

컴퓨터 프로그래밍 (Computer Programming)

이 선 순



10. 배열



목차

1. 배열의 이해

2. 2차원 배열

3. 배열의 활용

01

배열의 이해

01 배열의 이해

■ 배열(array)

- 여러 개의 변수를 나란히 나열한 것
- 배열은 그림과 같이 상자(변수)를 한 줄로 붙여놓은 개념
- 각 상자에 이름을 붙여서 aa[0], aa[1], aa[2], aa[3]과 같이 번호(첨자)를 붙여서 사용함

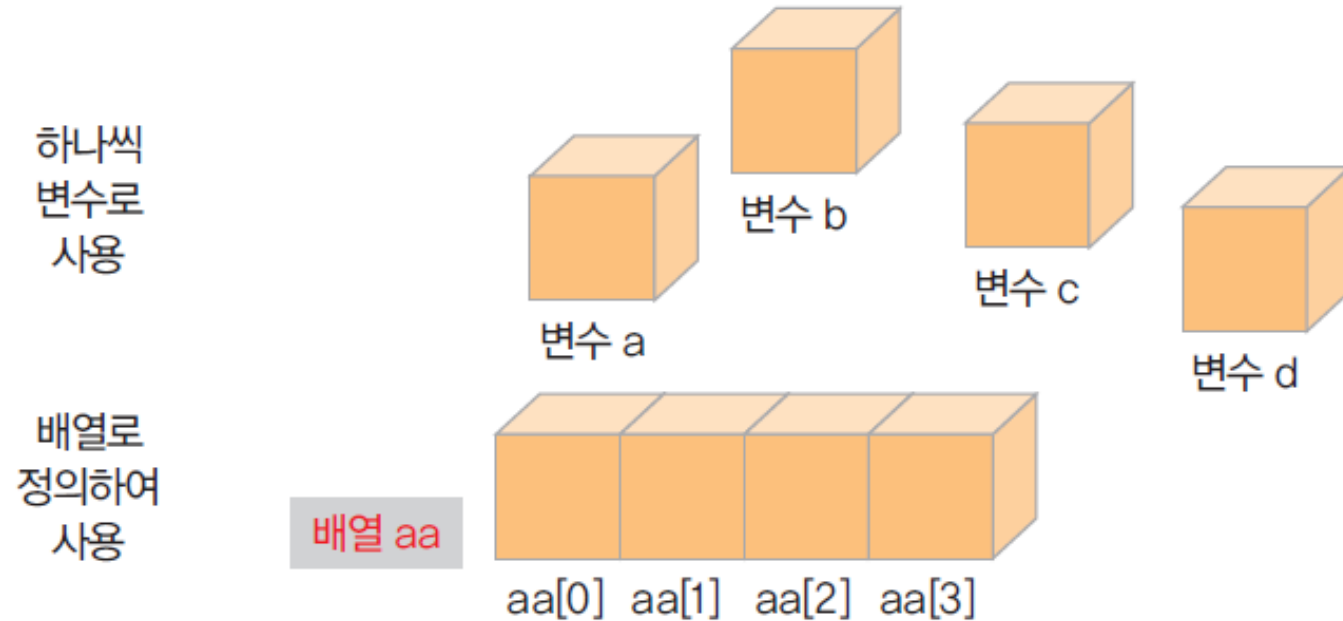


그림 8-1 배열의 개념

01 배열의 이해

■ 배열을 사용하는 이유

- 배열은 자료형이 같은 자료 여러 개를 한번에 관리하기 위해 사용함
- 예를들어 정수 20개를 입력받아 그 합계를 출력하는 프로그램을 작성한다 하자. 일일이 정수변수를 선언해야 한다. 만약에 정수를 100개로 늘리면 어떻게 될까? 이렇게 되면 비효율적이고 변수관리가 힘들다
=> 배열은 동일한 자료형의 변수를 한꺼번에 순차적으로 관리할 수 있다.

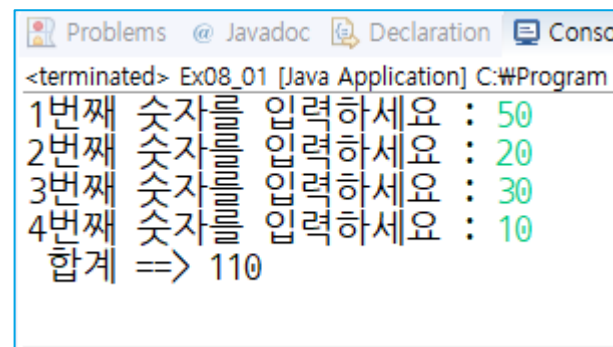
01 배열의 이해

■ 배열을 사용하는 이유

- 예. 여러 개의 변수값을 선언하여 출력

```
1 //실습8-1 여러개의 변수값을 선언하여 출력
2
3 import java.util.Scanner;
4
5 public class Ex08_01 {
6     public static void main(String[] args) {
7         Scanner s = new Scanner(System.in);
8
9         int a, b, c, d;          //각각의 입력 변수 선언
10
11         int hap;                //합계 변수 선언
12
13         System.out.printf("1번째 숫자를 입력하세요 : ");
14         a = s.nextInt();
15         System.out.printf("2번째 숫자를 입력하세요 : ");
16         b = s.nextInt();
17         System.out.printf("3번째 숫자를 입력하세요 : ");
18         c = s.nextInt();
19         System.out.printf("4번째 숫자를 입력하세요 : ");
20         d = s.nextInt();
21
22         hap = a + b + c + d;      //입력받은 숫자를 합한다
23
24         System.out.printf(" 합계 ==> %d \n", hap);
25
26         s.close();
27     }
28 }
29
```

