Java의 정석定石 3판 - 연습문제 풀이

33

# Chapter

객체지향 프로그래밍 I

Object-oriented Programming I

## [ 연습문제 ]

[6-1] 다음과 같은 멤버변수를 갖는 SutdaCard클래스를 정의하시오.

|  |  |  |
| --- | --- | --- |
| 타 입 | 변수명 | 설 명 |
| int | num | 카드의 숫자.(1~10사이의 정수) |
| boolean | isKwang | 광(光)이면 true, 아니면 false |

[6-2] 문제6-1에서 정의한 SutdaCard클래스에 두 개의 생성자와 info()를 추가해서 실행 결과와 같은 결과를 얻도록 하시오.

class Exercise6\_2 {

public static void main(String args[]) { SutdaCard card1 = new SutdaCard(3, false); SutdaCard card2 = new SutdaCard();

System.out.println(card1.info()); System.out.println(card2.info());

}

}

class SutdaCard {

**/\***

**(1)** 알맞은 코드를 넣어 완성하시오**.**

**\*/**

}

[연습문제]**/ch6/Exercise6\_2.java**

3

1K

[실행결과]

[6-3] 다음과 같은 멤버변수를 갖는 Student클래스를 정의하시오.

|  |  |  |
| --- | --- | --- |
| 타 입 | 변수명 | 설 명 |
| String | name | 학생이름 |
| int | ban | 반 |
| int | no | 번호 |
| int | kor | 국어점수 |
| int | eng | 영어점수 |
| int | math | 수학점수 |

[6-4] 문제6-3에서 정의한 Student클래스에 다음과 같이 정의된 두 개의 메서드

getTotal()과 getAverage()를 추가하시오.

1. 메서드명 : getTotal

기 능 : 국어(kor), 영어(eng), 수학(math)의 점수를 모두 더해서 반환한다. 반환타입 : int

매개변수 : 없음

1. 메서드명 : getAverage

기 능 : 총점(국어점수+영어점수+수학점수)을 과목수로 나눈 평균을 구한다. 소수점 둘째자리에서 반올림할 것.

반환타입 : float 매개변수 : 없음

class Exercise6\_4 {

public static void main(String args[]) { Student s = new Student();

s.name = "홍길동";

s.ban = 1;

s.no = 1;

s.kor = 100;

s.eng = 60;

s.math = 76;

System.out.println("이름:"+s.name); System.out.println("총점:"+s.getTotal()); System.out.println("평균:"+s.getAverage());

}

}

class Student {

**/\***

**(1)** 알맞은 코드를 넣어 완성하시오**.**

**\*/**

}

[연습문제]**/ch6/Exercise6\_4.java**

이름:홍길동 총점:236 평균:78.7

[실행결과]

[6-5] 다음과 같은 실행결과를 얻도록 Student클래스에 생성자와 info()를 추가하시오.

class Exercise6\_5 {

public static void main(String args[]) {

Student s = new Student("홍길동",1,1,100,60,76);

System.out.println(s.info());

}

}

class Student {

**/\***

**(1)** 알맞은 코드를 넣어 완성하시오**.**

**\*/**

}

[연습문제]**/ch6/Exercise6\_5.java**

홍길동,1,1,100,60,76,236,78.7

[실행결과]

[6-6] 두 점의 거리를 계산하는 getDistance()를 완성하시오.

[Hint] 제곱근 계산은 Math.sqrt(double a)를 사용하면 된다.

class Exercise6\_6 {

// 두 점 (x,y)와 (x1,y1)간의 거리를 구한다.

static double getDistance(int x, int y, int x1, int y1) {

**/\***

**(1)** 알맞은 코드를 넣어 완성하시오**.**

**\*/**

}

public static void main(String args[]) { System.out.println(getDistance(1,1,2,2));

}

}

[연습문제]**/ch6/Exercise6\_6.java**

1.4142135623730951

[실행결과]

[6-7] 문제6-6에서 작성한 클래스메서드 getDistance()를 MyPoint클래스의 인스턴스메서 드로 정의하시오.

