# ☀ 이-a : Gradle로 프로젝트 폴더 준비하기

```
~/git/bitcamp-study$ mkdir study-project ⏎
              $ cd study-project ⏎
study-project $ gradle init ⏎
              ① 프로젝트유형: application
              ② 언어 : Java
              ③ 프로젝트 개수 : only one ⟨ 빌드스크립트 언어 : Groovy
              ④ 단위테스트 : Junit
              ⑤ 프로젝트명 : (default) study-project
              ⑥ 자바기본패키지 : com.eomcs.pms
          $ gradle run ⏎
              -g
```
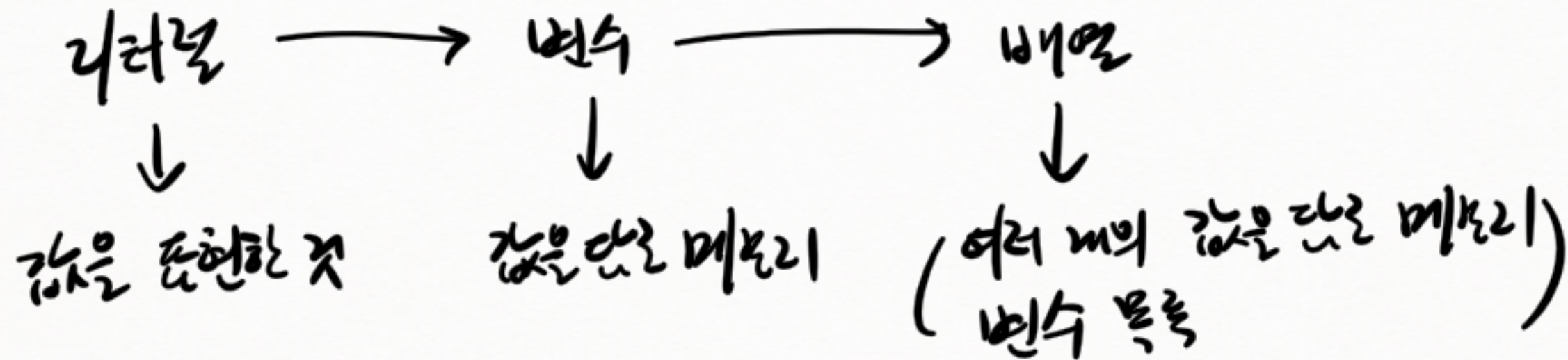
* 01-a : class diagram

App

＊ 02 : 리터럴(literal), 변수(variables), 배열(array)

리터럴 ⟶ 변수 ⟶ 배열
  ↓              ↓              ↓
값을 표현한 것   값을 담은 메모리   (여러 개의 값을 담은 메모리)
                                변수 묶음

# *02-a : 리터럴라 콘솔 출력

## 리터럴 ( literal )

↓

### 값을 표현한 것

- 정수
- 부동소수점
- 논리
- 문자
- 문자열

**정수**
- ✓ 10진수 → 100
- 8진수 → 0144
- 2진수 → 0b0110 0100
- 16진수 → 0x6    L

**부동소수점**
- 3.14
- 31.4e-1
- 314e-2
- 0.314e1
- 0.0314e2

**논리**
- true
- false

**문자**
- 'A'
- '가'
- '\u0041'
- '\t'
- '\b'
- '\n'

**문자열**
- "A"
- "AB가격"
- "AB\t가격"
- "AB\n가격"

제어문자
(escape character)

*02-a : 리터럴라 콘솔 출력
　　　　　콘솔 출력

what　　do
　↓　　↓
에교. 끄다()
지롱.
필요5.

객체(what)
↙
표현
값 이나 식(express.제)
System.out.print( ↙ ) ← 값 출력
　　‾‾‾　‾‾‾‾　‾‾‾‾‾
　　속득　콘솔

System.out.println( ↙ ) ← 값 출력 후 줄바꿈
　　　　‾‾‾　　‾‾‾‾‾‾‾

System.out.printf("출력형식", 값, 값, …)
　　　‾‾‾　‾‾‾‾‾
　　　　　　↑　　　↖ 출력형식에 맞춰 값 출력
메서드=함수(do)
⇊
기능수행

※ 02-b : 변수 와 키워드 입력

정수 → int __no__;

문자열 → String __name__;

날짜 → java.util.Date __registeredData__;

new java.util.Scanner(__System.io__);

↑

키워드

\* 02-C : 배열과 조건제어문

에) 데이터타입[ ] 변수 = new 데이터타입 [개수];

int[ ]  no  =  new  int [5];

if (조건) {

}
≡

while (조건) {

}
≡

for ( 변수초기화 ; 조건 ; 증감식 ) {

}
≡