

2025년 상반기 K-디지털 트레이닝

## 클래스

[KB] IT's Your Life



#### ♥ 다음과 같이 Car 클래스를 정의한 경우, 각 멤버 변수가 가지는 기본값은 무엇인가?

```
package ch06.sec06.exam01;

public class Car {
    //필드 선언
    String model;
    boolean start;
    int speed;
}
```

## CarExample.java

```
public class CarExample {
    public static void main(String[] args) {
        //Car 객체 생성
        Car myCar = new Car();

        //Car 객체의 필드값 읽기
        System.out.println("모델명: " + myCar.model);
        System.out.println("시동여부: " + myCar.start);
        System.out.println("현재속도: " + myCar.speed);
    }
}
```

모델명: null 시동여부: false 현재속도: 0

#### ♡ 다음과 같이 클래스가 주어졌을 때, Car 클래스의 멤버 변수와 생성자를 정의하세요.

```
package ch06.sec07.exam01;
public class Car {
package ch06.sec07.exam01;
public class CarExample {
        public static void main(String[] args) {
                 Car myCar = new Car("그랜저", "검정", 250);
                 //Car myCar = new Car(); //기본 생성자 호출 못함
```

## Car.java

```
public class Car {
    String model:
    String color:
    int maxSpeed:

    //생성자 선언
    Car(String m, String c, int s) {
        model = m;
        color = c;
        maxSpeed = s;
    }
}
```

```
package ch06.sec07.exam01;

public class CarExample {
    public static void main(String[] args) {

        Car myCar = new Car("그랜저", "검정", 250);
        //Car myCar = new Car(); //기본 생성자 호출 못함
    }
}
```

#### ♡ 다음 Korean 클래스에 생성자의 매개변수로 name, ssn 필드를 초기화하는 생성자를 정의하세요.

```
package ch06.sec07.exam02;

public class Korean {
    //필드 선언
    String nation = "대한민국";
    String name;
    String ssn;

    //생성자 선언
}
```

## Korean.java

```
package ch06.sec07.exam02;
public class Korean {
        //필드 선언
        String nation = "대한민국";
        String name;
        String ssn;
        //생성자 선언
         public Korean(String n, String s) {
          name = n;
          ssn = s;
```

#### ♡ 다음과 같이 출력되도록 앞에서 만든 Korean의 인스턴스를 2개 만들어 운영하세요.

k1.nation : 대한민국 k1.name : 박자바

k1.ssn: 011225-1234567

k2.nation : 대한민국 k2.name : 김자바

k2.ssn: 930525-0654321

## KoreanExample.java

```
package ch06.sec07.exam02;
public class KoreanExample {
         public static void main(String[] args) {
                 //Korean 객체 생성
                 Korean k1 = new Korean("박자바", "011225-1234567");
                 //Korean 객체 데이터 읽기
                 System.out.println("k1.nation: " + k1.nation);
                 System.out.println("k1.name: " + k1.name);
                 System.out.println("k1.ssn: " + k1.ssn);
                 System.out.println();
                 //또 다른 Korean 객체 생성
                 Korean k2 = new Korean("김자바", "930525-0654321");
                 //또 다른 Korean 객체 데이터 읽기
                 System.out.println("k2.nation: " + k2.nation);
                 System.out.println("k2.name: " + k2.name);
                 System.out.println("k2.ssn: " + k2.ssn);
```

#### ♡ Korean 클래스를 다음과 같이 정의했을 때, 생성자에서 name, ssn 필드를 초기화하세요.

```
package ch06.sec07.exam03;

public class Korean {
    // 필드 선언
    String nation = "대한민국";
    String ssn;

    // 생성자 선언
    public Korean(String name, String ssn) {

    }
}
```

## Korean.java

```
package ch06.sec07.exam03;
public class Korean {
  // 필드 선언
  String nation = "대한민국";
  String name;
  String ssn;
  // 생성자 선언
  public Korean(String name, String ssn) {
     this.name = name;
     this.ssn = ssn;
```