class MyPoint { int x;

int y;

MyPoint(int x, int y) { this.x = x;

this.y = y;

}

**/\***

**(1)** 인스턴스메서드 **getDistance**를 작성하시오**.**

**\*/**

}

class Exercise6\_7 {

public static void main(String args[]) { MyPoint p = new MyPoint(1,1);

// p와 (2,2)의 거리를 구한다. System.out.println(p.getDistance(2,2));

}

}

[연습문제]**/ch6/Exercise6\_7.java**

1.4142135623730951

[실행결과]

[6-8] 다음의 코드에 정의된 변수들을 종류별로 구분해서 적으시오.

class PlayingCard { int kind;

int num;

static int width; static int height;

PlayingCard(int k, int n) { kind = k;

num = n;

}

public static void main(String args[]) { PlayingCard card = new PlayingCard(1,1);

}

}

* 클래스변수(static변수) :
* 인스턴스변수 :
* 지역변수 :

[6-9] 다음은 컴퓨터 게임의 병사(marine)를 클래스로 정의한 것이다. 이 클래스의 멤버 중에 static을 붙여야 하는 것은 어떤 것들이고 그 이유는 무엇인가?

(단, 모든 병사의 공격력과 방어력은 같아야 한다.)

class Marine {

int x=0, y=0; // Marine의 위치좌표(x,y) int hp = 60; // 현재 체력

int weapon = 6; // 공격력

int armor = 0; // 방어력

void weaponUp() { weapon++;

}

void armorUp() { armor++;

}

void move(int x, int y) { this.x = x;

this.y = y;

}

}

[6-10] 다음 중 생성자에 대한 설명으로 옳지 않은 것은? (모두 고르시오)

1. 모든 생성자의 이름은 클래스의 이름과 동일해야한다.
2. 생성자는 객체를 생성하기 위한 것이다.
3. 클래스에는 생성자가 반드시 하나 이상 있어야 한다.
4. 생성자가 없는 클래스는 컴파일러가 기본 생성자를 추가한다.
5. 생성자는 오버로딩 할 수 없다.

[6-11] 다음 중 this에 대한 설명으로 맞지 않은 것은? (모두 고르시오)

1. 객체 자신을 가리키는 참조변수이다.
2. 클래스 내에서라면 어디서든 사용할 수 있다.
3. 지역변수와 인스턴스변수를 구별할 때 사용한다.
4. 클래스 메서드 내에서는 사용할 수 없다.

[6-12] 다음 중 오버로딩이 성립하기 위한 조건이 아닌 것은? (모두 고르시오)

1. 메서드의 이름이 같아야 한다.
2. 매개변수의 개수나 타입이 달라야 한다.
3. 리턴타입이 달라야 한다.
4. 매개변수의 이름이 달라야 한다.

[6-13] 다음 중 아래의 add메서드를 올바르게 오버로딩 한 것은? (모두 고르시오)

long add(int a, int b) { return a+b;}

* 1. long add(int x, int y) { return x+y;}
  2. long add(long a, long b) { return a+b;}
  3. int add(byte a, byte b) { return a+b;}
  4. int add(long a, int b) { return (int)(a+b);}

[6-14] 다음 중 초기화에 대한 설명으로 옳지 않은 것은? (모두 고르시오)

1. 멤버변수는 자동 초기화되므로 초기화하지 않고도 값을 참조할 수 있다.
2. 지역변수는 사용하기 전에 반드시 초기화해야 한다.
3. 초기화 블럭보다 생성자가 먼저 수행된다.
4. 명시적 초기화를 제일 우선적으로 고려해야 한다.
5. 클래스변수보다 인스턴스변수가 먼저 초기화된다.

[6-15] 다음중 인스턴스변수의 초기화 순서가 올바른 것은?