4개의 정수변수를 각각 선언함
이때 변수 100개를 선언한다면? 알파벳으로도 부족함...
=> 배열로 해결할 수 있음



```
<terminated> Ex08_01 [Java Application] C:\Program
1번째 숫자를 입력하세요 : 50
2번째 숫자를 입력하세요 : 20
3번째 숫자를 입력하세요 : 30
4번째 숫자를 입력하세요 : 10
합계 ==> 110
```

01 배열의 이해

■ 배열 선언

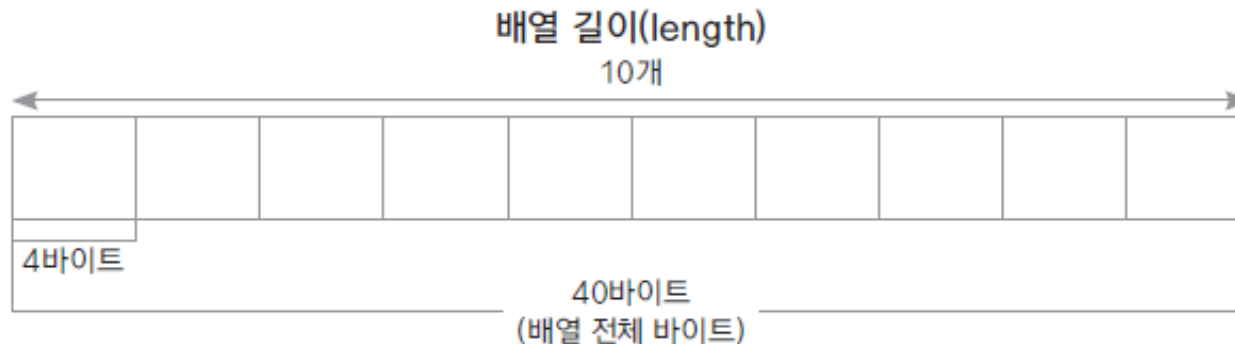
자료형[] 배열이름 = new 자료형[개수];

자료형 배열이름[] = new 자료형[개수];

자료형 [] 배열이름; (또는 자료형 배열이름[];)
배열이름 = new[개수]

- 예. `int[]arr = new int[10];`
`int arr[]= new int[10];`
`int[]arr; (또는 int arr[];)`
`arr = new int[10];`

메모리 구조



01 배열의 이해

■ 변수 선언과 배열 선언 비교

- 변수 선언 : 각각의 변수를 int a, b, c, d;와 같이 선언
- 배열 선언 : 첨자를 넣어 aa[0], aa[1], aa[2], aa[3].

이때 배열을 4개 선언한다면 첨자는 1~4가 아니라 0~3을 사용

① 변수 선언

```
int a, b, c, d;
```

a 사용

b 사용

c 사용

d 사용

② 배열 선언

```
int[ ] aa = new int[4];
```

aa[0] 사용

aa[1] 사용

aa[2] 사용

aa[3] 사용

01 배열의 이해

■ 배열의 첨자

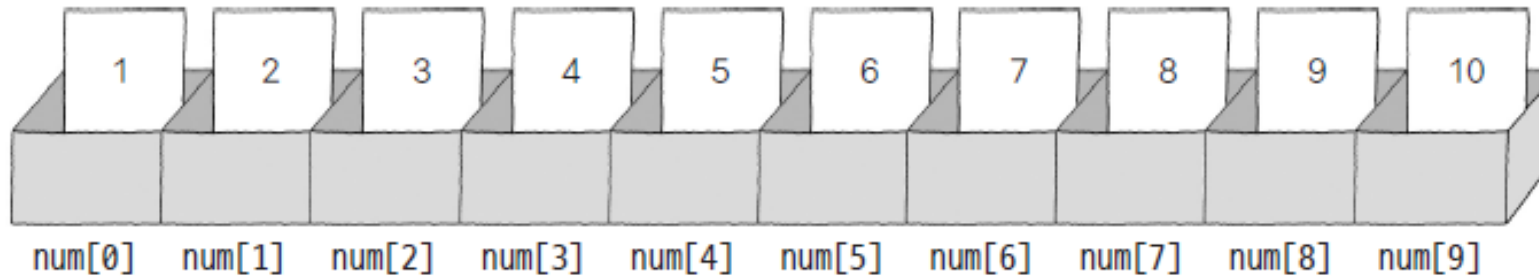
- 배열의 첨자는 0부터 시작한다는 것을 꼭 기억하기 바란다.

실무 현장의 프로그래머도 종종 하는 실수인데, `int a[3];`을 선언해놓고 `a[3] = 10;`과 같이 변수 `a[3]`을 사용하기도 한다. 이렇게 `int a[3];`으로 선언하면 `a[0]`, `a[1]`, `a[2]`, 3개의 변수를 사용할 수 있고 `a[3]`은 사용할 수 없다.

