KH정보교육원

JAVA 객체지향 I

KH 정보교육원

KH정보교육원

Chap01. 객체지향언어란?

KH정보교육원

객체지향언어

현실 세계는 사물이나 개념처럼 독립되고 구분되는 각각의 객체로 이루어져 있으며, 발생하는 모든 사건들은 객체간의 상호 작용이다.
이 개념을 컴퓨터로 옮겨 놓아 만들어낸 것이 객체지향 언어이다.

객체지향 프로그래밍

프로그래밍에서는 현실세계의 객체(사물, 개념)를 클래스(class)와 객체(Object)의 개념으로 컴퓨터에서 구현한다.

객체지향 3대 원칙

KH정보교육원

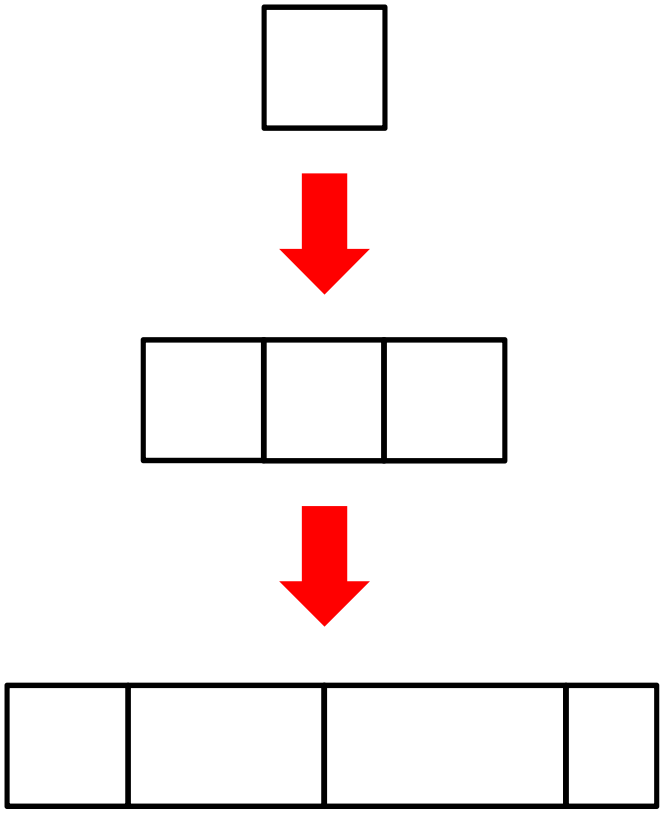
캡슐화(encapsulation)

상속(inheritance)

다형성(polymorphism)

KH정보교육원

클래스의 등장 배경



변수

1개의 자료형
1개의 데이터

배열

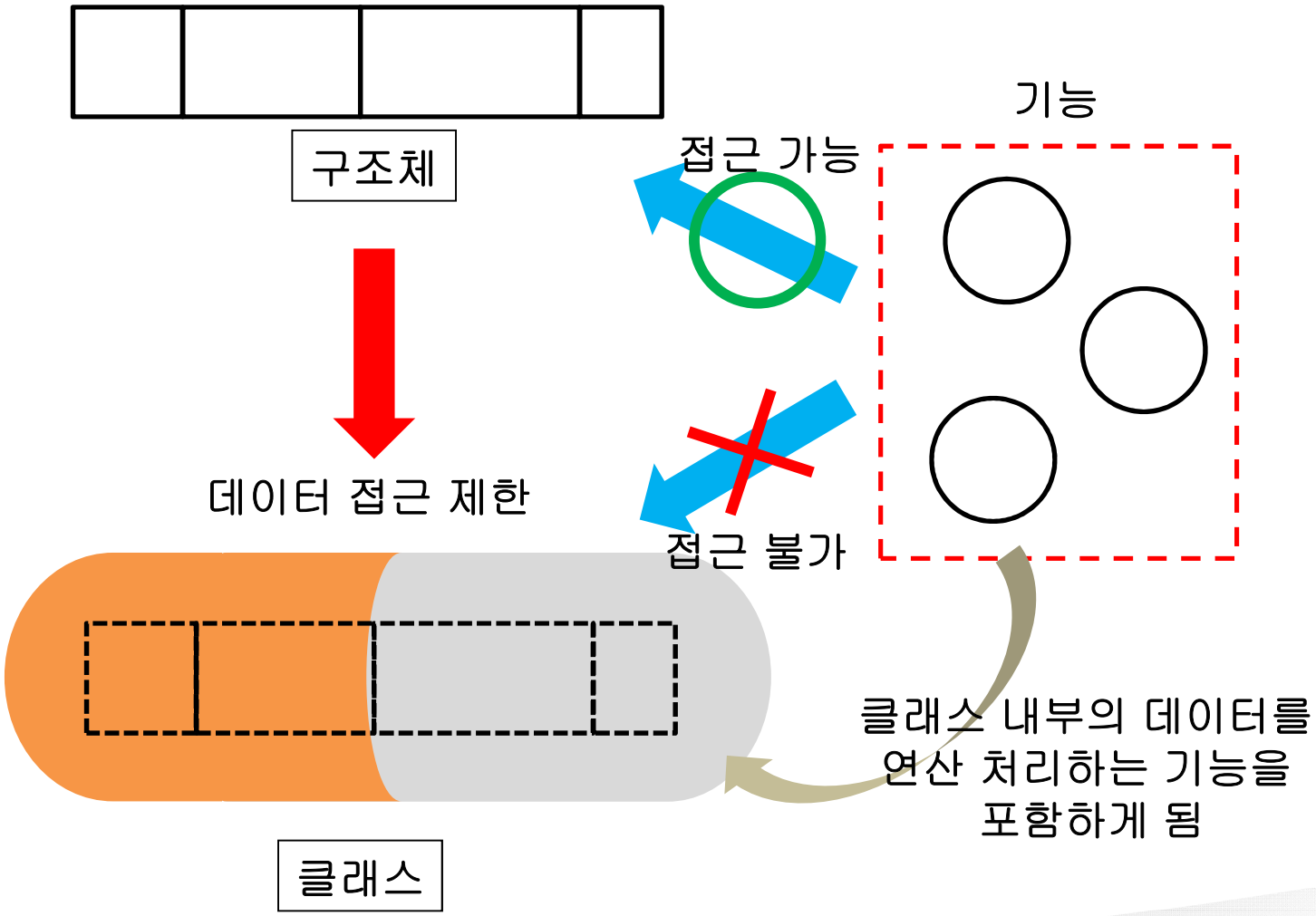
1개의 자료형
여러 개의 데이터

구조체

여러 개의 자료형
여러 개의 데이터

KH정보교육원

클래스의 등장 배경



캡슐화란

추상화를 통해 정리된 데이터들과 기능을 하나로 묶어 관리하는 기법을 말한다.

클래스의 가장 중요한 목적인 데이터의 접근제한을 원칙으로 하여 클래스 외부에서 데이터의 직접 접근을 막고, 대신 데이터를 처리하는 함수(매소드)들을 클래스 내부에 작성하여 데이터에 접근하는 방식을 캡슐화라고 한다.

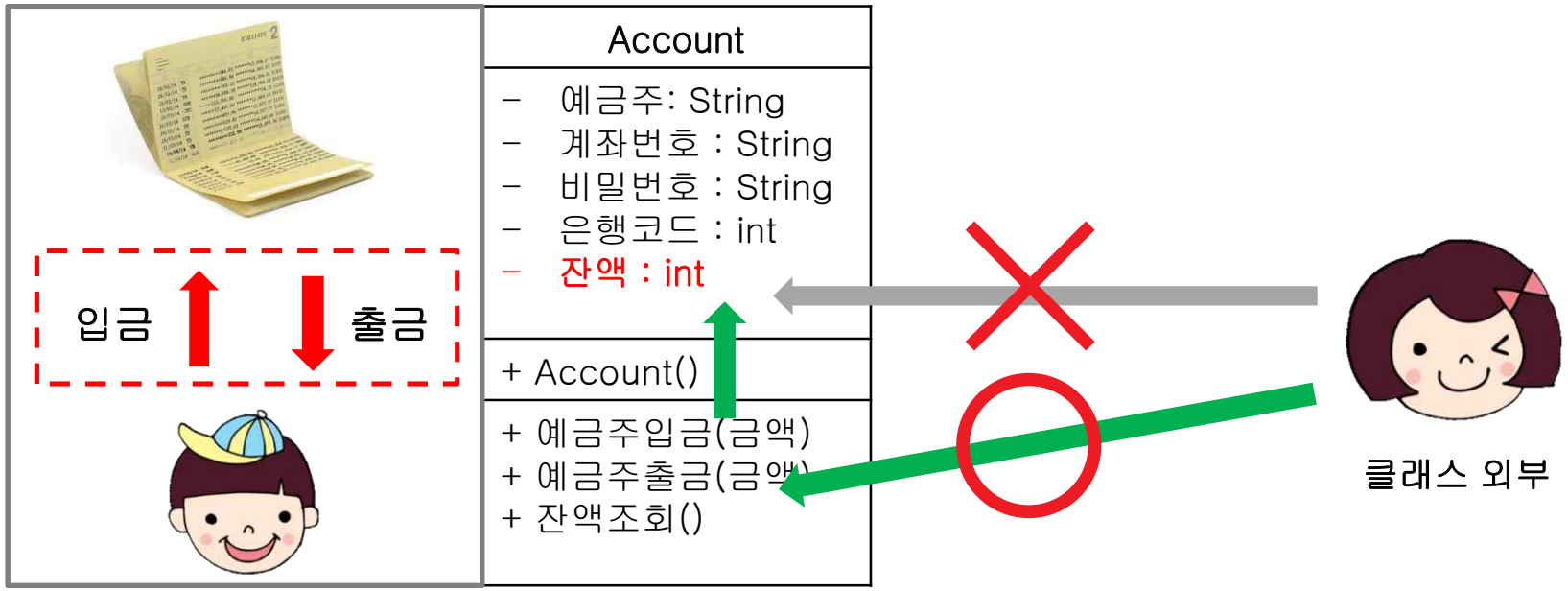
KH정보교육원

캡슐화 원칙

클래스의 멤버 변수에 대한 접근권한은 private을 원칙으로 한다.

