**Web개발\_Day07\_노트정리**

**2차원 배열,**

ㆍ다차원 배열 정의 : 2차원 이상의 배열,(다차원 배열) 메모리의 용량이 허용하는 한 차원의 제한이 없지만 주로, 1,2차원 배열이 사용된다.

ㆍ2차원 배열 선언 : 기본적으로 1차원 배열의 선언과 크게 다르지 않다.

* Int[][] 배열명;
* Int[][] 배열명 = new int[행길이][열길이];
* Int[][] 열명 = new int[2][3]{ {1,2,3}, {4,5,6} }
* Int[][] 배열명 = { {값1, 값2, 값3, …} , { 값1, 값2, 값3, …}, { 값1, 값2, 값3, …} }
* 초기값을 입력하지 않은 경우, 0으로 설정된다.

ㆍ2차원 배열의 인덱스

* 2차원 배열은 행(row)과 열(column)으로 구성되어 있기 때문에, index도 행,열 각각 하나씩 존재한다.
* 접근방법: 배열명[0][0], 배열명[0][1], … 배열명[n][n]

ㆍ2차원 배열의 삽입

* 배열명[0][0] = 데이터값;
* 이미 값이 존재한다면, 새로운 값으로 수정된다.

ㆍ2차원배열의 데이터 뽑기

* For(int [] temp: 배열명){

For(int I : temp){

Int temp\_data = I;

}

}

* For(변수:배열)을 하여 반복하듯, “배열: 2차원배열” 을 하여 반복문을 돌리는 것도 가능하다.

**객체지향언어, OPP**

ㆍJava가 1995년 등장하면서 1990년대 말에 인터넷과 함께 크게 유행하면서 객체지향언어는 현재까지 프로그래밍 언어의 주류로 자리잡았다.

ㆍ객체지향 언어의 특징

* 코드의 재사용성이 높다
* 코드의 관리가 용이하다
* 신뢰성이 높은 프로그래밍을 가능하게 한다.

ㆍ객체지향 언어의 가장 큰 장점은 ‘코드의 재사용성’이 높고 ‘유지보수’가 용이하다.

**클래스와 객체**

ㆍ클래스란, 객체를 정의해 놓은 설계도

* “틀”이라고도 정의할 수 있다.
* 객체는 클래스에 정의된 대로 생성된다.
* 클래스의 용도: 객체를 생성하는데에 사용된다.

ㆍJDK(Java Development Kit)는 프로그래밍을 위한 유용한 클래스를 기본적으로 제공한다

* Scanner, Random, …

ㆍ클래스의 구성요소 ( 속성과 기능 )

* 속성(property) : (맴버)변수 등등
* 기능(function) : 특정 동작을 실행
* Function vs Method
* Function은 클래스 밖에서 선언된 기능함수
* Method는 클래스 내부에서 선언된 기능함수
  + Java는 클래스 내부에서 모든 것이 이뤄지므로 method만 존재한다.
* 예시 ) TV
* 속성 : 크기, 길이, 색상, 채널, 브랜드, …
* 기능 : 켜기, 끄기, 소리높이기, 채널돌리기, …

ㆍ 클래스의 선언

* class 클래스명 {

변수;

void 메소드명() { … }

}

* “class”라는 예약어와 클래스명을 명시하고, { … } 블록을 구성한다.
* 변수와 메소드는 사용자 지정에 따라 달라진다. (생략 가능)

ㆍ객체란, 클래스를 통해서 실제 만들어진 것

* 객체의 용도 : 객체의 기능과 속성에 따라 각기 다르다.

ㆍ객체의 선언

* 선언 : 클래스명 객체명;
* 할당 : 객체명 = new 클래스명();
* 선언 및 할당 : 클래스명 객체명 = new 클래스명();

ㆍ객체의 사용

* 객체 안에 포함된 변수나 메소드는 점(.) 으로 연결하여 접근한다.

1. 값 = 객체명.변수; (값 복사)
2. 객체명.변수 = 값; (값 대입/수정)
3. 객체명.메소드명; (메소드호출)
4. 객체명.메소드명(매개변수1, 매개변수2, …); (매개변수 전달하는 메소드)

* 하나의 클래스로부터 생성된 여러 개의 인스턴스들은 각각의 속성값을 따로 자신의 메모리안에 저장한다.
* 같은 속성명이라도 객체마다 따로 값을 저장한다.

ㆍ객체와 인스턴스

* 어떤 클래스로부터 만들어진 객체를 그 클래스의 인스턴스라고 한다.
* 인스턴스는 클래스와의 관계를 얘기할 때 사용하는 단어이다.
* “어떤 클래스로부터 만들어낸 객체”보다는 “어떤 클래스로부터 만들어진 인스턴스”가 바람직하다.
* 객체는 만들어진 그 데이터, 그 자체를 부를 때 사용한다.
* “Scanner객체는 입력받을 때 사용한다.”

ㆍ클래스마다 따로 파일을 만드는 것이 FM이다. 하나의 파일에 2개 이상의 클래스를 사용하는건 바람직하지 않다.