

코웨이

coding
wherever
easily

소프트웨어공학부

20132889 신종혁 20123043 박세용

20123087 이성준 20142773 최인정

I 목차

1. 프로젝트 소개

- 프로젝트 개요
- 추진 배경 및 필요성
- 프로젝트 목표 및 기대효과

2. 수행 내용

- Workflow
- Android (FE)
- Server
- 영상처리
- Interpreter

3. 시연 동영상

4. Q & A

1. 프로젝트 소개

프로젝트 개요

손으로 작성된 수도 코드를 Text 형태의 완성된
Java 코드로 변환하는 어플리케이션



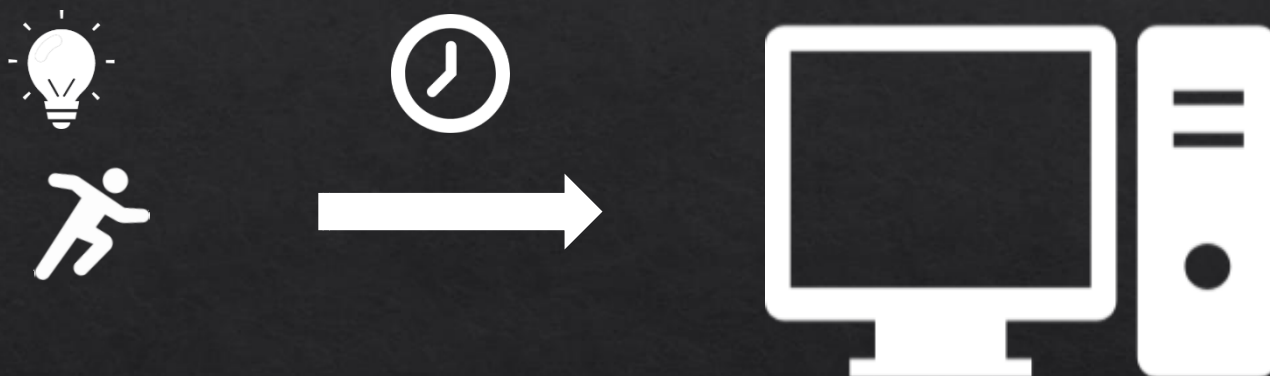
Pseudo Code



Text File
(Java)

■ 추진 배경 및 필요성

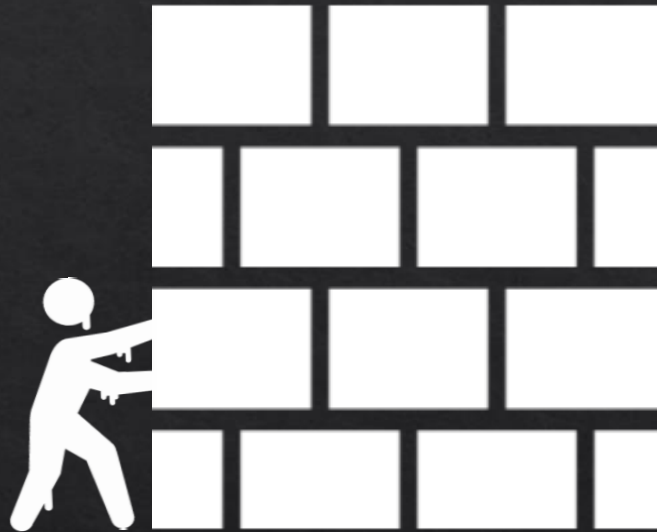
- 코드 작성의 시공간적 제약



개발환경이 갖추어지지 않은 상태에서 떠오른 아이디어?

■ 추진 배경 및 필요성

- 상이한 문법, 높은 진입장벽



문법 비숙련자의 아이디어 구현?