#### Car 클래스가 다음과 같이 정의되어 있다. 다음을 만족하는 생성자를 추가하세요.

- 매개변수가 없는 생성자(디폴트 생성자)
- 매개변수로 model만 가지는 생성자
- 매개변수로 model과 color를 가지는 생성자
- 매개변수로 model, color, maxSpeed를 가지는 생성자

```
package ch06.sec07.exam04;

public class Car {
    //필드 선언
    String company = "현대자동차";
    String model;
    String color;
    int maxSpeed;
}
```

#### 클래스

## Car.java

```
package ch06.sec07.exam04;
public class Car {
          //필드 선언
          String company = "현대자동차";
          String model;
          String color;
          int maxSpeed;
          //생성자 선언
          Car() {}
          Car(String model) {
                    this.model = model;
          Car(String model, String color) {
                    this.model = model;
                    this.color = color;
```

#### ♥ 앞에서 정의한 Car 클래스를 이용하여, 다음 출력이 나오도록 운영하세요.

```
package ch06.sec07.exam04;
public class CarExample {
    public static void main(String[] args) {
    }
}
```

car1.company : 현대자동차
car2.company : 현대자동차
car2.model : 자가용

car3.company : 현대자동차
car3.model : 자가용
car3.color : 빨강

car4.company : 현대자동차
car4.model : 택시
car4.color : 검정
car4.maxSpeed : 200

14

## CarExample.java

```
package ch06.sec07.exam04;
public class CarExample {
          public static void main(String[] args) {
                     Car car1 = new Car();
                     System.out.println("car1.company: " + car1.company);
                     System.out.println();
                     Car car2 = new Car("자가용");
                     System.out.println("car2.company: " + car2.company);
                     System.out.println("car2.model: " + car2.model);
                     System.out.println();
                     Car car3 = new Car("자가용", "빨강");
                     System.out.println("car3.company: " + car3.company);
                     System.out.println("car3.model: " + car3.model);
                     System.out.println("car3.color: " + car3.color);
                     System.out.println();
                     Car car4 = new Car("택시", "검정", 200);
                     System.out.println("car4.company: " + car4.company);
                     System.out.println("car4.model: " + car4.model);
                     System.out.println("car4.color: " + car4.color);
                     System.out.println("car4.maxSpeed: " + car4.maxSpeed);
```

# ○ Car 클래스가 다음과 같이 정의된 경우, 다른 생성자를 호출하여 필드를 초기화하도록 생성자를 수정하세요.

```
Car(String model, String color, int maxSpeed) {
package ch06.sec07.exam05;
                                                                             this.model = model;
public class Car {
                                                                              this.color = color;
         // 필드 선언
                                                                              this.maxSpeed = maxSpeed;
         String company = "현대자동차";
         String model;
         String color;
                                                              car1.company : 현대자동차
         int maxSpeed;
         Car(String model) {
                                                              car2.company : 현대자동차
                                                              car2.model: 자가용
                                                              car3.company : 현대자동차
         Car(String model, String color) {
                                                              car3.model: 자가용
                                                              car3.color: 빨강
                                                              car4.company : 현대자동차
                                                              car4.model: 택시
                                                              car4.color: 검정
                                                              car4.maxSpeed: 200
```

#### 클래스

## Car.java

```
package ch06.sec07.exam05;
                                                             Car(String model, String color, int maxSpeed) {
                                                                      this.model = model;
public class Car {
                                                                      this.color = color;
        // 필드 선언
                                                                      this.maxSpeed = maxSpeed;
        String company = "현대자동차";
        String model;
        String color;
        int maxSpeed;
        Car(String model) {
                 //20라인 생성자 호출
                 this(model, "손색", 250);
        Car(String model, String color) {
                 //20라인 생성자 호출
                 this(model, color, 250);
```