01 배열의 이해

■ 배열 사용하기

- `[]`: 인덱스 혹은 첨자 연산자
- 선언한 배열의 각 요소에 값을 넣을 때나 배열 요소에 있는 값을 가져올 때는 `[]`를 사용함
- `[i]`: `i` 번째 요소의 위치를 찾아 해당위치에 값을 넣거나 이미 저장되어 있는 값을 가져와서 사용함
- 모든 배열의 순서는 0부터 시작함
- `n`개 배열은 0부터 `n-1` 위치까지 자료가 존재함
- 예. `Int[] num = new int[]{1,2,3,4,5,6,7,8,9,10};`

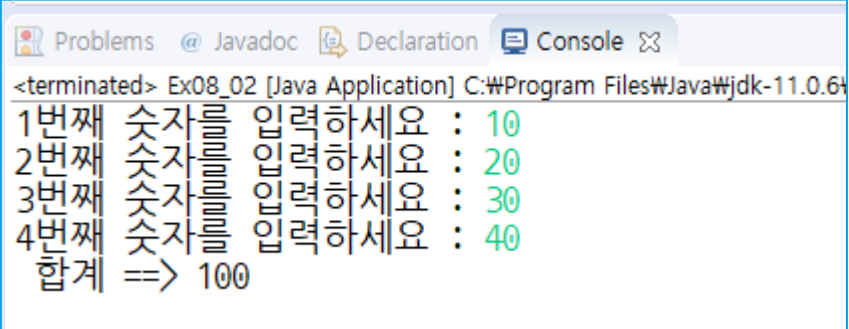


01 배열의 이해

■ 배열 사용하기

- 예. 배열에 값을 대입하여 출력 : 배열을 사용하여 [실습8-1] 수정

```
1 //실습 8-2 배열에 값을 대입하여 출력
2
3 import java.util.Scanner;
4
5 public class Ex08_02 {
6     public static void main(String[] args) {
7         Scanner s = new Scanner(System.in);
8
9         int[] aa = new int[4]; //정수형 배열 선언
10
11         int hap;
12
13         System.out.printf("1번째 숫자를 입력하세요 : ");
14         aa[0] = s.nextInt(); //aa[0]에 숫자 입력
15
16         System.out.printf("2번째 숫자를 입력하세요 : ");
17         aa[1] = s.nextInt(); //aa[1]에 숫자 입력
18
19         System.out.printf("3번째 숫자를 입력하세요 : ");
20         aa[2] = s.nextInt(); //aa[2]에 숫자 입력
21
22         System.out.printf("4번째 숫자를 입력하세요 : ");
23         aa[3] = s.nextInt(); //aa[3]에 숫자 입력
24
25         hap = aa[0] + aa[1] + aa[2] + aa[3]; //배열에 저장된 숫자를 합한다
26
27         System.out.printf(" 합계 ==> %d \n", hap);
28
29         s.close();
30     }
31 }
```



Problems Javadoc Declaration Console

<terminated> Ex08_02 [Java Application] C:\Program Files\Java\jdk-11.0.6\bin\java.exe

```
1번째 숫자를 입력하세요 : 10
2번째 숫자를 입력하세요 : 20
3번째 숫자를 입력하세요 : 30
4번째 숫자를 입력하세요 : 40
합계 ==> 100
```

01 배열의 이해

■ 배열 사용하기

- 예. 배열에 값을 대입하여 출력 : 배열을 사용하여 [실습8-1] 수정
- 만약 100개의 숫자를 더하려면 `int[] aa = new int[100]`을 선언해야 함
`aa[0] + aa[1] + aa[2] + ... + aa[100] = ?`

01 배열의 이해

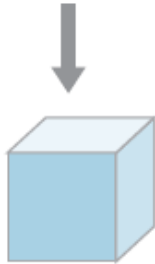
■ 배열의 활용 범위

- 배열의 첨자를 반복문(for, while, do~while)에서 활용하면 좀더 편리하게 프로그래밍 할 수 있음
- 예. for 문을 활용하여 배열 값 입력

for (4번 반복)

{

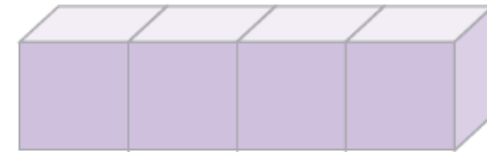
값 입력



aa[i]

}

aa[0]부터 aa[3]까지 네 번 변함



aa[0] aa[1] aa[2] aa[3]

i 값이 0부터 3까지 변함

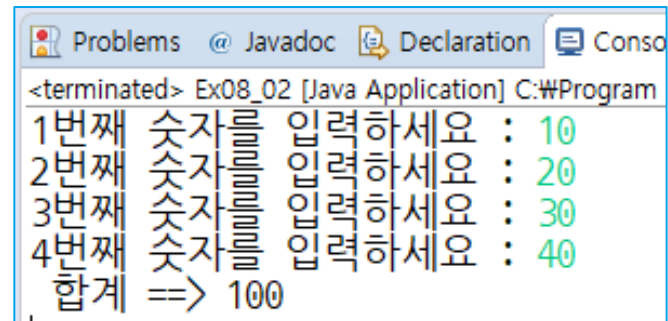
그림 8-4 for 문을 활용하여 배열 값 입력하기

01 배열의 이해

■ 배열의 활용 범위

■ 예. for문으로 배열의 첨자 활용 예

```
1 //실습8-3 for문으로 배열의 첨자 활용 예
2
3 import java.util.Scanner;
4
5 public class Ex08_03 {
6     public static void main(String[] args) {
7         Scanner s = new Scanner(System.in);
8
9         int[] aa = new int[4]; //배열 선언
10
11         int hap = 0, i; // 합계변수와 첨자를 선언
12
13         for (i = 0; i <= 3; i++) {
14             System.out.printf("%d번째 숫자를 입력하세요 : ", i + 1);
15             aa[i] = s.nextInt();
16         }
17
18         hap = aa[0] + aa[1] + aa[2] + aa[3]; //배열에 저장된 숫자를 합한다
19
20         System.out.printf(" 합계 ==> %d \n", hap);
21
22         s.close();
23     }
24 }
```



Problems Javadoc Declaration Console

<terminated> Ex08_02 [Java Application] C:\#Program

1번째	숫자를	입력하세요	:	10
2번째	숫자를	입력하세요	:	20
3번째	숫자를	입력하세요	:	30
4번째	숫자를	입력하세요	:	40
합계 ==>				100