클래스의 멤버 변수에 대한 연산처리를 목적으로 하는 함수들을 클래스 내부에 작성한다.

멤버 함수는 클래스 밖에서 접근할 수 있도록 public으로 설정한다.



Account 클래스로 생성된 김철수학생 명의의 계좌 객체

객체지향언어

KH정보교육원

클래스

객체의 특성에 대한 정의를 한 것으로 캡슐화를 통해 기능 포함한 개념,
사물이나 개념의 공통 요소(관련내용, 하는 기능)를 추상화(abstraction)
하여 정의함.
예) 제품의 설계도, 제품을 만드는 공장

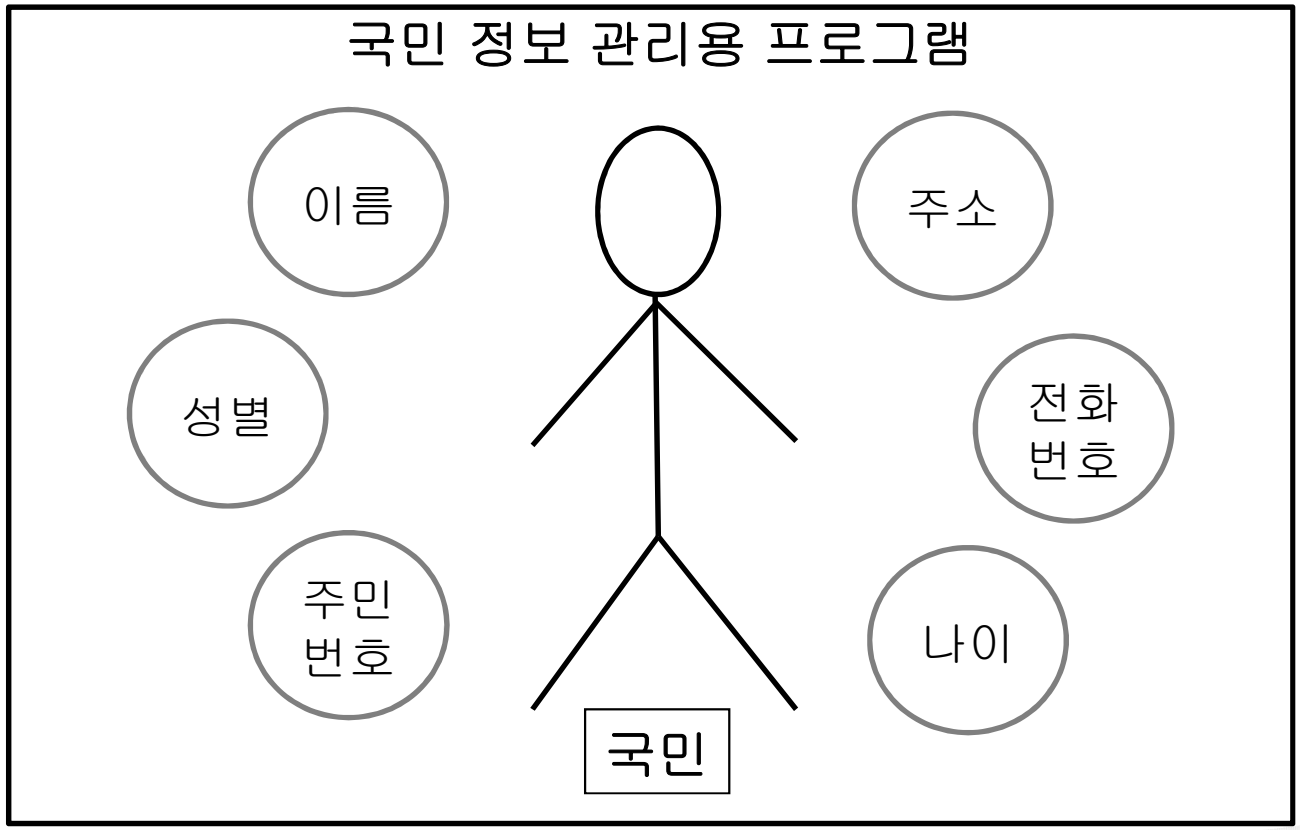
추상화(abstraction)

유연성을 확보하기 위해 구체적인 것은 제거한다는 의미.
프로그램에서 필요한 공통점을 추출하고, 불필요한 공통점을 제거하는 과정

KH정보교육원

추상화(abstraction) 예시

국가에서 국민 정보 관리용 프로그램을 만들려고 할 때,
프로그램에서 요구되는 “국민 한 사람”의 정보를 추상화 한다면?



추상화(abstraction) 예시

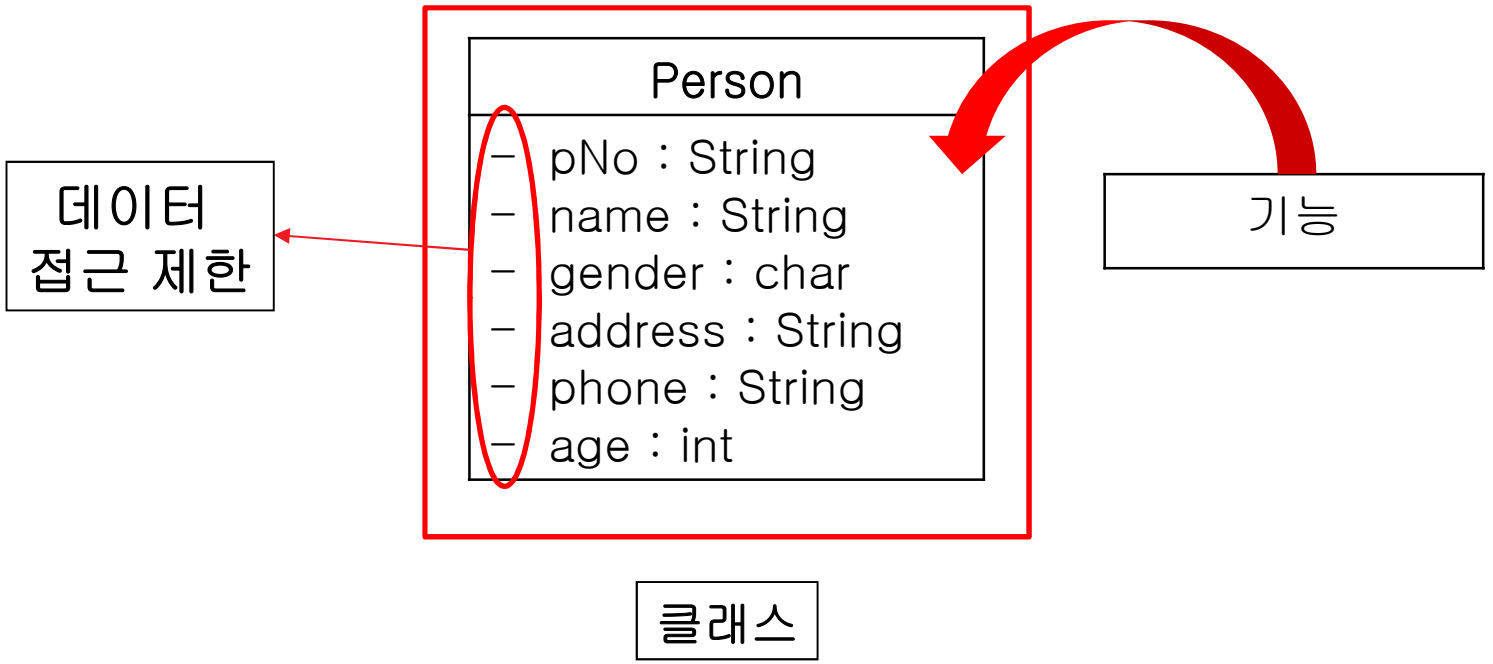
앞 페이지에서 추상화한 결과물을 객체 지향 프로그래밍 언어를 사용해서 변수명(데이터이름)과 자료형(데이터타입)을 정리함

항목	변수명	자료형(type)
주민등록번호	pNo	String
이름	name	String
성별	gender	char
주소	address	String
전화번호	phone	String
나이	age	int

KH정보교육원

추상화(abstraction) 예시

앞 페이지에서 정리된 변수명과 자료형을 클래스 다이어그램(UML)으로 표현한다면 아래와 같다.



실습문제

KH정보교육원

다음 상황에서 객체지향적으로 프로그래밍을 하고자한다. 각각 클래스를 설계해보자.

