**Web개발\_Day12\_노트정리**

**수퍼, super**

ㆍ자식클래스에서 부모를 가리키는데 사용되는 참조 변수이다.

* 맴버 변수와 지역변수의 이름이 같을 때 this를 붙여서 구별하듯이, 상속받은 맴버와 자기자신의 맴버 이름이 같은 경우, super로 구별할 수 있다.

ㆍ맴버변수(int x)가 부모클래스인 parent에도 있고, 자손클래스인 child에도 있다면, super.x와 this.x가 서로 다른 값을 참조하게 된다.

* This.x 는 자식클래스에 선언된 맴버변수 x를 의미
* Super.x 는 부모클래스에 선언된 맴버변수 x를 의미

ㆍ생성자를 공부할 때 this를 사용해서 같은 이름의 생성자를 호출하였다.

* 역시, 생성자에서도 super()를 사용해서 부모클래스의 생성자를 호출하는데 사용된다.

ㆍ자식클래스에서 객체를 생성하면서 자식의 맴버와 부모의 맴버가 모두 합쳐진 하나의 객체가 생성된다. 그래서, 자식클래스의 인스턴스(객체)가 부모클래스의 맴버들을 사용할 수 있다.

* 이때 부모클래스의 맴버의 “초기화 작업” 이 수행되어야 한다. 따라서 자손클래스에서의 생성자에서도 부모클래스의 생성자를 호출해주어야 한다.
* 모든 클래스에서 생성자 첫줄에 super()를 호출해주어야 한다. 그렇지 않으면 자바가 자동적으로 super();를 생성한다.
* 기본생성자가 아닌 경우, 직접 부모클래스의 생성자를 super()를 사용하여 정의해주어야 한다.
* Default생성자를 추가해주거나, 정의된 생성자에서 super()로 부모생성자를 호출
* 오버로드된 생성자들은 모두 부모생성자를 호출해주어야 한다.

**접근제어자, public**

ㆍ외부에서 접근하지 못하도록 제한하는 역할을 한다.

ㆍ접근제어자는 클래스 , 맴버변수, 메소드, 생성자에 쓰인다.

ㆍ접근제어자 종류:

* 1. Private : 같은 클래스 내에서만 접근가능
* 2. (default) : 같은 패키지 내에서만 접근가능(디렉토리)
* 명시하지 않으면 default로 자동할당
* 3. Protected : 같은 패키지내에서도, 다른 패키지의 자손클래스에서는 접근가능
* 4. Public : 접근 제한이 없음

ㆍ사용처 : 클래스는 public, default 둘중 하나만 사용한다

* 메소드, 맴버변수는 public, protected, (default), private 다 사용가능
* 지역변수는 없음

ㆍ클래스나 맴버, 주로 맴버에 접근제어자를 사용하는 이유는, 클래스 내부의 선언된 데이터를 보호하기 위해서이다.

* 데이터가 유효한 값을 유지 할 수 있도록 비밀번호와 같은 중요한 데이터를 외부에서 함부로 변경하지 못하게 외부로부터의 접근을 제한하는 역할을 해준다.

ㆍ**캡슐화** : 외부에서 접근할 필요가 없는 맴버들은 private으로 지정하여 외부에 노출을 막음으로 복잡성을 줄일 수 있다.