01 배열의 이해

■ 배열의 활용 범위

```
hap = aa[0] + aa[1] + aa[2] + aa[3];    //배열에 저장된 숫자를 합한다
```

- 13행 : 4개의 변수를 더함.
- 4개의 변수를 더하는데 이것 역시 변수의 개수가 많아지면 일일이 코딩하기 어려움 : for문으로 변경할 수 있음

```
for (i=0 ; i<=3 ; i++)  
{  
    hap = hap + aa[i];  
}
```

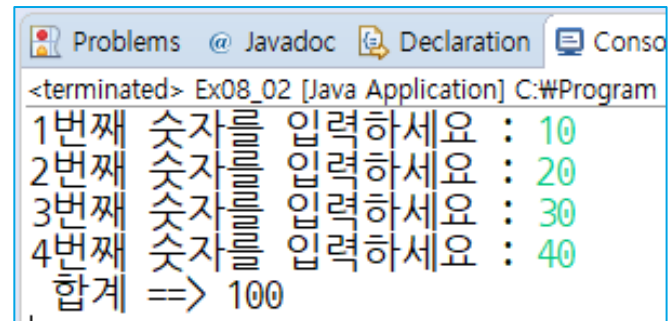
- 만약 배열이 100개라면 13행은 `hap = aa[0] + aa[1] + ... + aa[99]`로 일일이 코딩하는 번거로움이 생김. 즉 for 문으로 변경해야 함

01 배열의 이해

■ 배열의 활용 범위

- 예. 합계 구하는 식을 for문으로 변경

```
1 //실습8-3-1 합계구하는 식을 for문으로 바꾸기
2
3 import java.util.Scanner;
4
5 public class Ex08_03_0 {
6
7     public static void main(String[] args) {
8         Scanner s = new Scanner(System.in);
9
10        int[] aa = new int[4]; //배열 선언
11
12        int hap = 0, i; // 합계변수와 첨자를 선언
13
14        for (i = 0; i <= 3; i++) {
15            System.out.printf("%d번째 숫자를 입력하세요 : ", i + 1);
16            aa[i] = s.nextInt();
17        }
18
19        for (i=0 ;i<=3; i++) {
20
21            hap = hap+aa[i];
22        }
23
24        System.out.printf(" 합계 ==> %d \n", hap);
25
26        s.close();
27    }
28
29 }
```



Problems @ Javadoc Declaration Console

<terminated> Ex08_02 [Java Application] C:\#Program

1번째	숫자를	입력하세요 :	10
2번째	숫자를	입력하세요 :	20
3번째	숫자를	입력하세요 :	30
4번째	숫자를	입력하세요 :	40
합계 ==>			100

01 배열의 이해

■ 배열의 초기화

- 배열은 선언과 동시에 초기화 할 수 있음
- 배열을 초기화 할 때 다음과 같이 배열 선언과 동시에 특정값으로 초기화할 경우에는 요소의 개수만큼 생성되므로 배열의 개수를 명시하지 않음

```
int[ ] studentIDs = new int[ ] {101, 102, 103}; //개수는 생략함
```

```
int[ ] studentIDs = new int[3] {101, 102, 103}; //오류 발생
```

```
int[ ] studentIDs = {102, 102, 103}; //int형 요소가 3개인 배열 생성
```

- 아무런 초기화 값이 없이 선언만 한 경우, 정수는 0, 실수는 0.0 객체 배열은 null 로 초기화 됨

01 배열의 이해

■ 배열의 초기화

- 배열을 정의하는 동시에 값을 대입하는 것

- 예. 정수형 변수의 초기화

```
int a=100, b=200, c=300, d=400;
```

- 4개의 값을 담은 배열 aa의 초기화

```
int aa[ ] = {100, 200, 300, 400};
```

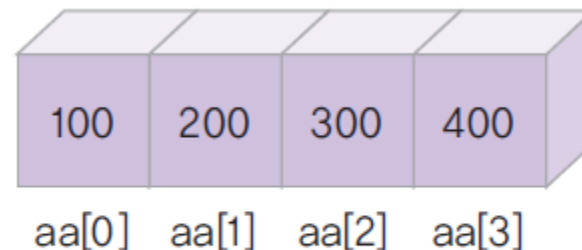


그림 8-6 배열의 초기화 1

- 다른 방식으로 먼저 선언한 뒤 초깃값을 대입

```
int aa[];  
aa = new int[] {100, 200, 300, 400};
```

- 아무런 초기화 값이 없이 선언만 한 경우, 정수는 0, 실수는 0.0 객체 배열은 null 로 초기화 됨

01 배열의 이해

■ 배열의 초기화

- 배열을 선언만 하고 초기화하지 않은 경우 : 정수는 0, 실수는 0.0 객체 배열은 null 로 초기화 됨

```
int[] aa = new int[4];
```