프로젝트 개요

- 목표

손으로 작성된 간단한 수도 코드를 사진으로 찍어 서버로 전송 후
java 문법이 적용된 text로 변환해주는 어플리케이션 개발



아이디어 저장



코딩의 시 공간적
제약 탈피



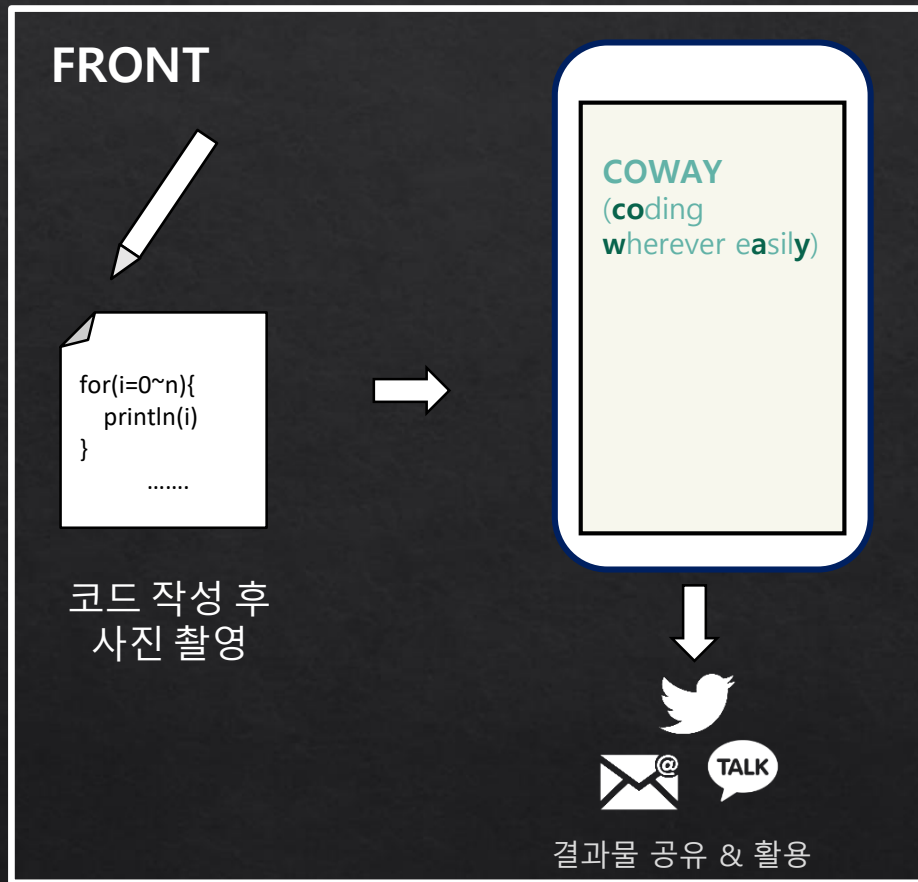
로직에 집중