1)야구 스포츠게임에서 플레이어객체를 만들고자 한다. 실제 야구게임에서 모티브를 얻어 추상화 해보자.

2)결혼정보프로그램을 만들고자 한다.
회원객체를 설계해보자.

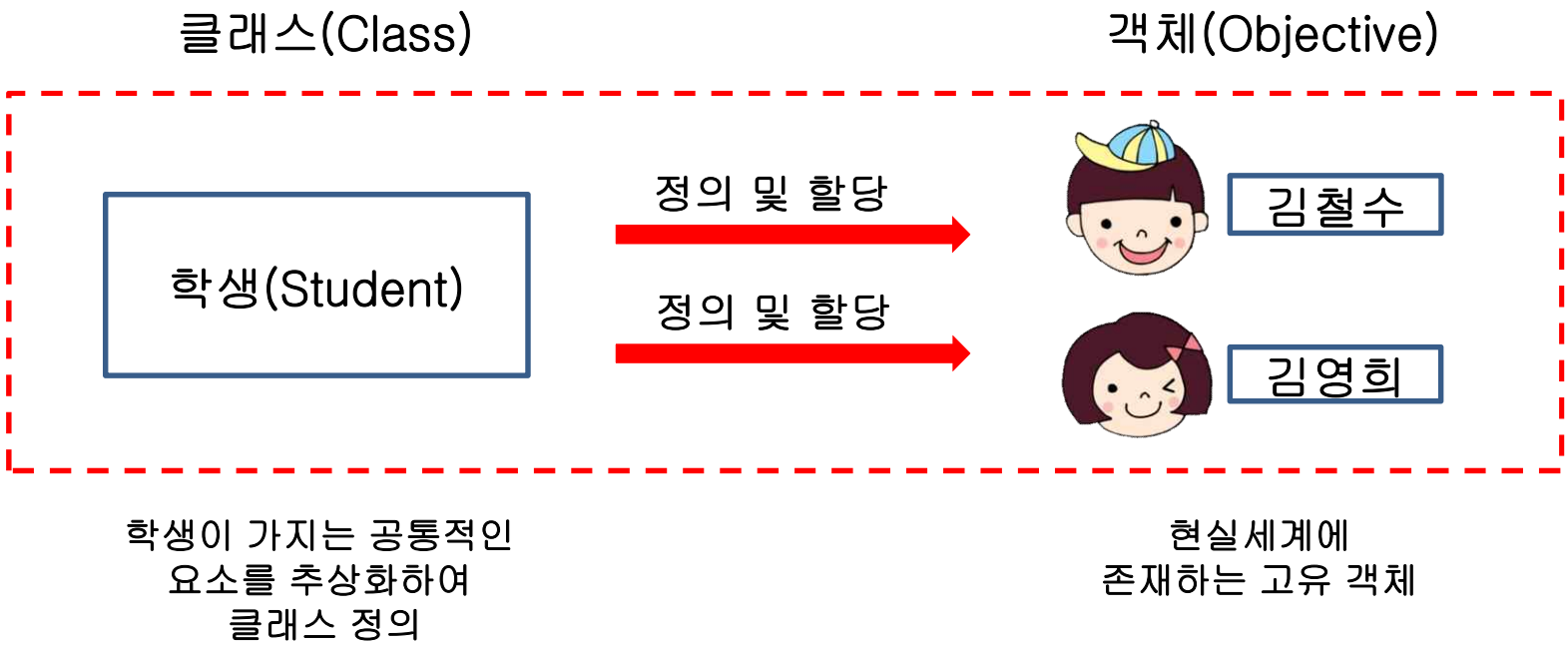
3)가구점에서 판매할 상품들에 대해 판매정보를 객체화하려고 한다. 어떤 클래스를 어떻게 설계하겠는가.

객체지향언어

KH정보교육원

객체란?

현실에 존재하는 독립적이면서 하나로 취급되는 사물이나 개념을 말한다.
객체 지향 언어에서 객체의 개념은 new연산자를 통해 클래스가 지정한 데이터를 메모리에 할당한 결과물(Object)이다.



KH정보교육원

Chap02. 클래스(class)

클래스 선언

KH정보교육원

클래스란?

자바에서 하나의 설계된(추상화된) 객체를 의미한다. 클래스명은 하나의 자료형이 된다.

[접근제한자] [예약어] class 클래스명{

```
자료형 변수명;
자료형 변수명;
```

속성값 설정

[접근제한자] 생성자명(){}

기능정의 설정

```
[접근제한자]리턴형 메소드명(매개변수){
        기능정의
}
```

}

클래스 접근제한자

KH정보교육원

접근제한자

구분		같은 패키지 내	전체
+	public	O	O
~	(default)	O	

예) public class 클래스명 {

.....

}

(default) class 클래스명{

.....

}

클래스 예약어

KH정보교육원

클래스 선언시 사용 가능한 예약어

- final : 변경될 수 없는 클래스, 상속 불가

예) public final class 클래스명 {

.....

}

- abstract : 클래스내 추상 메서드가 선언되어있는 추상클래스, 반드시 상속받아 생성함

예) public abstract class 클래스명 {

abstract 리턴값 메소드명(매개변수);

abstract 리턴값 메소드명(매개변수);

}

두 개 예약어를 동시에 사용할 수 없음

클래스 선언

KH 정보교육원

예제

```
public class Person{
        private String name;
        private int age;

        public Person(){}

        public String getName(){
            return name;
        }
        public void setName(String name){
            this.name = name;
        }
}
```

클래스 생성

KH정보교육원

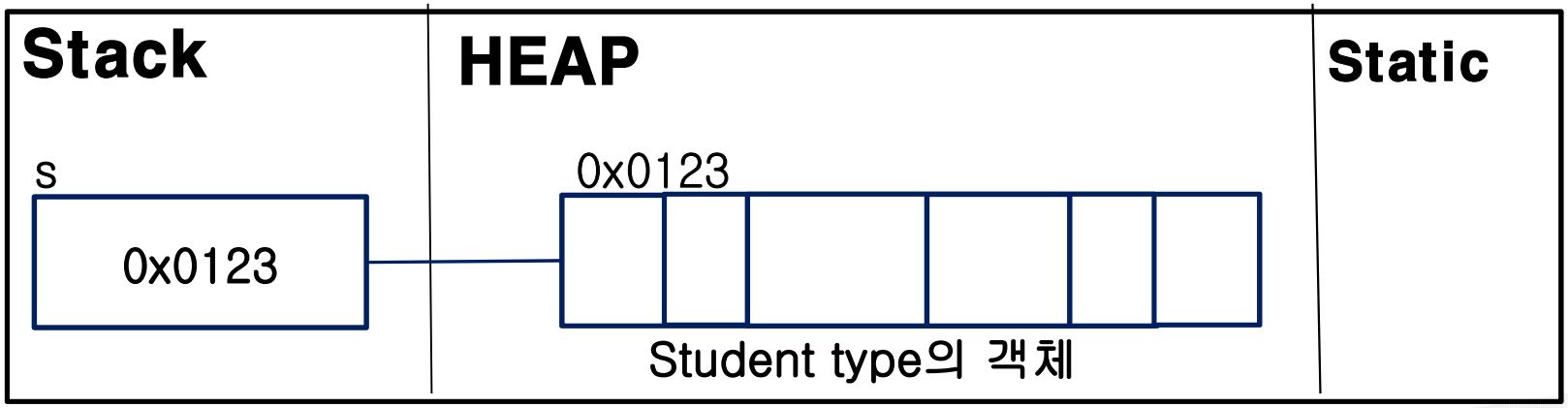
클래스의 생성(할당)

new 연산자(생성자)를 사용하여 객체를 생성하면 heap 메모리 공간에 서로 다른 자료형의 데이터가 연속으로 나열 할당된 객체 공간이 만들어진다. 이것을 인스턴스(instance)라고 한다.

표현식

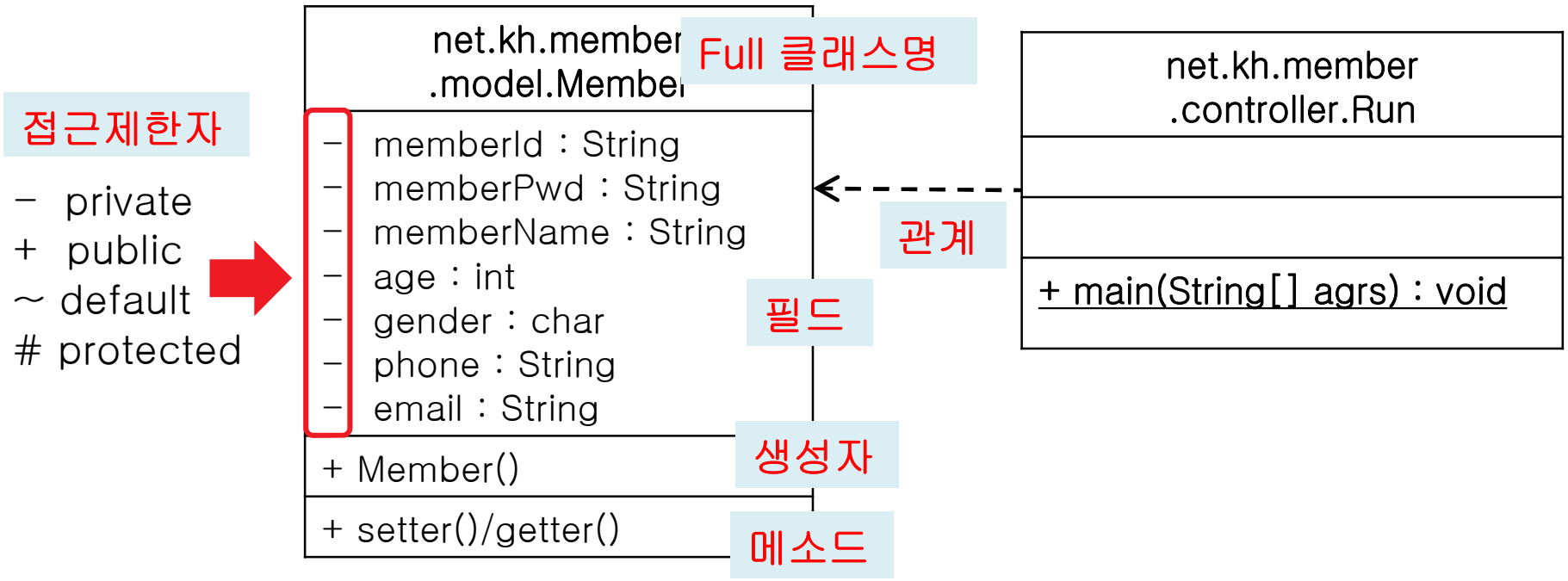
클래스명 변수명(참조형) = new 클래스명();

