

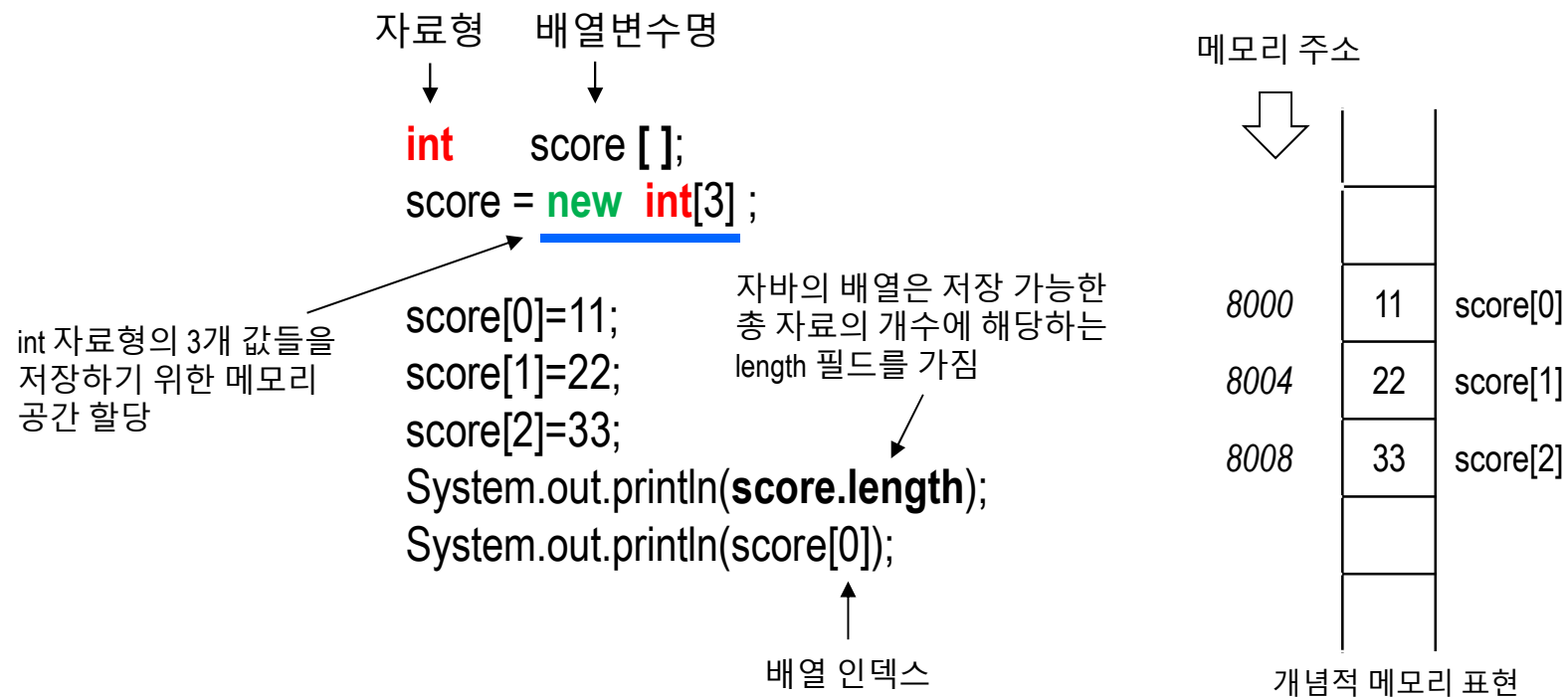
자바 기초 문법

배열

배열(array)

배열

- 인덱스(index)로 접근 가능한 동일 자료형 값들을 연속된 메모리 공간에 저장할 수 있는 자료구조



배열 생성, 초기화

```
public class Test {  
    public static void main(String[] args) {  
        int    score[ ];  
        score = new int[3];  
        score[0]=11;  
        score[1]=22;  
        score[2]=33;  
        for (int i = 0; i < score.length; i++) {  
            System.out.println(score[i]);  
        }  
    }  
}
```

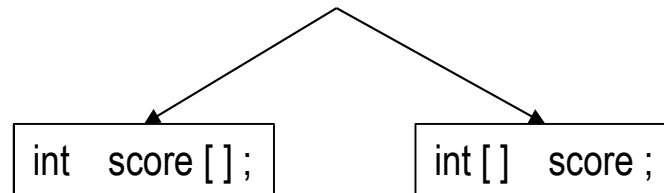
```
public class Test {  
    public static void main(String[] args) {  
        int    score[ ] = new int[3];  
        score[0]=11;  
        score[1]=22;  
        score[2]=33;  
        for (int v : score) {  
            System.out.println(v);  
        }  
    }  
}
```

for-each 문

```
public class Test {  
    public static void main(String[] args) {  
        int    score[] = {11,22,33};  
        System.out.println(Arrays.toString(score));  
    }  
}
```

배열 내 자료들을 문자열로 반환하는 메소드

배열 변수 선언 시 아래 두 방법 모두 가능



배열 인덱스 범위

다음 코드의 오류를 모두 찾으시오

```
public class Test {  
    public static void main(String[] args) {  
        int    score[] = {11,22,33};  
  
        System.out.println(score[-1]);  
        System.out.println(score[3]);  
  
        for (int i = 0; i <= score.length; i++) {  
            System.out.println(score[i]);  
        }  
    }  
}
```