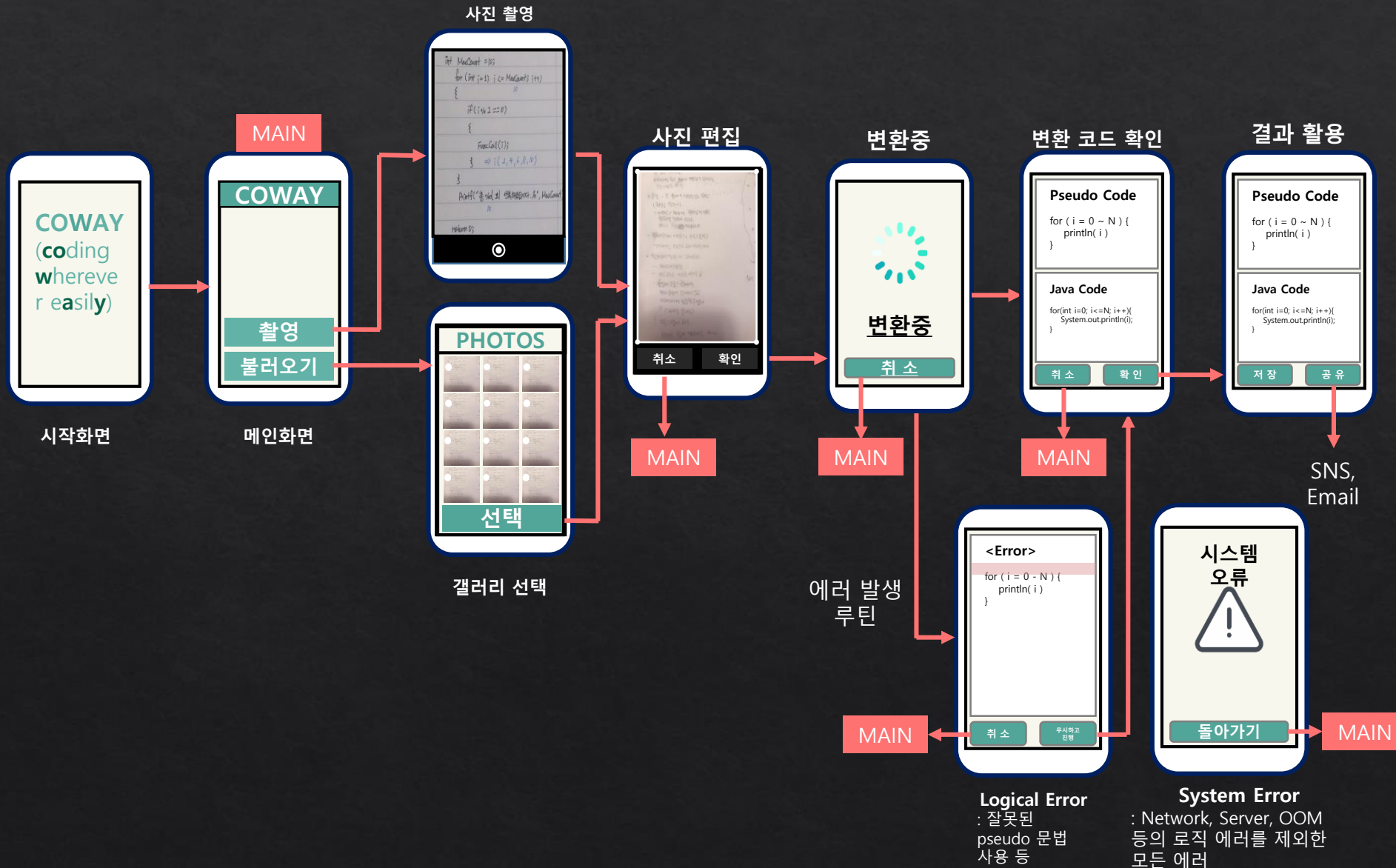


진입장벽 해소

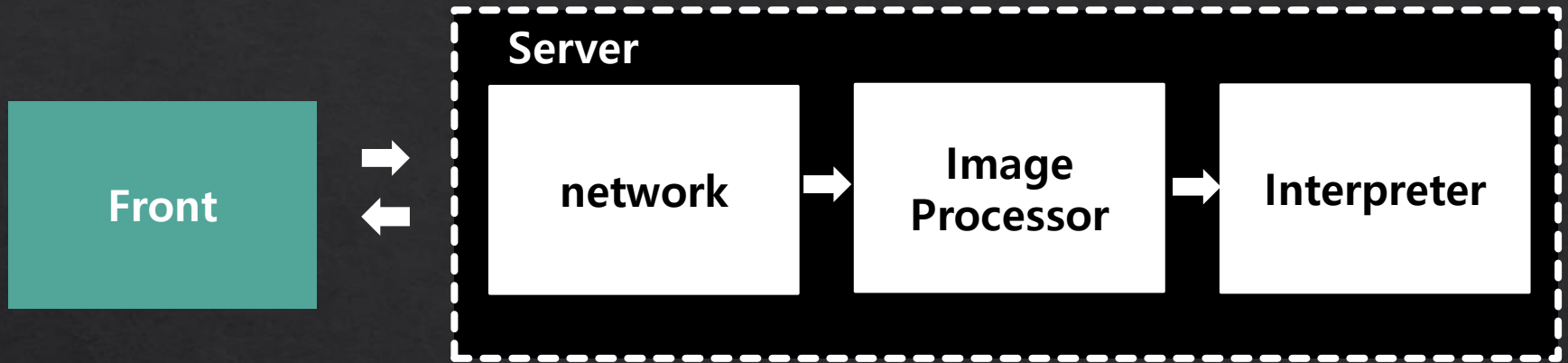
2. 수행 내용

Workflow





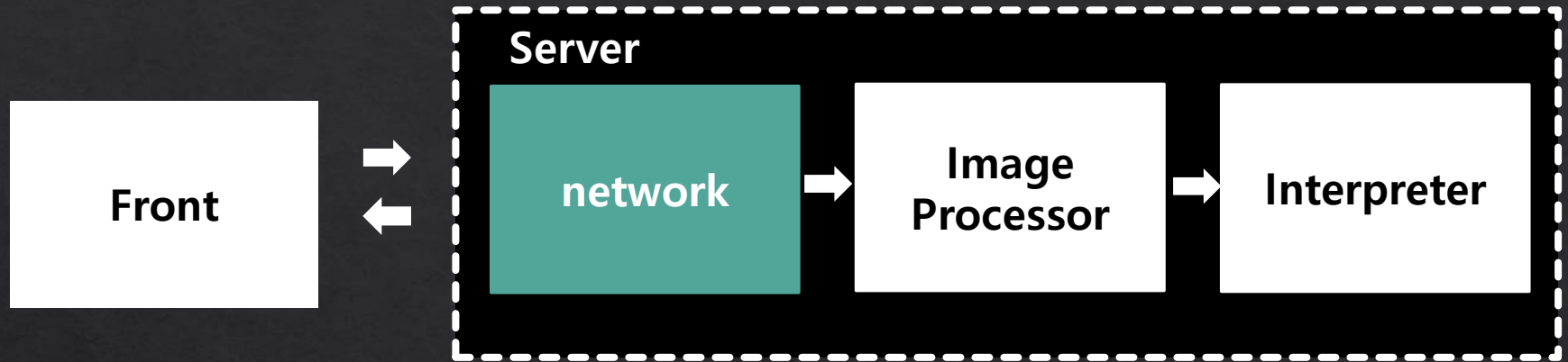
Front-End



Front

- 사용자의 촬영 혹은 기존 이미지를 서버로 전송
- 선택한 이미지의 편집
- 사진 전송 및 결과 수신을 위한 비동기 통신
- 변환 결과에 따른 화면 창 분리

Network



Network

- Socket 통신 기반
- FE(Android) – BE(Server) 간 통신 프로토콜 정립, 안정된 통신 구축
- 변환 결과에 따른 통신 시퀀스 구현
- 서버 내부 모듈 호출 방식으로 동작