#### ♡ 앞에서 정의한 Car 클래스를 이용하여, 다음 출력이 나오도록 운영하세요.

```
package ch06.sec07.exam05;

public class CarExample {
         public static void main(String[] args) {
          }
}
```

car1.company: 현대자동차

car2.company: 현대자동차

car2.model: 자가용

car3.company: 현대자동차

car3.model : 자가용

car3.color : 빨강

car4.company : 현대자동차

car4.model : 택시 car4.color : 검정

car4.maxSpeed: 200

## CarExample.java

```
package ch06.sec07.exam05;
public class CarExample {
          public static void main(String[] args) {
                     Car car1 = new Car("자가용");
                     System.out.println("car1.company: " + car1.company);
                     System.out.println("car1.model: " + car1.model);
                     System.out.println();
                     Car car2 = new Car("자가용", "빨강");
                     System.out.println("car2.company: " + car2.company);
                     System.out.println("car2.model: " + car2.model);
                     System.out.println("car2.color: " + car2.color);
                     System.out.println();
                     Car car3 = new Car("택시", "검정", 200);
                     System.out.println("car3.company: " + car3.company);
                     System.out.println("car3.model: " + car3.model);
                     System.out.println("car3.color: " + car3.color);
                     System.out.println("car3.maxSpeed: " + car3.maxSpeed);
```

#### 다음 결과가 나오도록 Calculator 클래스를 정의하세요.

```
package ch06.sec08.exam01;
public class CalculatorExample {
          public static void main(String[] args) {
                     Calculator myCalc = new Calculator();
                                                                          //Calculator 객체 생성
                     myCalc.powerOn();
                                                    //리턴값이 없는 powerOn() 메소드 호출
                     int result1 = myCalc.plus(5, 6);
                     System.out.println("result1: " + result1);
                     int x = 10;
                     int y = 4;
                     double result2 = myCalc.divide(x, y);
                     System.out.println("result2: " + result2);
                     myCalc.powerOff();
```

```
전원을 켭니다.
result1: 11
result2: 2.5
전원을 끕니다.
```

## Calculator.java

```
package ch06.sec08.exam01;
public class Calculator {
          //리턴값이 없는 메소드 선언
          void powerOn() {
                     System.out.println("전원을 켭니다.");
          //리턴값이 없는 메소드 선언
          void powerOff() {
                     System.out.println("전원을 끕니다.");
          //호출 시 두 정수 값을 전달받고,
          //호출한 곳으로 결과값 int를 리턴하는 메소드 선언
          int plus(int x, int y) {
                     int result = x + y;
                     return result; //리턴값 지정;
          //호출 시 두 정수 값을 전달받고,
          //호출한 곳으로 결과값 double을 리턴하는 메소드선언
          double divide(int x, int y) {
                     double result = (double)x / (double)y;
                     return result; //리턴값 지정;
```

## CalculatorExample.java

```
package ch06.sec08.exam01;
public class CalculatorExample {
         public static void main(String[] args) {
                  Calculator myCalc = new Calculator();
                                                                //Calculator 객체 생성
                  myCalc.powerOn();
                                              //리턴값이 없는 powerOn() 메소드 호출
                  //plus 메소드 호출 시 5와 6을 매개값으로 제공하고,
                  //덧셈 결과를 리턴 받아 result1 변수에 대입
                  int result1 = myCalc.plus(5, 6);
                  System.out.println("result1: " + result1);
                  int x = 10;
                  int y = 4;
                  //divide() 메소드 호출 시 변수 x와 y의 값을 매개값으로 제공하고,
                  //나눗셈 결과를 리턴 받아 result2 변수에 대입
                  double result2 = myCalc.divide(x, y);
                  System.out.println("result2: " + result2);
                  //리턴값이 없는 powerOff() 메소드 호출
                                                                   전원을 켭니다.
                  myCalc.powerOff();
                                                                   result1: 11
                                                                   result2: 2.5
                                                                   전원을 끕니다.
```