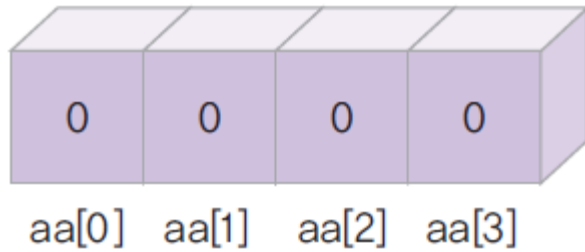


그림 8-7 배열의 초기화 2

01 배열의 이해

■ 배열의 초기화

```
1 // 실습8-4 배열의 초기화1
```

```
2  
3 public class Ex08_04 {  
4     public static void main(String[] args) {
```

```
5         int aa[] = { 100, 200, 300, 400 };      배열 선언과 동시에 직접 초깃값을 대입
```

```
6  
7         int bb[] = new int[] { 100, 200, 300 };  배열 선언과 동시에 new 연산자와 함께 초깃값을 대입
```

```
8  
9         int cc[];  
10        cc = new int[] { 100, 200 };            배열 선언 후 초깃값 대입
```

```
11  
12        int[] dd = new int[1];  
13        dd[0] = 100;                            배열 선언과 동시에 크기를 결정한 후 초깃값 대입
```

```
14  
15        int i;
```

```
16  
17        for (i = 0; i < 4; i++)  
18            System.out.printf("aa[%d]==>%d\t", i, aa[i]);  aa배열의 개수만큼 반복하고 배열 aa의 내용을 출력한다.  
19        System.out.printf("\n");
```

```
20  
21        for (i = 0; i < 3; i++)  
22            System.out.printf("bb[%d]==>%d\t", i, bb[i]);  bb배열의 개수만큼 반복하고 배열 bb의 내용을 출력한다.  
23        System.out.printf("\n");
```

```
24  
25        for (i = 0; i < 2; i++)  
26            System.out.printf("cc[%d]==>%d\t", i, cc[i]);  cc배열의 개수만큼 반복하고 배열 cc의 내용을 출력한다.  
27        System.out.printf("\n");
```

```
28  
29        for (i = 0; i < 1; i++)  
30            System.out.printf("dd[%d]==>%d\t", i, dd[i]);  dd배열의 개수만큼 반복하고 배열 dd의 내용을 출력한다.  
31        System.out.printf("\n");
```

```
32  
33    }  
34 }
```

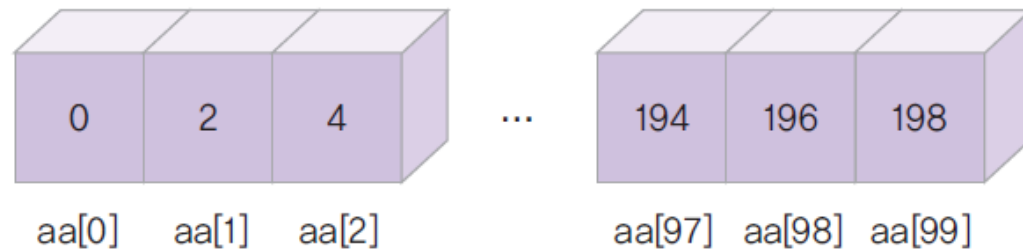
```
Problems @ Javadoc Declaration Console  
<terminated> Ex08_04 [Java Application] C:\Program Files\Java\jdk-11.0.6\bin\javaw.exe (2020. 5. 2  
aa[0]==>100    aa[1]==>200    aa[2]==>300    aa[3]==>400  
bb[0]==>100    bb[1]==>200    bb[2]==>300  
cc[0]==>100    cc[1]==>200  
dd[0]==>100
```

01 배열의 이해

■ 배열의 초기화

- 100개의 배열 aa를 0, 2, 4, 8, ...(2의 배수)로 초기화한 다음, 배열 bb에 역순으로 넣기

❶ 배열 aa를
짝수로 초기화



❷ 배열 bb에
역순으로 대입

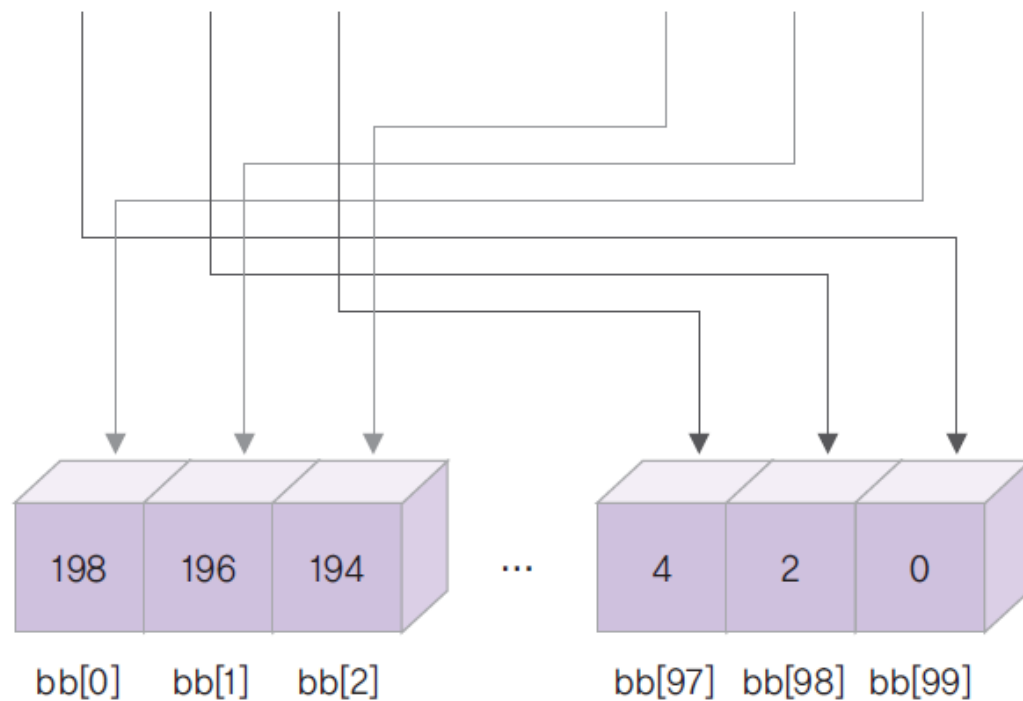


그림 8-9 배열의 초기화 3

01 배열의 이해

■ 배열의 초기화

- 100개의 배열 aa를 0, 2, 4, 8, ...(2의 배수)로 초기화한 다음, 배열 bb에 역순으로 넣기

```
1 //실습8-5 배열의 초기화2
```

```
2  
3 public class Ex08_05 {  
4     public static void main(String[] args) {
```

```
5  
6     int[] aa = new int[100];  
7     int bb[] = new int[100];
```

