 :

1. implements로 다중 상속이 가능
2. 인터페이스 상속시에 추상메소드의 구현을 ‘강제’한다.

**제어문, 조건문**

ㆍif문 : 하나의 코드를 수행함에 있어서, 조건에 따라 “각기 다른 코드를 수행 하고 싶을 때” 사용한다.

- 사용목적 : 코드를 작성하다가 ‘분기점’ 이 필요할 경우에 사용

ㆍif~else if~else문 : if문과 동일하나 조건을 추가적으로 작성할 수 있다.

ㆍswitch문 : if~else문과 동일하나, 분기점의 수가 정해진 경우 대체로 쓰인다.

- 계산기의 +-\*/%나, 1~10의 정해진 수의 경우 등등…

**제어문, 반복문**

ㆍfor문 : 같은 코드를 반복시키되, 반복의 “횟수”가 정해져 있을 때 사용

ㆍwhile문 : 같은 코드를 반복시키되, 반복의 “조건”이 필요할 때 사용

- 무한 루프용도로도 쓰인다.

ㆍ사용장점 : 동일한 기능을 하는 코드를 여러 번 작성할 필요 없어, 사용성이 높다.

- 예 ) 사람이 직접 처리하기 힘든 반복적인 작업

**컬랙션**

ㆍ정의 : 다수의 데이터를 쉽게 처리할 수 있는 표준화된 방법을 제공하는 클래스들

ㆍ특징 : 기존의 ‘배열’형태에서 좀더 다양한 ‘형태’와 다양한 ‘데이터’의 저장을 구현할 수 있다.

ㆍ종류

1. List : ArrayList<타입>

: Vector

1. Map : HashMap

: SortedMap

ㆍ특징 : ArrayList는 중복허용, 순서있음, 인덱싱

: HashMap은 중복미허용, 순서없음, 키값관리

**시험 정리**

ㆍ상속의 범위 : 부모의 접근제어자의 범위는 자식클래스 범위보다 작아야 한다.

ㆍ문자열 연산 vs 산술연산

- “값” + 정수값 + 정수값을 출력할 경우의 결과는?

1. “값정수값” + 정수값

2. “값정수값정수값”이 출력된다.

- 정수값의 연산을 산술연산으로 실행하고 싶다면, 뒤에 있는 두 정수값을 먼저 ( ) 로 묶어서, 먼저 실행시켜주어야 한다.