result2: 15 result3: 15 result4: 15

#### 다음 결과가 나오도록 Computer 클래스를 작성하세요.

```
package ch06.sec08.exam02;
public class ComputerExample {
           public static void main(String[] args) {
                     Computer myCom = new Computer();
                                                                            //Computer 객체 생성
                     int result1 = myCom.sum(1, 2, 3);
                     System.out.println("result1: " + result1);
                     int result2 = myCom.sum(1, 2, 3, 4, 5);
                     System.out.println("result2: " + result2);
                     int[] values = { 1, 2, 3, 4, 5 };
                     int result3 = myCom.sum(values);
                     System.out.println("result3: " + result3);
                     int result4 = myCom.sum(new int[] { 1, 2, 3, 4, 5 });
                     System.out.println("result4: " + result4);
      result1: 6
```

## Computer.java

```
package ch06.sec08.exam02;
public class Computer {
        //가변길이 매개변수를 갖는 메소드 선언
         int sum(int ... values) {
                  //sum 변수 선언
                  int sum = 0;
                  //values는 배열 타입의 변수처럼 사용
                  for (int i = 0; i < values.length; i++) {
                           sum += values[i];
                  //합산 결과를 리턴
                  return sum;
```

#### ♡ 다음과 같이 Car 클래스의 기본 정의가 있다. 여기에 주석부분을 만족하도록 클래스의 메서드를 완성 하세요.

```
package ch06.sec08.exam03;
public class Car {
        int gas;
       // 리턴값이 없는 메소드로 매개값을 받아서 gas 필드값을 변경
       void setGas(int gas) {
       // 리턴값이 boolean인 메소드로 gas 필드값이 0이면 false를, 0이 아니면 true를 리턴
       // 리턴하기전 조건에 따라 "gas가 없습니다" 또는 "gas가 없습니다" 출력
        boolean isLeftGas() {
       // 무한 루프를 돌면서 루프를 한 번 돌 때마다 gas의 값은 1 감소
       // 주행이 가능한 상태라면 "달립니다.(gas잔량: xxx)" 출력
       // 주행할 수 없는 상태라면 "멈춥니다.(gas잔량: xxx)" 출력 후 메소드를 종료
       void run() {
```

## Car.java

```
package ch06.sec08.exam03;
public class Car {
        //필드 선언
        int gas;
        //리턴값이 없는 메소드로 매개값을 받아서 gas 필드값을 변경
        void setGas(int gas) {
                  this.gas = gas;
        //리턴값이 boolean인 메소드로 gas 필드값이 0이면 false를, 0이 아니면 true를 리턴
         boolean isLeftGas() {
                  if (gas == 0) {
                           System.out.println("gas가 없습니다.");
                           return false; // false를 리턴하고 메소드 종료
                  System.out.println("gas가 있습니다.");
                  return true; // true를 리턴하고 메소드 종료
```

## Car.java

#### ♥ 앞에서 정의한 Car 클래스를 이용하여 다음과 같이 출력되도록, 운영 클래스를 정의하세요.

```
package ch06.sec08.exam03;
public class CarExample {
    public static void main(String[] args) {
}
```

```
gas가 있습니다.
출발합니다.
달립니다.(gas잔량:5)
달립니다.(gas잔량:4)
달립니다.(gas잔량:3)
달립니다.(gas잔량:2)
달립니다.(gas잔량:1)
멈춥니다.(gas잔량:0)
gas를 주입하세요.
```