배열aa, bb 선언

```
8  
9     int i;
```

```
10  
11     for (i = 0; i < 100; i++) {  
12         aa[i] = i * 2;  
13     }
```

배열 aa[0], aa[1], ... , aa[99]에 0,2,4,...198을 대입

```
14  
15  
16     for (i = 0; i < 100; i++) {  
17         bb[i] = aa[99 - i];  
18     }
```

배열bb[0]에 aa[99]를 대입하는 방식으로 값을 역순으로 대입

```
19  
20     System.out.printf("bb[0]는 %d, bb[99]는 %d 입력됨\n", bb[0], bb[99]);  
21 }  
22 }  
23
```

Problems Javadoc Declaration Console

<terminated> Ex08_05 [Java Application] C:\Program Files\Ja

bb[0]는 198, bb[99]는 0 입력됨

01 배열의 이해

■ 배열 요소의 개수 알아내기

- 배열 요소의 개수

```
배열요소개수 = 배열이름.length;
```

- 예. `int aa[] = new int[4];` 의 배열의 크기 구하기

-

```
배열요소개수 = aa.length;
```

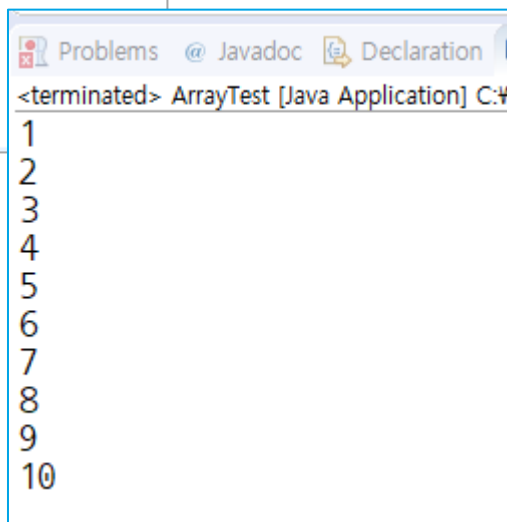

01 배열의 이해

■ 배열 요소의 개수 알아내기

■ 예1. 배열 요소 개수만큼 출력하기

```
1 //예1.배열초기화 하고 출력하기
2
3 package array;
4
5 public class ArrayTest {
6
7     public static void main(String[] args) {
8
9         int[] number = new int[] {1,2,3,4,5,6,7,8,9,10};
10
11         for(int i=0; i<number.length; i++){
12             System.out.println(number[i]);
13         }
14     }
15 }
```

배열의 첫번째 요소(num[0])부터
열번째 요소(num[9])까지 10개 요소값 출력



Problems @ Javadoc Declaration

<terminated> ArrayTest [Java Application] C:\

```
1
2
3
4
5
6
7
8
9
10
```

01 배열의 이해

■ 배열 요소의 개수 알아내기

■ [실습8-6] 배열 요소 개수와 크기 계산

```
1 //실습8-6 배열의 크기 계산
2
3 public class Ex08_06 {
4     public static void main(String[] args) {
5
6         int aa[] = { 10, 20, 30, 40, 50 }; 배열 선언, 초기화 개수에 의해 개수가 5개로 설정됨
7         int count, size; 배열 요소의 개수와 배열의 실제 크기를 저장할 변수 선언
8
9         count = aa.length; 배열 요소의 개수 계산 : 5개
10        size = count * Integer.BYTES; 배열의 전체 크기(바이트)를 계산 : 정수형은 4바이트
11
12        System.out.printf("배열 aa[]의 요소의 개수는 %d 개 입니다.\n", count);
13        System.out.printf("배열 aa[]의 요소의 전체 크기는 %d 바이트입니다.\n", size);
14    }
15 }
```

Problems | Javadoc | Declaration | Console

<terminated> Ex08_06 [Java Application] C:\Program Files\Java\jdk-11.0.6\bin\javaw.exe

배열 aa[]의 요소의 개수는 5 개 입니다.
배열 aa[]의 요소의 전체 크기는 20 바이트입니다.

01 배열의 이해

■ 전체 배열 길이와 유효한 요소 값

- 배열을 사용할 때 처음 선언한 배열 길이만큼 값을 저장해서 사용하지 않을 수 있음
- 예2. 배열길이출력 : 전체 배열 길이와 전체 배열에 유효한 값이 저장되어 있는 배열 요소 개수가 같다고 혼동하면 안됨

```
1 //예2. 배열 길이만큼 출력하기
2
3 package array;
4
5 public class ArrayTest2 {
6
7     public static void main(String[] args) {
8
9         double[] data = new double[5];
10
11         data[0] = 10.0;    첫번째 요소에 값 10.0 대입
12         data[1] = 20.0;    두번째 요소에 값 20.0 대입
13         data[2] = 30.0;    세번째 요소에 값 30.0 대입
14
15         for(int i=0; i<data.length; i++){
16             System.out.println(data[i]);
17         }
18     }
19 }
20 }
```

