

[5-1] 다음은 배열을 선언하거나 초기화 한 것이다. 잘못된 것을 고르고 그 이유를 설명

하시오.

2차원 배열 선언 방식 (타입 뒤에 두개 or 변수 뒤에 두개 or 타입 뒤에 하나 변수

- a. `int[] arr[];` 뒤에 하나)
- b. `int[] arr = {1,2,3};` // 마지막의 쉼표 ',' 는 있어도 상관없음.
- c. `int[] arr = new int[5];`
- d. `int[] arr = new int[5]{1,2,3,4,5};` // 두 번째 대괄호[]에 숫자 넣으면 안됨.
- e. `int arr[5];` // 배열을 선언할 때는 배열의 크기를 지정할 수 없음.
- f. `int[] arr[] = new int[3][];` 가변 배열(행마다 다른 열 길이 생성)

[정답] d, e

[해설] d. `int[] arr = new int[] {1,2,3,4,5};`에서는 대괄호[]안에 배열의 크기를 지정할 수 없다. 괄호{}안의 데이터의 개수에 따라 자동적으로 결정되기 때문이다.

```
import java.util.*;

public class Ex {
    public static void main(String[] args) {
        // 2차원 배열 arr의 모든 값의 총합과 평균을 구하는 프로그램

        int[][] arr = {
            {5, 5, 5, 5, 5},
            {10, 10, 10, 10, 10},
            {20, 20, 20, 20, 20},
            {30, 30, 30, 30, 30}
        };

        int total = 0;
        float average = 0;

        for (int i = 0; i < arr.length; i++) {
            for (int j = 0; j < arr[i].length; j++) {
                total += arr[i][j];
            }
        }

        average = (float) (total) / (arr.length * arr[0].length);

        System.out.println("total=" + total);
        System.out.println("average=" + average);
    }
}
```

```

import java.util.*;

public class Ex {
    public static void main(String[] args) {
        // 1과 9 사이의 중복되지 않은 숫자로 이루어진 3자리 숫자를 만들어내기
        int[] ballArr = {1, 2, 3, 4, 5, 6, 7, 8, 9};
        int[] ball3 = new int[3];

        // 배열 ballArr의 임의의 요소를 골라서 위치를 바꾼다.
        for (int i = 0; i < ballArr.length; i++) {
            int j = (int) (Math.random() * ballArr.length);
            int temp = 0;

            temp = ballArr[i];
            ballArr[i] = ballArr[j];
            ballArr[j] = temp;
        }

        // 배열 ballArr의 앞에서 3개의 수를 배열 ball3으로 복사한다.
        System.arraycopy(ballArr, 0, ball3, 0, 3);

        // 출력
        for (int i = 0; i < ball3.length; i++) {
            System.out.print(ball3[i]);
        }
    }
}

```

```

import java.util.*;

public class Ex {
    public static void main(String[] args) {
        // 거스름돈 계산하기
        int[] coinUnit = {500, 100, 50, 10};
        int money = 2680;

        System.out.println("money=" + money);

        for (int i = 0; i < coinUnit.length; i++) {
            int temp = money / coinUnit[i];
            money = money % coinUnit[i];
            System.out.printf("%d원: %d\n", coinUnit[i], temp);
        }
    }
}

```

```

import java.util.*;

public class Ex {
    public static void main(String[] args) {
        // 거스름돈 계산하기 + 동전의 갯수 추가

        // 문자열을 숫자로 변환한다. 입력한 값이 숫자가 아닐 경우 예외가 발생한다.
        Scanner scan = new Scanner(System.in);
        System.out.print("금액을 입력하세요 : ");
        String amount = scan.nextLine();
        int money = Integer.parseInt(amount);

        System.out.println("money=" + money);

        int[] coinUnit = {500, 100, 50, 10}; // 동전의 단위
        int[] coin = {5, 5, 5, 5}; //단위별 동전의 개수

        for (int i = 0; i < coinUnit.length; i++) {
            int coinNum = 0;

            // 필요한 동전의 갯수
            coinNum = money / coinUnit[i];

            // 충분한 동전이 없다면 배열 coin에 있는 만큼만 coin에서 빼고,
            // 있다면 배열 coin 에서 coinNum만큼의 동전을 뺀다.
            if (coin[i] < coinNum) {
                coinNum = coin[i];
                coin[i] = coin[i] - coinNum;
            } else {
                coin[i] = coin[i] - coinNum;
            }

            // 금액에서 동전의 개수(coinNum)와 동전단위를 곱한 값을 뺀다
            money = money - (coinNum * coinUnit[i]);

            System.out.println(coinUnit[i]+"원: " + coinNum);

        }

        if (money > 0) {
            System.out.println("거스름 돈이 부족합니다.");
            System.exit(0); // 프로그램을 종료한다.
        }

        System.out.println("=남은 동전의 개수 =");

        for (int i = 0; i < coinUnit.length; i++) {
            System.out.println(coinUnit[i] + "원:" + coin[i]);
        }

    }
}

```

```

import java.util.*;

public class Ex {
    public static void main(String[] args) {
        // 거스름돈 계산하기 + 동전의 갯수 추가 (자판기 원리)

        // 문자열을 숫자로 변환한다. 입력한 값이 숫자가 아닐 경우 예외가 발생한다.
        Scanner scan = new Scanner(System.in);
        System.out.print("금액을 입력하세요 : ");
        String amount = scan.nextLine();
        int money = Integer.parseInt(amount);

        System.out.println("money=" + money);

        int[] coinUnit = {500, 100, 50, 10}; // 동전의 단위
        int[] coin = {5, 5, 5, 5}; //단위별 동전의 개수

        for (int i = 0; i < coinUnit.length; i++) {
            int coinNum = 0;

            // 필요한 동전의 갯수
            coinNum = money / coinUnit[i];

            // 충분한 동전이 없다면 배열 coin에 있는 만큼만 coin에서 빼고,
            // 있다면 배열 coin 에서 coinNum만큼의 동전을 뺀다.
            if (coin[i] < coinNum) {
                coinNum = coin[i];
                coin[i] = 0;
            } else {
                coin[i] = coin[i] - coinNum;
            }

            // 금액에서 동전의 개수(coinNum)와 동전단위를 곱한 값을 뺀다
            money = money - (coinNum * coinUnit[i]);

            System.out.println(coinUnit[i]+"원: " + coinNum);

        }

        if (money > 0) {
            System.out.println("거스름 돈이 부족합니다.");
            System.exit(0); // 프로그램을 종료한다.
        }

        System.out.println("=남은 동전의 개수 =");

        for (int i = 0; i < coinUnit.length; i++) {
            System.out.println(coinUnit[i] + "원:" + coin[i]);
        }

    }
}

```

```
import java.util.*;

public class Ex {
    public static void main(String[] args) {
        // answer의 데이터를 읽고 각 숫자의 개수를 세어서 개수만큼 '*'을 찍기
        int[] answer = {1, 4, 4, 3, 1, 4, 4, 2, 1, 3, 2};
        int[] counter = new int[4];

        for (int i = 0; i < answer.length; i++) {
            counter[answer[i] - 1]++;          // answer의 값은 1-4, counter의 인덱스는 0-3
        }

        for (int i = 0; i < counter.length; i++) {
            System.out.print(counter[i]);
            for (int j = 0; j < counter[i]; j++) {
                System.out.print("*");
            }
            System.out.println();
        }
    }
}
```

Resource: 자바의 정석