예) Student s = new Student();



참고 - UML 다이어그램

KH정보교육원



참고 - UML 다이어그램

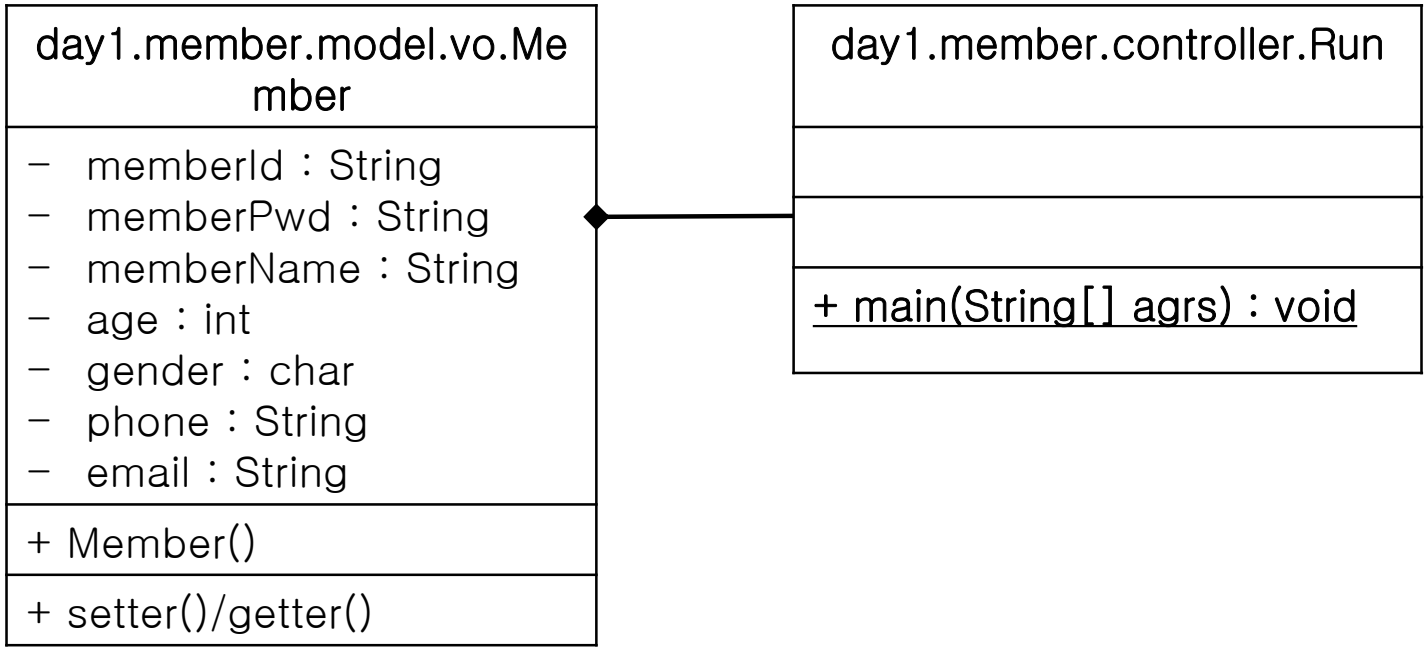
KH정보교육원

관계	UML 표기	설명
Genelization 일반화		상속관계 자식클래스→부모클래스
Realization 실체화		구현관계 구현클래스→ 인터페이스
Dependency 의존		객체생성, 객체사용 사용클래스→피사용클래스
Association 연관		필드로 다른 객체를 참조. 사용클래스→피사용클래스
Aggregation 약집합		객체의 라이프 사이클이 다름 사용클래스→피사용클래스
Composition 강집합 (완전포함)		객체의 라이프사이클이 동일 사용클래스→피사용클래스

실습문제

KH정보교육원

1) 아래 클래스 다이어그램을 보고 클래스를 작성하세요.



package	class	method	설명
day1.member.controller	Run	*Main(args:String[]) : void*	실행용 메소드 기본 생성자로 객체를 만들고 Setter를 이용해 값 변경 후 Getter를 이용해 출력

KH정보교육원

Chap03. 필드(field)

클래스 필드(데이터)

KH 정보교육원

필드란?

추상화된 객체에 실체화(인스턴스화)할때 실체화된 객체의 데이터를 보관하는 저장장소를 의미하고, 변수이다. 멤버변수를 의미함.

*** 클래스의 변수 종류**

범위에 따라 구분되며 클래스변수, 멤버변수, 지역변수 나뉨 (선언위치로 구분)

표현식

```
접근제한자 class 클래스명{
    접근제한자 [예약어] 자료형 변수명 [= 초기값];
}
```

클래스 필드 접근제한자

KH정보교육원

접근제한자

구분		해당 클래스 내부	같은 패키지 내	후손 클래스 내	전체
+	public	O	O	O	O
#	protected	O	O	O	
~	(default)	O	O		
-	private	O			

참고 - package

KH정보교육원

패키지란?

서로 관련된 클래스 혹은 인터페이스의 묶음이다.

클래스가 클래스파일(*.class)인 것 처럼, 패키지는 폴더와 비슷하다.

패키지는 서브 패키지를 가질 수 있으며 '.'으로 구분한다.

예시) Scanner클래스의 full name은 패키지명이 포함된 java.util.Scanner이다.

참고 - package

KH정보교육원

패키지의 선언

패키지는 소스 파일 첫 번째 문장에 단 한번 선언한다.
하나의 소스 파일에 둘 이상의 클래스가 포함된 경우, 모두 같은 패키지에 속한다.
모든 클래스는 하나의 패키지에 속하며, 패키지가 선언되지 않은 클래스는 지동적으로 이름없는 패키지(default)에 속하게 된다.

예시) package java.util;

참고 - import

KH정보교육원

import란?

사용할 클래스가 속한 패키지를 지정하는데 사용

Import문을 사용하면 클래스를 사용할 때 패키지명을 생략할 수 있다.

java.lang 패키지의 클래스는 import 하지 않고도 사용할 수 있다.

java.lang 패키지의 클래스 -> String, Object, System…

참고 - import

KH정보교육원

import문의 선언

Import문은 패키지문과 클래스 선언의 사이에 선언한다.

Import문은 컴파일시에 처리되므로 프로그램 성능에 아무런 영향을 주지 않는다.

지정된 패키지에 포함된 클래스는 import 가능하지만,

서브패키지에 속하는 모든 클래스까지 import는 불가능하다.

예시)
```
import java.util.Calendar;
import java.util.Date;
import java.util.*;

import java.*;    //불가능
```

참고 - import

KH정보교육원

import문 주의사항

이름이 같은 클래스가 속한 두 패키지를 import 할 때는 클래스 앞에 패키지명을 붙여 구분해 주어야 한다.