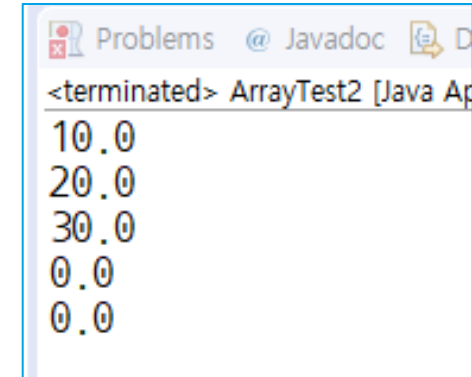
11행~13행을 보면 배열의 첫번째 요소부터 세번째 요소까지만 값을 저장

15행 : for문에서 i가 0부터 배열길이인 data.length미만까지 반복하며 배열에 저장된 요소값을 출력함

배열의 네번째 요소와 다섯번째 요소에는 값을 저장하지 않았기 때문에 0이 출력됨

double형 길이 5인 배열 선언

전체 배열 길이만큼 반복



```
<terminated> ArrayTest2 [Java Ap
10.0
20.0
30.0
0.0
0.0
```

01 배열의 이해

■ 배열의 유효한 요소 값 출력하기

■ 예3. 유효한 값이 저장된 배열 요소까지만 출력하는 프로그램 작성

```
1 //예3.배열의 유효한 요소 값 출력하기
2
3 package array;
4
5 public class ArrayTest3 {
6
7     public static void main(String[] args) {
8         double[] data = new double[5];
9
10        int size = 0;
11
12        data[0] = 10.0; size++;
13        data[1] = 20.0; size++;
14        data[2] = 30.0; size++;
15
16        for(int i=0; i<size; i++){
17            System.out.println(data[i]);
18        }
19    }
20 }
21 }
```

10행에 유효한 값이 저장된 배열 요소 개수를 저장할 **size**변수를 선언

12행~14행 : 배열요소에 순서대로 값을 저장할 때마다 **size**변수의 값을 하나씩 증가. 즉, 유효한 값을 저장하고 있는 배열 요소 개수를 알 수 있음.

16행 반복문은 전체 배열 길이만큼 반복하는게 아니라 유효한 요소 개수만큼만 반복++

int size = 0;

유효한 값이 저장된 배열 요소 개수를 저장할 변수 선언

data[0] = 10.0; size++;
data[1] = 20.0; size++;
data[2] = 30.0; size++;

값을 저장한 후 **size** 변수값 증가

for(int i=0; i<size; i++){
 System.out.println(data[i]);
}

유효한 값이 저장된 배열 요소 개수만큼 반복문 실행

Problems @ Javadoc Declaration
<terminated> ArrayTest3 [Java Application]
10.0
20.0
30.0

01 배열의 이해

■ 문자 저장 배열 만들기

- 예4. 문자 자료형 배열을 만들고 알파벳 문자를 A부터 Z까지 저장한 후 각 요소값을 알파벳 문자와 정수값(아스키코드 값)으로 출력
- 문자 자료형 배열 : `char[]`로 선언

```
1 //예4. 알파벳 문자와 아스키 코드값 출력하기
2
3 package array;
4
5 public class CharArray {
6
7     public static void main(String[] args) {
8
9         char[] alphabets = new char[26];
10        char ch = 'A';
11
12        for(int i=0 ; i<alphabets.length; i++, ch++){
13            alphabets[i] = ch; 아스키 값으로 각 요소에 저장
14        }
15
16        for(int i=0; i<alphabets.length; i++){
17            System.out.println(alphabets[i] + "," + (int)alphabets[i]);
18        }
19    }
20 }
```

9행: 대문자 알파벳 26개를 저장하기 위해 문자형 배열 선언

12행 : **for**문을 사용해 각 배열 요소에 알파벳 문자 저장

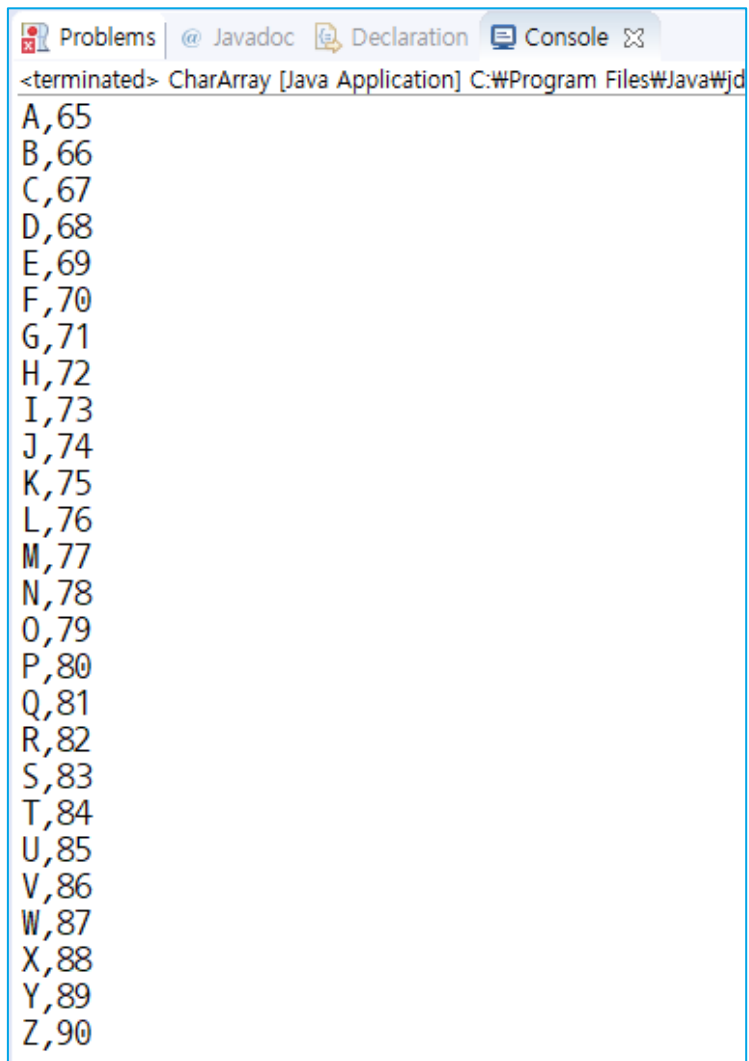
각 알파벳 문자는 실제 메모리에 아스키코드 값으로 저장되기 때문에 **ch**값에 1을 더하면(**ch++**) 1만큼 증가한 값이 배열에 저장

