

# 프로그래밍 언어 개론 00분반

201402447 한원희

## CuteParser

```
// 키워드가 FunctionNode에 해당
case ATOM_Q:
case CAR:
case CDR:
case COND:
case CONS:
case DEFINE:
case EQ_Q:
case LAMBDA:
case NOT:
case NULL_Q:
    if(tLexeme == null)
        System.out.println("???");
    return new FunctionNode(tType);
```

Pdf에 나와 있는 코드와 비슷하게 다른 타입의 노드 또한 바꿔 주었습니다.

## NodePrinter

```
else if(listNode.car() instanceof QuoteNode) { // `가 붙은 list일때

    while(listNode.cdr() != null){

        printNode(listNode.car());
        listNode = listNode.cdr();
    }
    return;
}

else{
    sb.append("( ");

    while(listNode.cdr() != null){

        printNode(listNode.car());
        listNode = listNode.cdr();
    }

    sb.append(") ");
}
```

리스트가 들어오는 PrintList 메소드입니다.

Car() 는 리스트 노드의 첫번째 아이템을 확인할 때 사용되는 함수입니다.

Cdr()은 리스트의 첫번째 아이템을 제거할 때 사용되는 함수입니다.

때문에 car로는 첫번째 인자를 확인할 수 있고 cdr로는 그 다음 노드로 이동할때 사용할 수 있습니다.

다시 코드로 돌아가면 어퍼스트로피는 하나의 새로운 리스트로 취급됩니다. 하지만 ( ) 를 붙이면 안되므로 기본 리스트와 구분해 주었습니다. while 문으로 안의 리스트를 순회하며 하나씩 printNode하는 방식으로 만들어 주었습니다.

```
private void printNode(Node node) {
    if (node == null)
        return;

    if (node instanceof ListNode) {
        ListNode ln = (ListNode) node;
        printList(ln);
    }
    else if (node instanceof QuoteNode) {
        sb.append(node);
    }
    else {
        sb.append "[" + node + " ] ";
    }
}
```

printNode 메소드입니다. ListNode일땐 printList로 안의 리스트를 순회하도록 만들었고, 만약 QuoteNode라면 바로 append하도록 만들었습니다.

```

public class QuoteNode implements Node {
    @Override
    public String toString() {
        return "\"";
    }
}

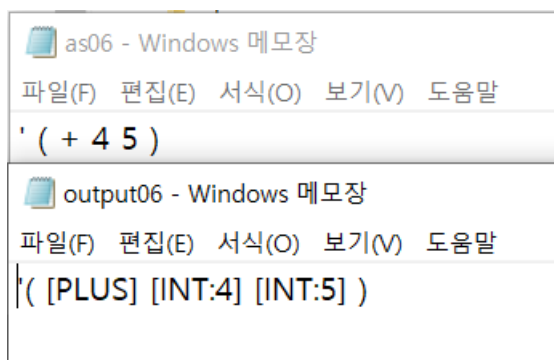
```

QuoteNode를 보면 어퍼스트로피를 출력해주는 것을 확인할 수 있습니다.

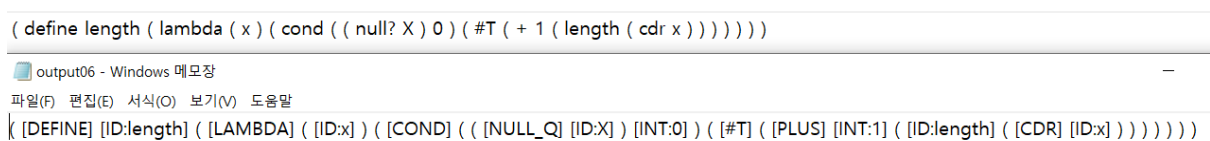
다른 노드들은 기존처럼 [ ] 안에 넣어주었습니다.

## 결과 화면

1.




2.



3.

```
( + ( length '( 2 3 16 ) ) -378 )
```

 output06 - Windows 메모장

파일(F) 편집(E) 서식(O) 보기(V) 도움말

```
( [PLUS] ( [ID:length] '( [INT:2] [INT:3] [INT:16] ) ) [INT:-378] )
```