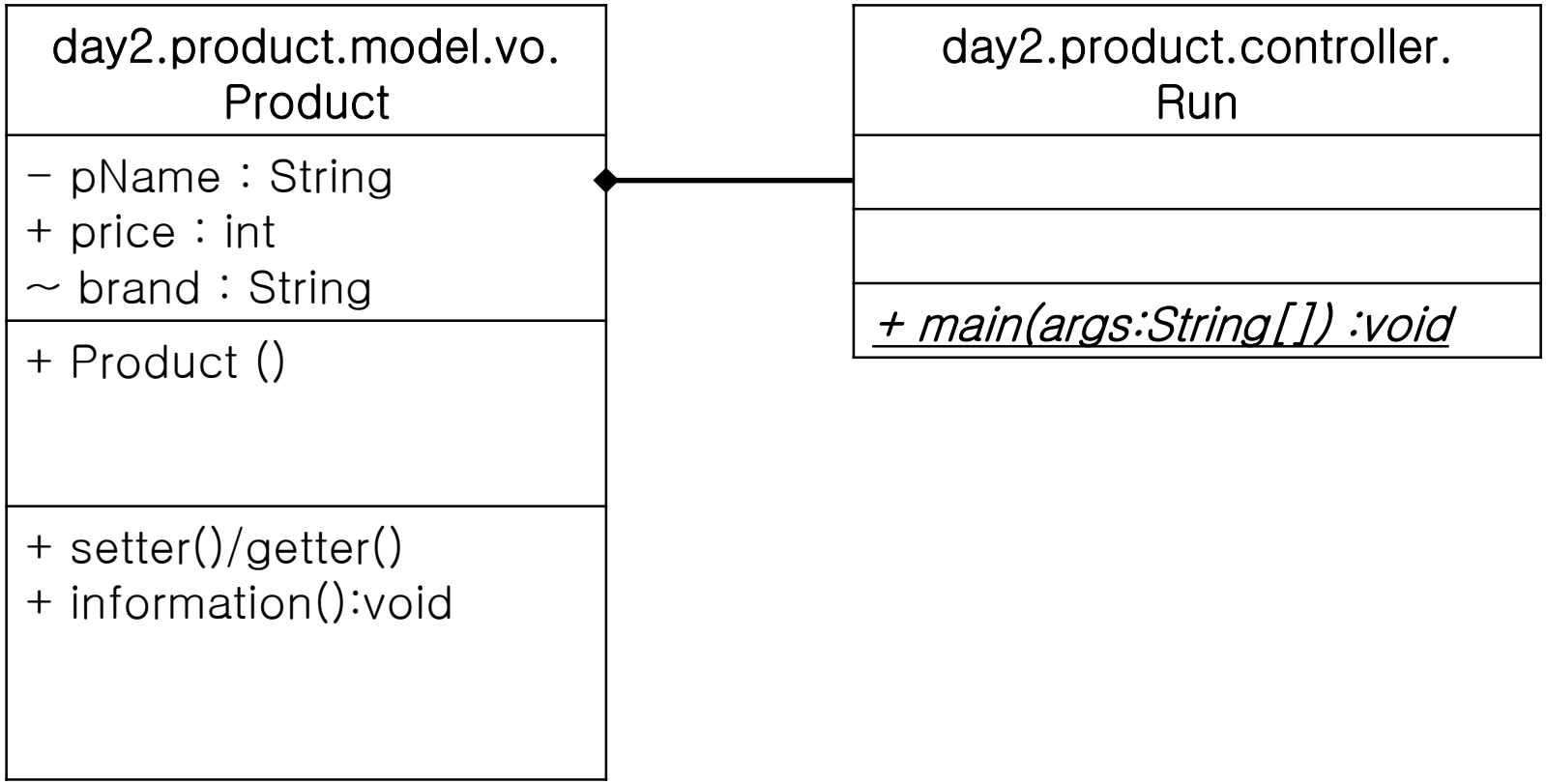
예시)
```
import java.sql.Date;
import java.util.Date;

public class ImportTest{
    public static void main(String[] agrs){
        java.util.Date today = new java.util.Date();
    }
}
```

실습문제

KH정보교육원

아래 클래스 다이어그램을 보고 클래스를 작성하세요.

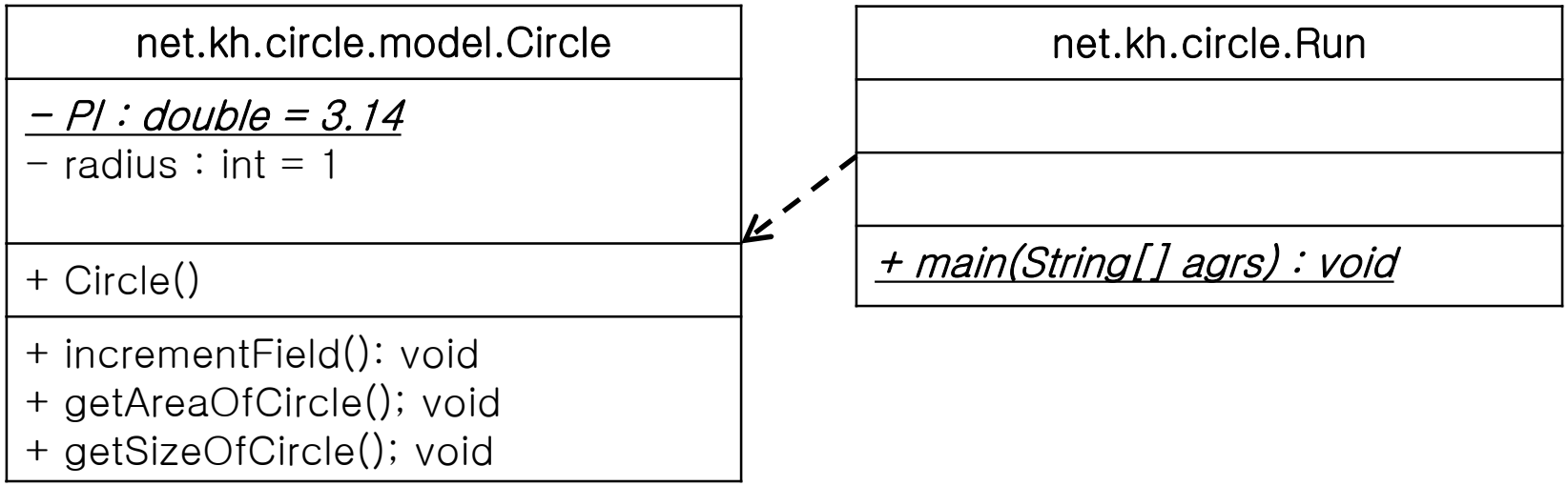


package	class	method	설명
day2.product.controller	Run	*Main(args:String[]) : void*	실행용 메소드 Product 객체 생성 information()으로 출력

실습문제

KH정보교육원

아래 클래스 다이어그램을 보고 클래스를 작성하세요.



package	class	method	설명
net.kh.circle	Run	*Main(args:String[]) : void*	필드에 값 셋팅후 넓이, 원둘레 구하기. incrementField를 통해 반지름 1증가후, 다시 넓이, 원둘레 구하기

클래스 필드예약어

KH정보교육원

static : 같은 타입의 여러 객체가 공유할 목적의 필드에 사용하며, 프로그램 start시에 정적 메모리영역에 자동 할당되는 멤버에 적용한다.

표현식

접근제한자 class 클래스명 {

접근제한자 static 자료형 변수명;

}

예) public class Academy {

private static int temp;

}

클래스 필드예약어

KH정보교육원

final : 하나의 값만 계속 저장해야 하는 변수에 사용

표현식

class 클래스명 {

접근제한자 final 자료형 변수명 = 초기값;

}

예) class Academy {

private final double tax = 3.3;

private int money;

}

클래스 필드 접근제한자

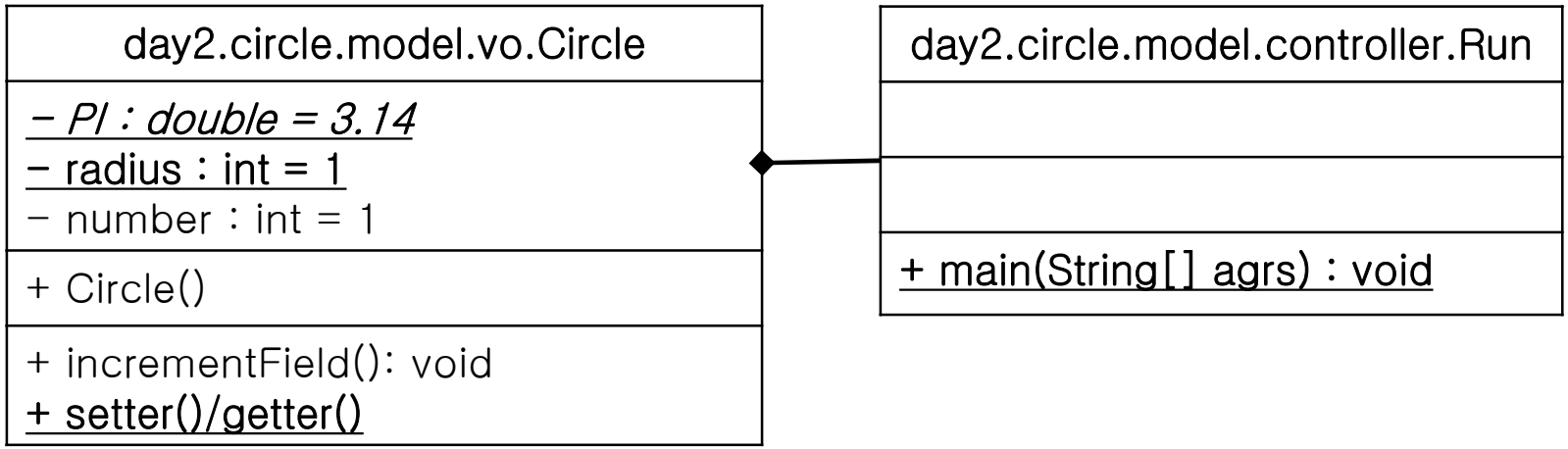
KH 정보교육원

```
예) class Academy {
        public int temp1;
        protected int temp2;
        int temp3;   //접근제한자 생략하면 default임
        private int temp4;  //캡슐화의 원칙임
    }
```

실습문제

KH정보교육원

아래 클래스 다이어그램을 보고 클래스를 작성하세요.



package	class	method	설명
day2.circle.controller	Run	*Main(args:String[]) : void*	실행용 메소드 Getter로 출력해보고 Setter로 값 변경 후 Getter로 다시 값 출력

