**Web개발\_Day04\_노트정리**

**제어문, Control Statement**

ㆍ반복문의 종류 : for(), while(), do~while() 등이 있다.

ㆍ사용자가 처리하기 부담스러운 반복작업을 처리하기에 용이하다.

* for(), while() : 조건에 따라 한번도 실행되지 않을 수도 있다.
* Do{}while(); : 조건에 상관없이 무조건 최소한 한번 실행되는 것을 보장한다.

ㆍ각 loop의 사용차이 :

* for() 는 반복 “횟수” 기반으로 사용한다.
* While() 은 반복 “조건” 기반으로 사용하며, 횟수가 불분명할 때 사용하기 용이하다.
* 두 loop는 서로 유사하여 서로 변환하여 구현이 가능하다.

**반복문1, for()**

ㆍfor( 초기값, 끝값, 증감치){

… 실행구문

}

ㆍ시작값 : int I = 0 ; (int i = 0, j = 0;으로도 사용가능)

ㆍ끝값 : I <= 100;

ㆍ증감치 : i++, i--, i+=2, i+=3 등등으로 사용가능

ㆍstart, stop, step 모두 생략가능하다.

* for( ; ; ) : 무한반복의 기능을한다.
* 이 경우 while을 좀더 많이 사용한다.

ㆍfor문에 사용되는 변수의 수가 적은 것이 효율적이고 간단하므로, 변수들의 관계를 잘 파악하여 불필요한 변수의 사용을 줄이는 것이 좋다.

ㆍ이중for()문 : for()의 실행블럭 안에 또 다른 for()문이 들어가는 경우

* O(n^2) 의 실행시간이 걸린다.

ㆍ고급 for()

* For(변수명 : 배열) { … }
* 일반 for()문과는 달리, 배열의 요소를 하나씩 읽어오는 용도로만 사용된다.
* 변수의 타입은 배열요소의 타입과 일치해야한다.
* 매 반복마다 변수에 배열요소를 담는다.

**반복문2, while()**

ㆍfor()문에 비해 구조가 간단하다.

ㆍfor()문과 while()문은 항상 서로 변환구현이 가능

ㆍ반복횟수가 정확하지 않을 때 주로 사용하며, 무한루프때도 자주 사용한다.

ㆍwhile ( 조건식 ) {

…조건이 true일 때 실현될 구문;

}