1. 기본값-명시적초기화-초기화블럭-생성자
2. 기본값-명시적초기화-생성자-초기화블럭
3. 기본값-초기화블럭-명시적초기화-생성자
4. 기본값-초기화블럭-생성자-명시적초기화

[6-16] 다음 중 지역변수에 대한 설명으로 옳지 않은 것은? (모두 고르시오)

1. 자동 초기화되므로 별도의 초기화가 필요없다.
2. 지역변수가 선언된 메서드가 종료되면 지역변수도 함께 소멸된다.
3. 매서드의 매개변수로 선언된 변수도 지역변수이다.
4. 클래스변수나 인스턴스변수보다 메모리 부담이 적다.
5. 힙(heap)영역에 생성되며 가비지 컬렉터에 의해 소멸된다.

[6-17] 호출스택이 다음과 같은 상황일 때 옳지 않은 설명은? (모두 고르시오)

|  |
| --- |
|  |
| println |
| method1 |
| method2 |
| main |

1. 제일 먼저 호출스택에 저장된 것은 main메서드이다.
2. println메서드를 제외한 나머지 메서드들은 모두 종료된 상태이다.
3. method2메서드를 호출한 것은 main메서드이다.
4. println메서드가 종료되면 method1메서드가 수행을 재개한다.
5. main-method2-method1-println의 순서로 호출되었다.
6. 현재 실행중인 메서드는 println 뿐이다.

[6-18] 다음의 코드를 컴파일하면 에러가 발생한다. 이유를 설명하시오.

컴파일 에러가 발생하는 라인과 그

class MemberCall { int iv = 10; static int cv = 20;

int iv2 = cv;

**static int cv2 = iv;**

**//** 라인 **A**

static void staticMethod1() { System.out.println(cv); **System.out.println(iv); //** 라인 **B**

}

void instanceMethod1() { System.out.println(cv); **System.out.println(iv); //** 라인 **C**

}

static void staticMethod2() { staticMethod1(); **instanceMethod1(); //** 라인 **D**

}

void instanceMethod2() { **staticMethod1();** instanceMethod1();

}

**//** 라인 **E**

}

[6-19] 다음 코드의 실행 결과를 예측하여 적으시오.

class Exercise6\_19

{

public static void change(String str) { str += "456";

}

public static void main(String[] args)

{

String str = "ABC123"; System.out.println(str); change(str);

System.out.println("After change:"+str);

}

}

[연습문제]**/ch6/Exercise6\_19.java**

[주의] Math.random()을 사용하는 경우 실행결과와 다를 수 있음.

메서드명 : shuffle

기 능 : 주어진 배열에 담긴 값의 위치를 바꾸는 작업을 반복하여 뒤섞이게 한다. 처리한 배열을 반환한다.

반환타입 : int[]

매개변수 : int[] arr - 정수값이 담긴 배열

class Exercise6\_20

{

**/\***

**(1) shuffle**메서드를 작성하시오**.**

**\*/**

public static void main(String[] args)

{

int[] original = {1,2,3,4,5,6,7,8,9}; System.out.println(java.util.Arrays.toString(original));

int[] result = shuffle(original); System.out.println(java.util.Arrays.toString(result));

}

}

[연습문제]**/ch6/Exercise6\_20.java**

[1, 2, 3, 4, 5, 6, 7, 8, 9]

[4, 6, 8, 3, 2, 9, 7, 1, 5]

[실행결과]

[6-21] Tv클래스를 주어진 로직대로 완성하시오. 완성한 후에 실행해서 주어진 실행결과 와 일치하는지 확인하라.