클래스 초기화블럭

KH 정보교육원

{ } : 클래스의 멤버 변수를 초기화 시키는 블록

static 블록 : static 필드 초기화, 프로그램 시작시 초기화

인스턴스 블록 : 인스턴스변수 초기화, 객체 생성시 마다 초기화

표현식

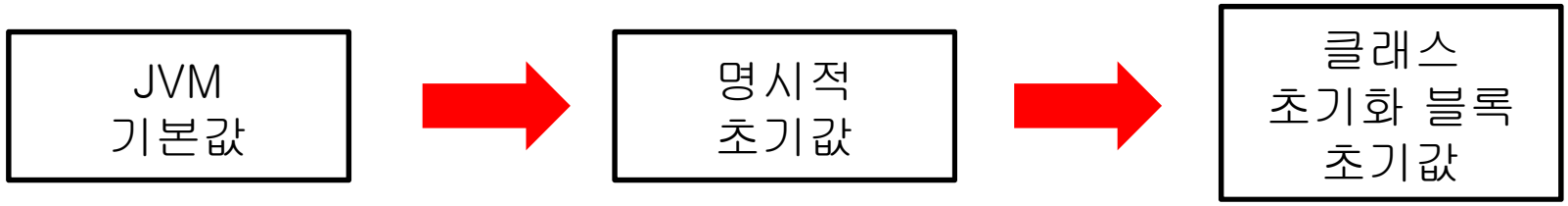
```
접근제한자 class 클래스명 {
        접근제한자 static 자료형 필드명;
        접근제한자 자료형 필드명;
      static{ 필드 = 초기값  };
      { 필드 = 초기값 };
 }
```

클래스 초기화블럭

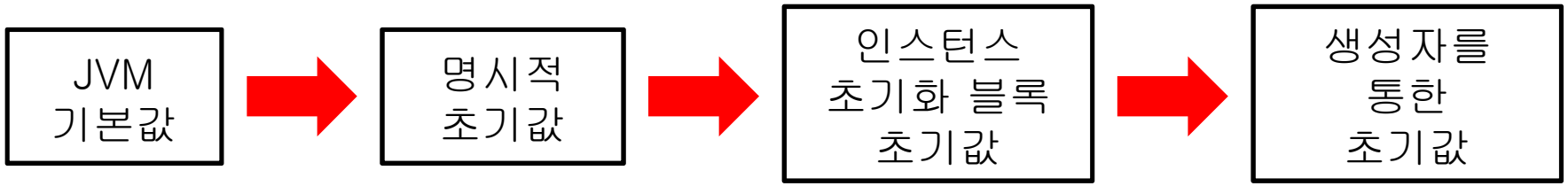
KH정보교육원

초기화 순서

- 클래스 변수



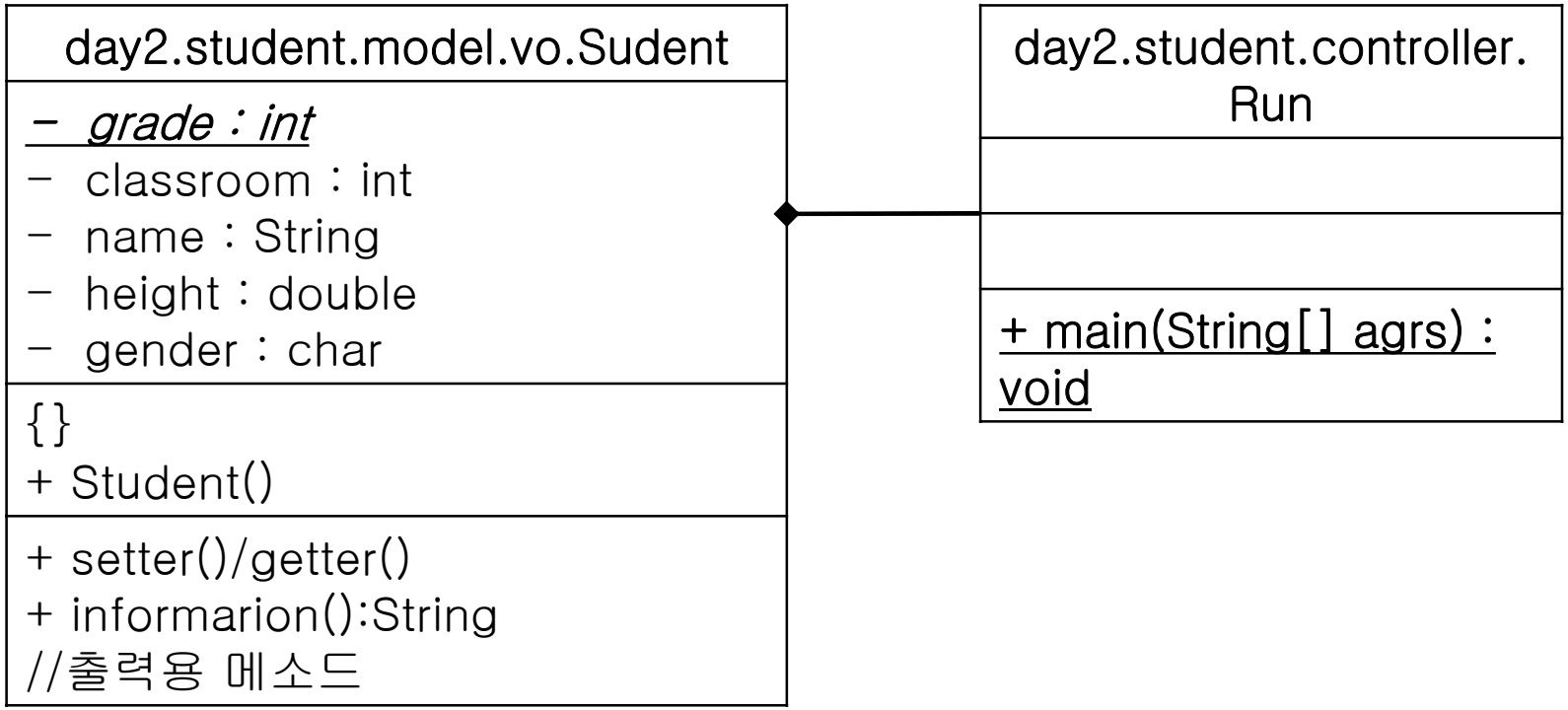
- 인스턴스 변수



실습문제

KH정보교육원

아래 클래스 다이어그램을 보고 클래스를 작성하세요.

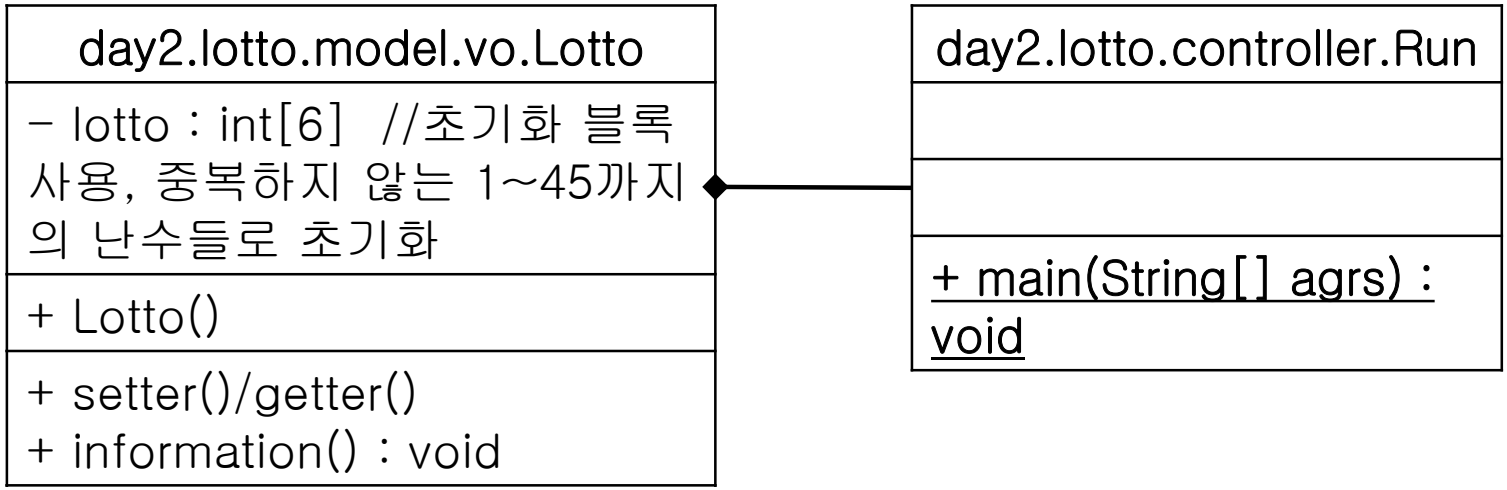


package	class	method	설명
day2.student.controller	Run	*Main(args:String[]) : void*	초기화 블록으로 멤버변수 임의값으로 초기화, static변수는 static 초기화 블럭 사용 실행용 메소드 User객체 생성 후 Information()으로 출력

실습문제

KH정보교육원

아래 클래스 다이어그램을 보고 클래스를 작성하세요.



package	class	method	설명
day2.lotto.model.vo	Lotto	information():String	반복문을 이용하여 배열을 출력하는 메소드
day2.lotto.controller	Run	*Main(args:String[]) : void*	실행용 메소드 User객체 생성 후 Information()으로 출력

KH정보교육원

Chap04. 생성자(constructor)

클래스 생성자

KH정보교육원

생성자란

객체가 heap에 할당될 때 객체 안에서 만들어지는 필드의 초기화를 담당함.

new 할때 실행되는 함수로 초기값을 생성자 쪽으로 전달함.