Image Processor

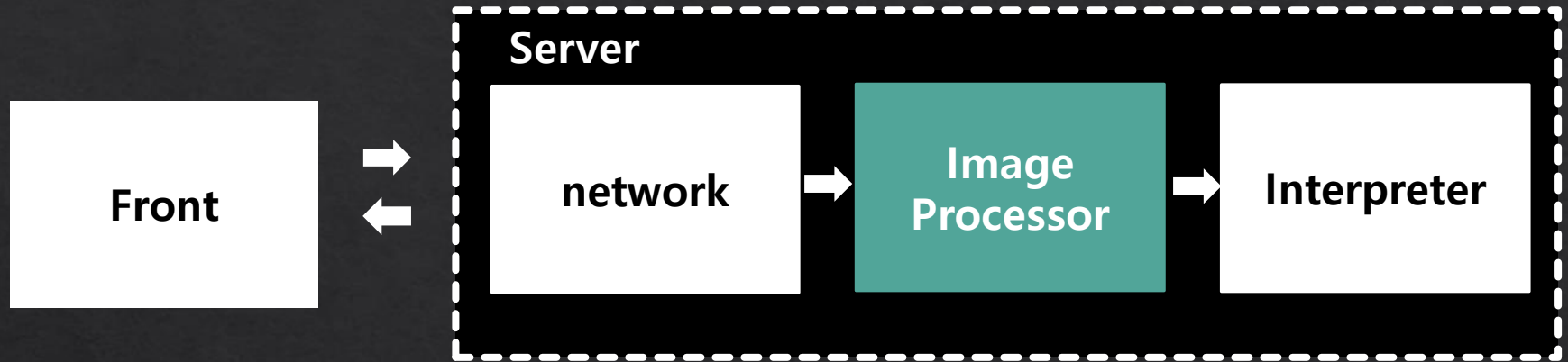
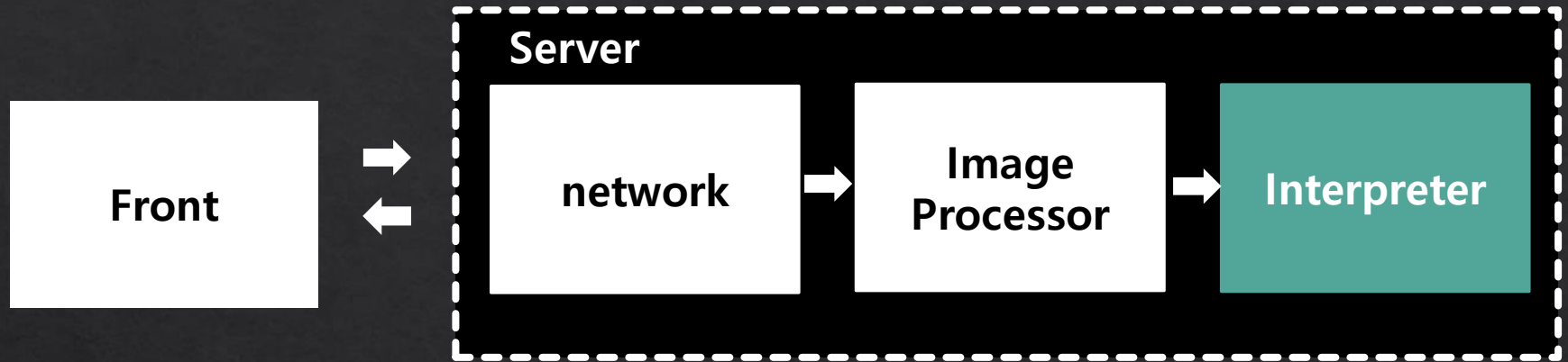


Image Processor

- 수기 이미지 이진화
- 수기 이미지 윤곽선 생성 및 처리
- 윤곽선 기반 캐릭터 추출
- Vision API를 통해 이미지에서 텍스트를 추출
- 후처리를 통한 원본 형식 그대로 텍스트를 정리

Interpreter



Interpreter

- For문 변환 규칙 구현
- print/ println 변환 규칙 구현
- 실행 스크립트 작성
- 변수 추론 변환 규칙 구현

인터프리티ng 알고리즘

변수 추론

- = 를 기준으로 좌변과 우변을 나눔
 - 좌변을 기준으로 신규 변수 선언인지, 기존 변수 재 정의인지 구분 (해시맵 사용)
 - 우변의 값을 기준으로 어떤 자료형인지를 정규식으로 추론
 - int, long, double, boolean, Object type 등
- 예시)

```
//신규변수_매칭_테스트  
a = 10  
b = '!'  
c = "Hello World!"  
d = 3.14  
e = new TestClass()
```



```
//신규변수_매칭_테스트  
int a = 10;  
char b = '!';  
String c = "Hello World!";  
double d = 3.14;  
TestClass e = new TestClass();
```



```
매칭 대상 라인: d = 3.14  
[변수추론]  
⇒ 좌변(d) 우변(3.14) ⇒ 정규식으로 추론된 우변의 타입 (double)  
⇒ 새로운 변수 (변수 선언 기록 테이블 = {a=int, b=char, c=String, d=double})  
⇒ 매칭 성공  
⇒ 매칭 종료 :: double d = 3.14;
```