[참고] 코드를 단순히 하기 위해서 유효성검사는 로직에서 제외했다.

class MyTv {

boolean isPowerOn; int channel;

int volume;

final int MAX\_VOLUME = 100; final int MIN\_VOLUME = 0; final int MAX\_CHANNEL = 100; final int MIN\_CHANNEL = 1;

void turnOnOff() {

**// (1) isPowerOn**의 값이 **true**면 **false**로**, false**면 **true**로 바꾼다**.**

}

[연습문제]**/ch6/Exercise6\_21.java**

void volumeUp() {

**// (2) volume**의 값이 **MAX\_VOLUME**보다 작을 때만 값을 **1**증가시킨다**.**

}

void volumeDown() {

**// (3) volume**의 값이 **MIN\_VOLUME**보다 클 때만 값을 **1**감소시킨다**.**

}

void channelUp() {

**// (4) channel**의 값을 **1**증가시킨다**.**

**//** 만일 **channel**이 **MAX\_CHANNEL**이면**, channel**의 값을 **MIN\_CHANNEL**로 바꾼다**.**

}

void channelDown() {

**// (5) channel**의 값을 **1**감소시킨다**.**

**//** 만일 **channel**이 **MIN\_CHANNEL**이면**, channel**의 값을 **MAX\_CHANNEL**로 바꾼다**.**

}

} // class MyTv

class Exercise6\_21 {

public static void main(String args[]) { MyTv t = new MyTv();

t.channel = 100;

t.volume = 0;

System.out.println("CH:"+t.channel+", VOL:"+ t.volume);

t.channelDown(); t.volumeDown();

System.out.println("CH:"+t.channel+", VOL:"+ t.volume);

t.volume = 100; t.channelUp(); t.volumeUp();

System.out.println("CH:"+t.channel+", VOL:"+ t.volume);

}

}

CH:100, VOL:0 CH:99, VOL:0 CH:100, VOL:100

[실행결과]

메서드명 : isNumber

기 능 : 주어진 문자열이 모두 숫자로만 이루어져있는지 확인한다. 모두 숫자로만 이루어져 있으면 true를 반환하고,

그렇지 않으면 false를 반환한다.

만일 주어진 문자열이 null이거나 빈문자열“”이라면 false를 반환한다. 반환타입 : boolean

매개변수 : String str - 검사할 문자열

[Hint] String클래스의 charAt(int i)메서드를 사용하면 문자열의 i번째 위치한 문자를 얻을 수 있다.

class Exercise6\_22 {

**/\***

**(1) isNumber**메서드를 작성하시오**.**

**\*/**

public static void main(String[] args) { String str = "123";

System.out.println(str+"는 숫자입니까? "+isNumber(str));

str = "1234o";

System.out.println(str+"는 숫자입니까? "+isNumber(str));

}

}

[연습문제]**/ch6/Exercise6\_22.java**

123는 숫자입니까? true 1234o는 숫자입니까? false

[실행결과]

[6-23] 다음과 같이 정의된 메서드를 작성하고 테스트하시오.

메서드명 : max

기 능 : 주어진 int형 배열의 값 중에서 제일 큰 값을 반환한다.

만일 주어진 배열이 null이거나 크기가 0인 경우, -999999를 반환한다. 반환타입 : int

매개변수 : int[] arr - 최대값을 구할 배열

class Exercise6\_23{

**/\***

**(1) max**메서드를 작성하시오**.**

**\*/**

[연습문제]**/ch6/Exercise6\_23.java**

public static void main(String[] args)

{

int[] data = {3,2,9,4,7}; System.out.println(java.util.Arrays.toString(data)); System.out.println("최대값:"+max(data));

System.out.println("최대값:"+max(null));

System.out.println("최대값:"+max(new int[]{})); // 크기가 0인 배열

}

}

[3, 2, 9, 4, 7]

최대값:9 최대값:-999999 최대값:-999999

[실행결과]

[6-24] 다음과 같이 정의된 메서드를 작성하고 테스트하시오.

메서드명 : abs

기 능 : 주어진 값의 절대값을 반환한다. 반환타입 : int

매개변수 : int value

class Exercise6\_24

{

**/\***

**(1) abs**메서드를 작성하시오**.**

**\*/**

public static void main(String[] args)

{

int value = 5;

System.out.println(value+"의 절대값:"+abs(value)); value = -10;

System.out.println(value+"의 절대값:"+abs(value));

}

}

[연습문제]**/ch6/Exercise6\_24.java**

5의 절대값:5

-10의 절대값:10

[실행결과]