생성자함수가 전달된 초기값을 받아서 필드에 기록한다.

생성자 규칙

생성자는 선언은 메소드 선언과 유사하나 리턴값이 없다.

클래스 이름이랑 똑같이 지정해야 한다.

클래스 생성자

KH정보교육원

표현식

접근제한자 class 클래스명 {

접근제한자 클래스명(){ };

}

예) public class Academy{

private int studentNo;

private String name;

public Academy(){ } //기본(default) 생성자

}

클래스에서 기본 디폴트 생성자만 있는 경우 멤버변수는 모두 기본 초기값으로 초기화 되어 생성됨.

클래스 생성자

KH정보교육원

생성자 종류

기본 생성자 (매개변수가 없는 생성자)

- 작성하지 않은 경우, 클래스 사용시 JVM이 기본 생성자 자동 생성

매개변수 있는 생성자

- 객체 생성시 전달 받은 값으로 객체를 초기화 하기 위해 사용함.
- 매개변수 있는 생성자 작성시 JVM이 기본 생성자 자동 생성 하지 않음
- 상속에 사용시 반드시 기본 생성자를 작성해야 함
- 오버로딩을 이용하여 작성함

클래스 매개변수 있는 생성자

KH정보교육원

표현식

```
class 클래스명 {
    [접근제한자] 클래스명(){}  //기본 생성자
    [접근제한자] 클래스명(매개변수){
        (this.)필드명 = 매개변수
    };
}
```

클래스 매개변수 있는 생성자

KH정보교육원

클래스에 생성자를 만들면 디폴트 생성자가 자동으로 생성되지 않아 디폴트 생성자를 활용 하려면 코드를 추가 해줘야 함.

```
예) class Academy{
        private int age;
        private String name;

        public Academy(){}
        public Academy(int age, String name){
            this.age = age;
            this.name = name;
        }
    }
```

this

KH정보교육원

this란

모든 인스턴스의 메소드에 숨겨진 채 존재하는 레퍼런스로, 할당된 객체를 가르킨다. 함수 실행시 전달되는 객체의 주소를 자동으로 받는다.

```
예) class Academy{
        String name;

      public Academy(){}
       public Academy(String name){
          this.name = name;
      }
   }
```

** 매개변수를 가지는 생성자에서 매개변수명이 필드명과 같을 경우 매개변수의 변수명이 우선하기 때문에 this객체를 활용해서 대입되는 변수가 필드라는 것을 구분해준다.

this()

KH 정보교육원

this()란

생성자, 같은 클래스의 다른 생성자를 호출할 때 사용

** 반드시 첫번째 줄에 선언해야 한다.

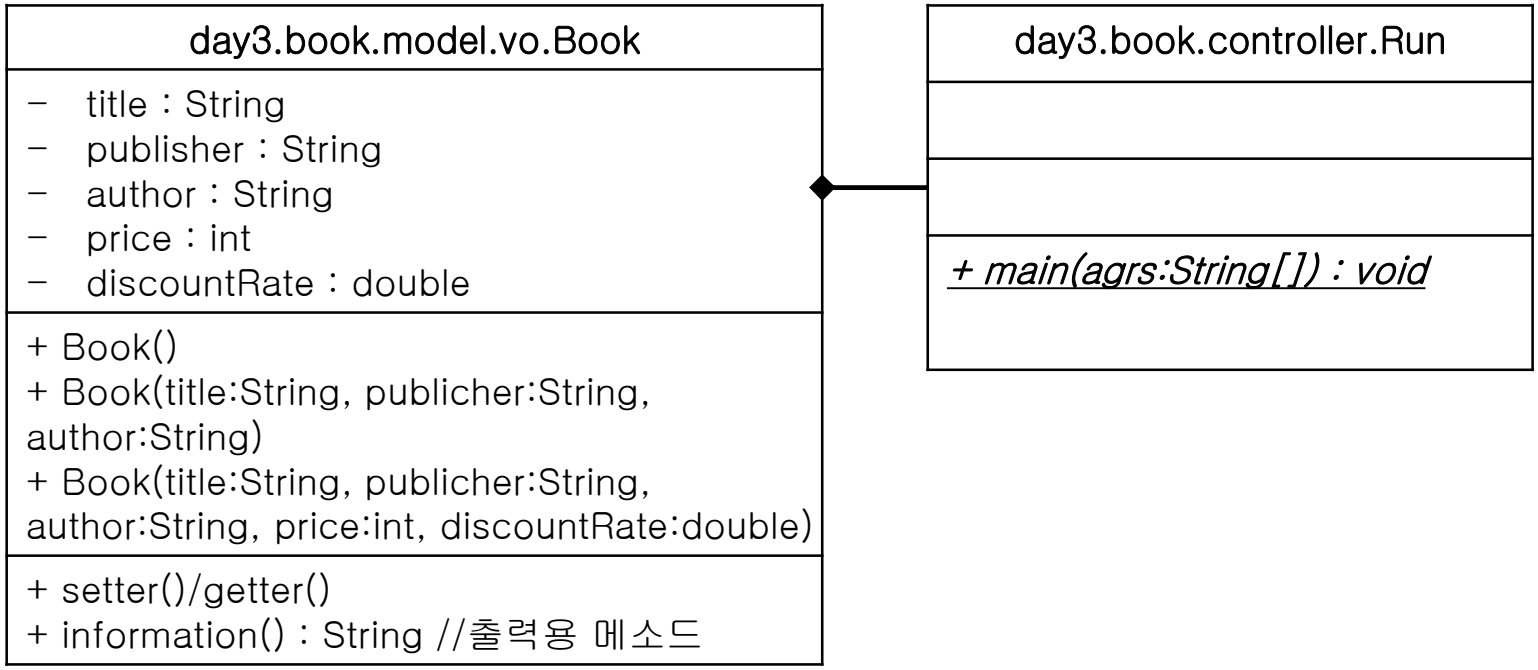
```
예) class Academy{
        int age;
    String name;

        public Academy(){
            this(20, "김철수");
        }
      public Academy(int age, String name){
          this.age = age;
          this.name = name;
    }
}
```

실습문제

KH정보교육원

1) 아래 클래스 다이어그램을 보고 클래스를 작성하세요.



package	class	method	설명
day3.book.model.vo	Run	*Main(args:String[]) : void*	실행용 메소드 매개변수 3개인 생성자와 매개변수 5개인 생성자로 각각 객체를 생성하여 Information()이용하여 출력

KH정보교육원

Chap05. 메소드(method)

메소드란

수학의 함수와 비슷하며 호출을 통해 사용한다. 전달 값이 없는 상태로 호출을 하거나, 어떤 값을 전달하여 호출을 하면, 함수 내에 작성된 연산을 수행하거나 수행 후 결과값을 반환하는 것을 말한다.

메소드

KH정보교육원

표현식

[접근제한자][예약어] 반환형 메소드명() {

실행내용 작성;

}

예) pubilc void showInformation(){

System.out.println(userName);

}

메소드의 접근제한자

KH정보교육원

구분	클래스	패키지	자손 클래스	전체
public	O	O	O	O
protected	O	O	O	
default	O	O		
private	O			

메소드의 예약어

KH정보교육원

구 분	설 명
static	static 영역에 할당하여 객체 생성 없이 사용함
final	종단의 의미, 상속시 오버라이딩 불가능
abstract	미완성된, 상속하여 오버라이딩으로 완성시켜 사용해야 함
syncronized	동기화처리, 공유 자원에 한 개의 스레드만 접근 가능함
static final (final static)	static과 final의 의미를 둘 다 가짐

메소드의 반환형

KH정보교육원

구 분	설 명
void	반환형이 없음을 의미, 반환값이 없을 경우 반드시 작성
기본자료형	연산 수행 후 반환값이 기본 자료형일 경우 사용함
배열	연산 수행 후 반환값이 배열인 경우 사용함
클래스	연산 수행 결과가 해당 클래스 타입의 객체일 경우 사용함 (클래스 == 타입)

* 반환형은 단 한 개만 가질 수 있다.

메소드의 매개변수

KH정보교육원

구 분	설 명
()	매개변수가 없는것을 의미함
기본자료형	기본형 매개변수 의미 값을 복사하여 전달하기 때문에 매개변수 값을 변경하여도 원래 값은 변경되지 않는다.
배열	배열, 클래스를 매개변수로 전달시에는 원데이터가 있는 곳의 주소값을 전달하는 것이기 때문에 전달 내용을 수정하면 원데이터가 수정된다.
클래스	
가변인자	매개변수의 개수를 유동적으로 설정하는 방법, 가변매개변수외 다른 매개변수가 있으면 가변매개변수를 마지막에 설정해야 함. 방법 : (자료형 ... 변수명)

* 매개변수의 수에 제한이 없다.

