

문자열 토큰(Token) 및 파싱(Parsing)

문자열 토큰 (Token)

- "08 : 45" "11 : 24"의 문자열이 있을 때 : (콜론)을 기준으로 문자열을 나누면
- 구분자를 기준으로 나누어진 문자열 조각 08, 45, 11, 24는 토큰 (Token)
- : (콜론)은 구분자 (Delimiter)

StringTokenizer Class

Constructor

- `public StringTokenizer(String str, String delim)`
- `String delim`이 문자열인 이유는 구분자가 둘 이상이어도 된다는 뜻이다. (" \" 공백+역슬래쉬)

Method

<code>boolean hasMoreTokens()</code>	더 반환받을 토큰이 있는지 판단
<code>String nextToken()</code>	다음 번 토큰 반환

```
public class StringTokenize {
    public static void main(String[] args){
        String strData = "11:22:33:44:55";
        StringTokenizer st = new StringTokenizer(strData, ":");

        while(st.hasMoreTokens()){
            System.out.println(st.nextToken());
        }
    }
}
```

데이터 분리작업 - 파싱 (Parsing)

필자는 Android를 배우면서 파싱(Parsing)이란 단어를 처음 들었다. 파싱을 이용한 프로그래밍이 다수 등장하기 때문에 파싱에 대해서 알아보자.

- 파싱
 - 사전적 의미 : 문장단위의 문자열을 토큰(token)으로 분류하고 이를 구문 트리(parse tree)로 재구성하는 구문분석
 - 구문 분석 : 여러 문자로 나열된 문자열을 의미있는 문자끼리 구조화하고 분석하는 작업
 - 결론:
 - 파싱은 문자열 혹은 바이너리 데이터에서, 원하는 데이터들을 분리하고 구조화 시키는 분석작업
 - 구분하는 기준은 기본적으로 구분자가 될 수도 있고, 구분자는 사용자가 임의로 정의한다.
 - 문자열 파싱 : 문자열을 특정한 구분자로 분리 후 여러개의 문자열(토큰)으로 추출하는 과정
 - 데이터 파싱 : 데이터를 특정한 규칙에 따라 원하는 데이터로 변환하는 과정