

프로그래밍 언어 개론 00분반

201402447 한원희

```
public class CuteInterpreter {
    static HashMap<String,Node> table = new HashMap<String,Node>();
```

테이블에 저장하기 위해 hashmap을 선언해 줍니다. 전역 변수로 사용하기 위해 static을 붙여 줍니다. 이름을 저장할 string과 value를 저장할 Node를 설정해 줍니다.

```
private void insertTable(String id, Node value){
    table.put(id, value);
}

private Node lookupTable(String id){
    return table.get(id);
}
```

테이블을 사용하기 위해 새로운 메소드를 생성합니다. insertTable은 테이블에 삽입해주는 역할이고 lookupTable은 string을 받아 테이블을 확인하는 역할을 수행합니다.

```
case DEFINE:
    Node key = operand.car();
    Node value = runExpr(operand.cdr().car());
    insertTable(key.toString(), value);

    return null;
```

Define을 발견했을 때의 수행 작업입니다. (define c 1) 일때 define의 다음은 key가 되고 cdr의 car은 value가 됩니다.

- `(define c (- 5 2))` 와 같은 경우도 가능하다. 이때에는 `insertTable`에 삽입하기 전에 `runExpr`을 통해서 `(- 5 2)` 대신 결과값 3으로 바꾸어 저장한다.

Ppt를 보면 결과값 3으로 저장한다고 되어 있습니다. 때문에 value 는 `runExpr`을 통해 값을 저장해서 줍니다.

결과 화면

```
> ( define a 1 )
...
> ( + a 1 )
... 2
> ( define b ' ( 1 2 3 ) )
...
> ( car b )
... ' ( 1 2 3 )
> ( define c ( - 5 2 ) )
...
> ( + c 1 )
... 4
> ( define a 2 )
...
> ( + a 1 )
|... 3
>
```

```
> ( define a #T )
...
> ( car a )
|... # T
>
```