16행 : **for**문은 **alphabets** 배열에 저장된 알파벳 문자와 그 문자에 해당하는 아스키 코드 값을 반복하여 출력

17행 : **(int)alphabets[i]** 문장에서 자료형 변환 연산자 **(int)**는 배열에 저장된 **char**형 문자를 **int**형 정수로 변환

01 배열의 이해

■ 문자 저장 배열 만들기

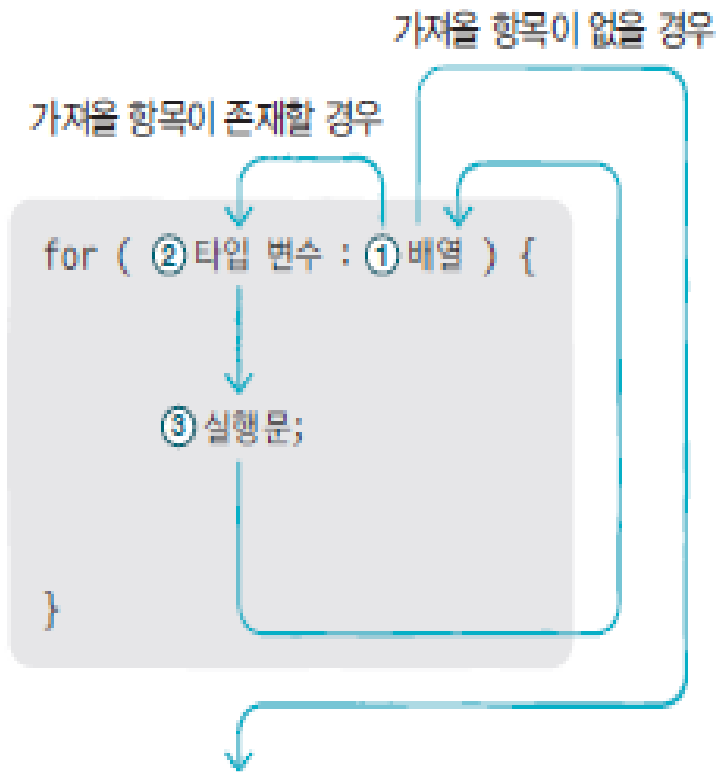


```
<terminated> CharArray [Java Application] C:\Program Files\Java\jdk...  
A,65  
B,66  
C,67  
D,68  
E,69  
F,70  
G,71  
H,72  
I,73  
J,74  
K,75  
L,76  
M,77  
N,78  
O,79  
P,80  
Q,81  
R,82  
S,83  
T,84  
U,85  
V,86  
W,87  
X,88  
Y,89  
Z,90
```

01 배열의 이해

■ 향상된 for문

- 배열이나 컬렉션을 좀 더 쉽게 처리
- 반복 실행 위해 루프 카운터 변수나 증감식 사용하지 않음
- For문의 반복횟수는 배열의 항목수가 됨



```
int[] scores = { 95, 71, 84, 93, 87 };  
  
int sum = 0;  
for (int score : scores) {  
    sum = sum + score;  
}  
System.out.println("점수 총합 = " + sum);
```

01 배열의 이해

■ 향상된 for문

- 예5. 시험점수의 합계, 평균 출력

```
1 //예5.시험점수의 합계, 평균 구하기
2
3 package sec02.exam10;
4
5 public class AdvancedForExample {
6     public static void main(String[] args) {
7         int[] scores = { 95, 71, 84, 93, 87 };
8
9         int sum = 0;
10        for (int score : scores) {
11            sum = sum + score;
12        }
13        System.out.println("점수 총합 = " + sum);
14
15        double avg = (double) sum / scores.length;
16        System.out.println("점수 평균 = " + avg);
17    }
18 }
19 }
20 }
```

Problems @ Javadoc Declaration Console

<terminated> AdvancedForExample [Java Application] C:\#Pro

점수 총합 = 430
점수 평균 = 86.0

Self study 10-1

1. [실습8-3]을 수정하여 변수 4개가 아니라 10개를 입력받도록 변경하여 실행하시오.
2. [실습8-3]을 수정하여 for 문을 while 문으로 변경하여 실행하시오.
3. [예1]을 수정하여 배열의 모든 요소 합을 계산하는 프로그램을 만드시오.
4. 키보드로 5개의 정수를 입력 받아 배열원소를 초기화하고 배열 원소의 정수 값의 합계를 구한 후 평균을 구하는 예제를 연습해보자.

```
1 package sec02;
2
3 import java.util.Scanner;
4
5 public class Array1Demo {
6     public static void main(String[] args) {
7         Scanner in = new Scanner(System.in);
8         int scores[] = new int[5];
9         int sum = 0;
10
11         for (int i = 0; i < scores.length; i++)
12             scores[i] = in.nextInt();
13
14         for (int i = 0; i < scores.length; i++)
15             sum += scores[i];
16
17         System.out.println("평균 = " + sum / 5.0);
18     }
19 }
20
```

질문은 이메일을 이용해주세요.
ds.june2@gmail.com

감사합니다