메소드의 종류

KH정보교육원

매개변수가 없고 리턴값이 있거나 없거나

표현식

- 리턴값 있는것

```
[접근제한자] 리턴형 메소드명() {
            ..........................
            return 반환값;
}
```

- 리턴값 없는것

```
[접근제한자] void 메소드명() {
            ..........................
}
```

메소드의 종류

KH정보교육원

매개변수가 있고 리턴값이 있거나 없거나

표현식

- 리턴값 있는것

```
접근제한자 리턴형 메소드명(자료형 변수명){
                ..........................
                return 반환값;
}
```

- 리턴값 없는것

```
접근제한자 void 메소드명(자료형 변수명){
                ..........................
}
```

메소드 오버로딩

KH정보교육원

오버로딩란

한 클래스 내에 동일한 이름의 메소드를 여러 개 작성하는 기법

조건

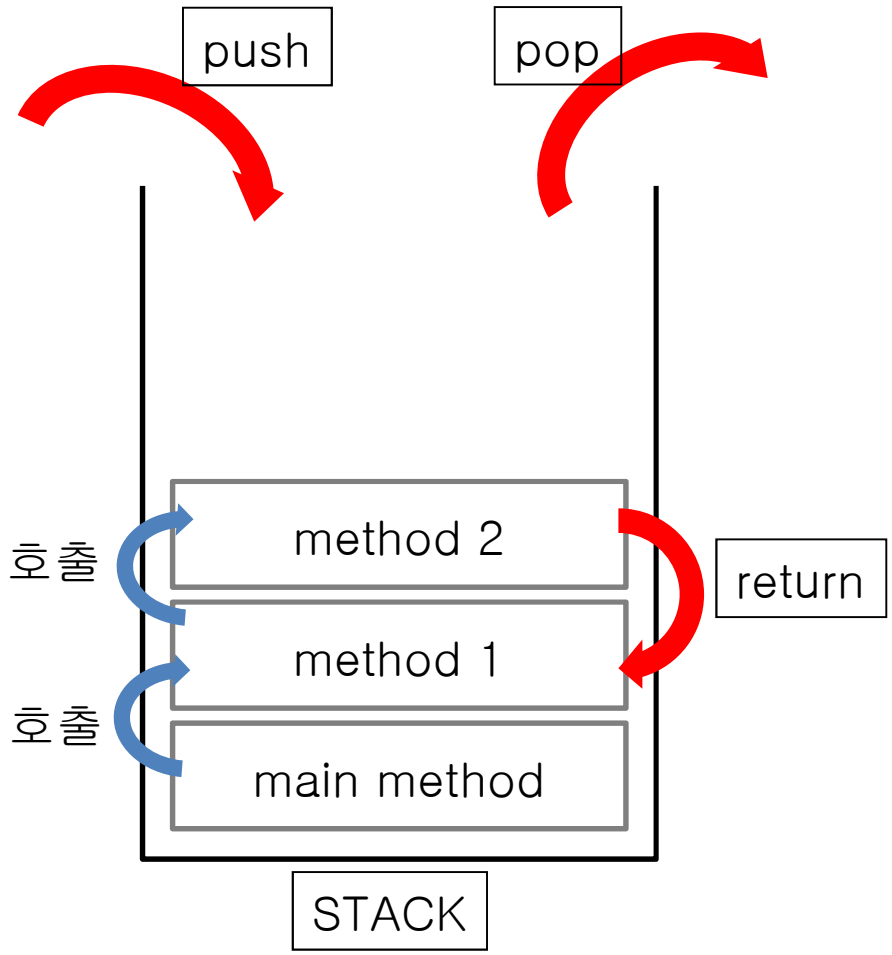
- 메소드 이름이 같아야 한다.
- 매개변수의 개수 또는 타입이 달라야 한다.

예) System.out.print() 매소드

java API document java.io.PrintStream.print 매소드 확인

return

KH정보교육원



return이란?

-해당 메소드를 종료하고,
자신을 호출한 메소드로
돌아가는 예약어

- 반환값(리턴값)을 가지고
자신을 호출한 메소드로
돌아갈 수 있다.

STACK의 자료구조

-LIFO(Last-Input-First-Out)

메소드 호출

KH정보교육원

호출할 클래스명, 매소드명을 적고 설정된 매개변수로 들어갈 값을 자료형과 일치한 값을 넣어야 한다.

표현식

매개변수 있는 경우

: 참조형 변수명.매소드명 (매개변수);

매개변수 없는 경우 : 참조형 변수명.매소드명();

```
예) pubilc void showInformation(){
        Person ps=new Person();
        ps.getAge() ; //매개변수 없는 경우
        ps.setAge(19);  int age=19; ps.setAge(age); //매개변수 있는 경우
       }
```

getter와 setter매소드

KH정보교육원

Setter 메소드

-필드에 변경할 값을 전달 받아서 필드값을 변경하는 메소드

-매개변수가 필드와 동일한 이름일 경우 this연산자를 붙여서 구분

표현식

```
접근제한자 void set필드명(자료형 변수) {
        (this.)필드명 = 자료형 변수;
}
```

Getter 메소드

- 필드에 기록된 값을 읽어서 요구하는 쪽으로 읽은 값을 넘기는 메소드

표현식

```
접근제한자 반환형 get필드명() {
        return 필드명;
}
```

getter와 setter매소드

KH정보교육원

Setter 메소드

```
예시) public void setBalance(int balance){
          this.balance = balance;
      }
```

Getter 메소드

```
예시) public int getBalace(){
          return balance;
      }
```

재귀호출(recursive call)

KH 정보교육원

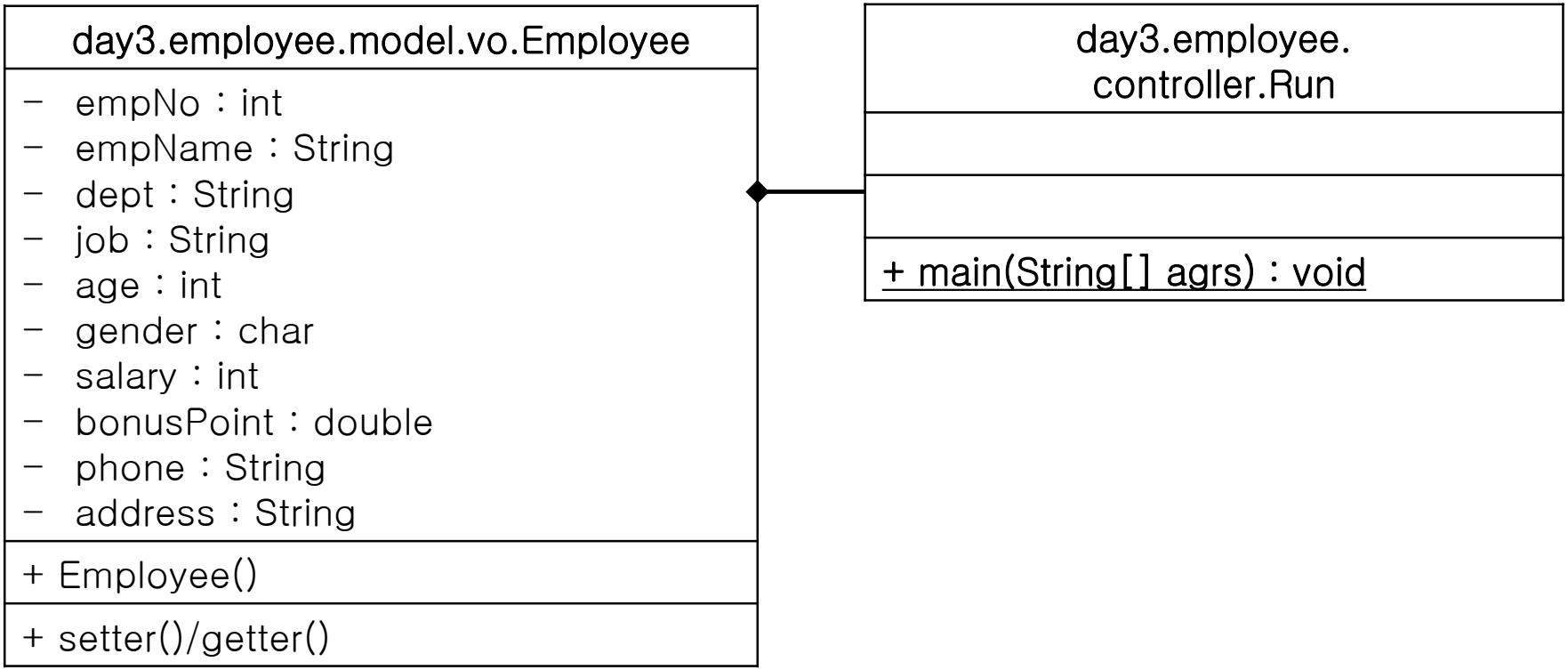
```
class Fatorial{
  public static void main(String[] args){
         int result=factor(5);
         System.out.println(result);
}
  public static int factor(int a){
        int temp;
        if(a==1){
           temp =  1;
         }
        else
            temp=a*factor(a-1);
        return temp;
}
```

실습문제

KH정보교육원

3) 아래 클래스 다이어그램을 보고 클래스를 작성하세요.

empNo	empName	dept	job	age	gender	salary	bonusPoint	phone	address
100	홍길동	영업부	과장	25	남	2500000	0.05	010-1234-5678	서울시 강남구



package	class	method	설명
day3.employee. controller	Run	*Main(args:String[ ]) : void*	실행용 메소드 기본 생성자로 객체를 만들고 Setter를 이용해 값 변경 후 Getter를 이용해 출력