#### 클래스

## CarExample.java

```
package ch06.sec08.exam03;
public class CarExample {
        public static void main(String[] args) {
                 //Car 객체 생성
                 Car myCar = new Car();
                 //리턴값이 없는 setGas() 메소드 호출
                 myCar.setGas(5);
                 //isLeftGas() 메소드를 호출해서 받은 리턴값이 true일 경우 if 블록 실행
                 if(myCar.isLeftGas()) {
                         System.out.println("출발합니다.");
                                                                gas가 있습니다.
                         //리턴값이 없는 run() 메소드 호출
                                                                출발합니다.
                         myCar.run();
                                                                달립니다.(gas잔량:5)
                                                                달립니다.(gas잔량:4)
                 System.out.println("gas를 주입하세요.");
                                                                달립니다.(gas잔량:3)
                                                                달립니다.(gas잔량:2)
                                                                달립니다.(gas잔량:1)
                                                                멈춥니다.(gas잔량:0)
                                                                gas를 주입하세요.
```

#### ○ 다음 결과가 나오도록 Calculator 클래스를 정의하세요.

```
package ch06.sec08.exam04;
public class CalculatorExample {
        public static void main(String[] args) {
                 //객체 생성
                 Calculator myCalcu = new Calculator();
                 //정사각형의 넓이 구하기
                 double result1 = myCalcu.areaRectangle(10);
                 //직사각형의 넓이 구하기
                 double result2 = myCalcu.areaRectangle(10, 20);
                 System.out.println("정사각형 넓이=" + result1);
                 System.out.println("직사각형 넓이=" + result2);
```

```
정사각형 넓이=100.0
직사각형 넓이=200.0
```

## Calculator.java

- 다음 조건을 만족하는 Car 클래스를 완성하세요.
  - o model 필드는 생성자로 초기화
  - o speed 필드의 getter/setter 메서드 정의
  - o run() 메서드 호출 시 다음과 같이 출력
    - xxx(model 값)가 달립니다.(시속: yyy(speed 값)km/h)

```
package ch06.sec09;

public class Car {
    //필드 선언
    String model;
    int speed;
}
```

Car 클래스가 올바르게 동작하는지 확인하는 클래스를 작성하세요.

## Car.java

```
package ch06.sec09;
public class Car {
         //필드 선언
         String model;
         int speed;
         //생성자 선언
         Car(String model) {
                   this.model = model; //매개변수를 필드에 대입(this 생략 불가)
         //메소드 선언
         void setSpeed(int speed) {
                   this.speed = speed; //매개변수를 필드에 대입(this 생략 불가)
         void run() {
                   this.setSpeed(100);
                   System.out.println(this.model + "가 달립니다.(시속:" + this.speed + "km/h)");
```

## CarExample.java

```
package ch06.sec09;

public class CarExample {
    public static void main(String[] args) {
        Car myCar = new Car("포르쉐");
        Car yourCar = new Car("벤츠");

        myCar.run();
        yourCar.run();
    }
}
```

```
포르쉐가 달립니다.(시속:100km/h)
벤츠가 달립니다.(시속:100km/h)
```



## Calculator 클래스가 다음과 같이 정의된 경우 CaculatorExample 코드를 완성하세요.

#### Calculator의 멤버를 이용해서 완성함

```
package ch06.sec10.exam01;

public class Calculator {
    static double pi = 3.14159;

    static int plus(int x, int y) {
        return x + y;
    }

    static int minus(int x, int y) {
        return x - y;
    }
}
```

```
package ch06.sec10.exam01;

public class CalculatorExample {
    public static void main(String[] args) {
        double result1 = 10 * 10 * _____;
        // 원의 넓이
        int result2 = _____; // 10, 5의 합
        int result3 = _____; // 10, 5의 차

        System.out.println("result1 : " + result1);
        System.out.println("result2 : " + result2);
        System.out.println("result3 : " + result3);
    }
}
```



## CalculatorExample.java

```
package ch06.sec10.exam01;

public class CalculatorExample {
    public static void main(String[] args) {
        double result1 = 10 * 10 * Calculator.pi;
        int result2 = Calculator.plus(10, 5);
        int result3 = Calculator.minus(10, 5);

        System.out.println("result1 : " + result1);
        System.out.println("result2 : " + result2);
        System.out.println("result3 : " + result3);
    }
}
```

```
result1: 314.159
result2: 15
result3: 5
```