- Object 타입의 객체 생성 시 Generics 매칭 부분이 불완전 하여 추후 구현 과제로 진행 예정

인터프리팅 알고리즘

예약어 매칭

- for 문 등의 변환해야 할 문법을 예약어라 지칭
- 예약어 마다 변환 코드를 담은 클래스를 생성 (화이트리스트)
- 변환 규칙
 - 수도 코드 한 라인을 받아 공백 제거 후, 영문자가 아닌 모든 문자를 기준으로 split
 - Split되어 생성된 배열을 첫번째 값 부터 지정한 예약어에 해당하는지 검사
 - 해당사항이 없으면 종료
 - 예약어라면 해당 클래스를 호출하여 변환 진행변수들 추출
 - 정규식 매칭을 하여 original java format으로 변경
 - 전에 추출한 변수들을 적절히 끼워넣음
- 예시)

```
for(i = 0 ~ N) {  
    print(i)  
    println("값:"+i)  
}
```



매칭 대상 라인: for(i = 0 ~ N) {
[예약어매칭]
⇒ 공백 제거: for(i=0~N){
⇒ 특수문자 기준 분리: [for, i, 0, N]
⇒ 검사할 단어: for, 예약어 화이트리스트 = [print, println, for]
⇒ 예약어(for) 정규식 매칭 시작
⇒ 라인에서 변수들만 추출: [i, 0, N, {}]
⇒ 변수들을 자바코드 포맷에 채워서 완성: for(int i=0;i<=N;i++){
⇒ 예약어 매칭 종료



```
for(int i=0;i<=N;i++){  
    System.out.print(i);  
    System.out.println("값:"+i);  
}
```

실제 동작 결과

```
a=0
b="Coway"
C=3.14
a=1
b="Project"
C=3.1415
d=new testClass()
```



```
a=0
b="Coway"
C=3.14
a=1
b="Project"
C=3.1415
d=new testClass()
```



COWAY

변환에 성공하였습니다

Pseudo Code

```
a=0
b = "Coway"
C = 3.14
a=1
b= "Project "
C = 3.14.15
d = new testClass()
```

Java Code

```
int a=0;
String b = "Coway";
double C = 3.14;
a=1;
b= "Project ";
C = 3.14.15;
testClass d = new testClass();
```



매칭 대상 라인: a=1
[변수추론]
⇒ 좌변(a) 우변(1)rhs = 1 type = 1
⇒ 정규식으로 추론된 우변의 타입 (int)
⇒ 이미 선언되었던 변수 (변수 선언 기록 테이블 = {a=int, b=String, C=double})
⇒ 매칭 성공
⇒ 매칭 종료 :: a=1;

매칭 대상 라인: b= "Project "
[변수추론]
⇒ 좌변(b) 우변("Project ")rhs = "Project " type = -1
rhs = "Project " type = -1
rhs = "Project " type = 8
⇒ 정규식으로 추론된 우변의 타입 (String)
⇒ 이미 선언되었던 변수 (변수 선언 기록 테이블 = {a=int, b=String, C=double})
⇒ 매칭 성공
⇒ 매칭 종료 :: b= "Project ";

매칭 대상 라인: C= 3.1415
[변수추론]
⇒ 좌변(C) 우변(3.1415)rhs = 3.1415 type = -1
rhs = 3.1415 type = -1
rhs = 3.1415 type = -1
rhs = 3.1415 type = 4
⇒ 정규식으로 추론된 우변의 타입 (double)
⇒ 이미 선언되었던 변수 (변수 선언 기록 테이블 = {a=int, b=String, C=double})
⇒ 매칭 성공
⇒ 매칭 종료 :: C= 3.1415;

3. 시연 동영상

Q&A

감사합니다