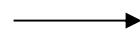
배열 생성, 초기화

```
public class Test {  
    public static void main(String[] args) {  
  
        char  genderA[]= {'여','남'};  
  
        char  genderB[]=new char[2];  
        genderB[0]='여';  
        genderB[1]='남';  
  
        char  genderC[];  
        genderC=new char[2];  
        genderC[0]='여';  
        genderC[1]='남';  
  
    }  
}
```

```
public class Test {  
    public static void main(String[] args) {  
  
        double eyeSightA[]= {1.2};  
  
        double eyeSightB[]=new double[1];  
        eyeSightB[0]=1.2;  
  
        double eyeSightC[];  
        eyeSightC=new double[1];  
        eyeSightC[0]=1.2;  
  
    }  
}
```

for-each 문, enhanced for 문

배열 score 내 인덱스 0부터 인덱스
score.length-1까지 각 원소 값에 대해
그 값을 변수 v에 대입한 후 반복문
블록을 실행한다.



```
int    score[] = {11, 22, 33};
```

```
for (int v : score) {  
    System.out.println(v);  
}
```

이차원배열 (two-dimensional array)

	0	1	2
0	11	22	33
1	44	55	66

```
public class Test {  
    public static void main(String[] args) {  
        int v[] [] = new int[2][3];  
        v[0][0]=11;  
        v[0][1]=22;  
        v[0][2]=33;  
        v[1][0]=44;  
        v[1][1]=55;  
        v[1][2]=66;  
        System.out.println(v.length); // 행 크기  
        System.out.println(v[0].length); // 0번째 행의 열 크기  
        System.out.println(v[1].length); // 1번째 행의 열 크기  
        for (int i = 0; i < v.length; i++) {  
            for (int j = 0; j < v[i].length; j++) {  
                System.out.print(v[i][j]+" ");  
            }  
            System.out.println();  
        }  
    }  
}
```

int v[][] = { {11,22,33}, {44,55,66} };

이차원배열 (two-dimensional array)

	0	1	2	3
0	11	22	33	44
1	55	66		
2	77	88	99	

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

```
        int v[ ][ ] = new int[3][ ];
```

```
        v[0]=new int[4];
```

```
        v[1]=new int[2];
```

```
        v[2]=new int[3];
```

```
        v[0][0]=11;
```

```
        v[0][1]=22;
```

```
        v[0][2]=33;
```

```
        v[0][3]=44;
```

```
        v[1][0]=55;
```

```
        v[1][1]=66;
```

```
        v[2][0]=77;
```

```
        v[2][1]=88;
```

```
        v[2][2]=99;
```

```
        System.out.println(v.length); // 행 크기
```

```
        System.out.println(v[0].length); // 0번째 행의 열 크기
```

```
        System.out.println(v[1].length); // 1번째 행의 열 크기
```

```
        System.out.println(v[2].length); // 2번째 행의 열 크기
```

```
        System.out.println(Arrays.toString(v[0]));
```

```
        System.out.println(Arrays.toString(v[1]));
```

```
        System.out.println(Arrays.toString(v[2]));
```

```
    }
```

```
}
```

```
int v[ ][ ] = { {11,22,33,44}, {55,66}, {77,88,99} };
```


다차원배열 (multi-dimensional array)

```
public class Test {  
    public static void main(String[] args) {  
        int v[][][] = new int[10][40][3];  
        System.out.println(v.length);  
        System.out.println(v[0].length);  
        System.out.println(v[0][0].length);  
        v[0][7][1]=95;  
        System.out.println(Arrays.toString(v[0][7]));  
    }  
}
```

- 10개 학급
- 학급별 최대 학생 40명
- 각 학생의 국영수 3개 과목 점수

배열 - 실습 A

- ✚ 94.5를 배열 score에 저장하는 코드를 작성하시오
 - 3가지 서로 다른 방법으로 작성해 보시오
- ✚ 'K', 'o', 'r', 'e', 'a'를 배열 country에 저장하는 코드를 작성하시오
 - 3가지 서로 다른 방법으로 작성해 보시오
- ✚ true, false를 불리언 배열 check에 저장하는 코드를 작성하시오
 - 3가지 서로 다른 방법으로 작성해 보시오
- ✚ 정수 99를 배열 grade에 저장하는 코드를 작성하시오
 - 3가지 서로 다른 방법으로 작성해 보시오
- ✚ "Korea", "UK", "UK"를 배열 grade에 저장하는 코드를 작성하시오
 - 3가지 서로 다른 방법으로 작성해 보시오

배열 - 실습 B

✚ 0부터 2020까지의 수를 변수 years에 저장한 후, years[1999]의 값을 출력하는 코드를 작성하시오

배열 - 실습 C

- ✚ 변수 x, y 에 저장된 두 정수의 합, 차, 곱, 몫(정수)을 배열 `stat`에 저장한 후 반복문을 사용하여 `stat`의 내용을 출력하는 코드를 작성하시오
 - 변수 x, y 는 프로그램 내에 그 값을 입력해 두시오
 - ◆ $x=10, y=3$ 에 대해 테스트하시오

배열 - 실습 D

- ✚ 다음 절차를 수행하는 코드를 차례로 작성하시오
 - 5명 학생의 수학 점수 75, 89, 100, 88, 91이 차례로 저장된 mathScore 배열을 생성하시오
 - 5명 학생의 영어 점수 90, 95, 99, 65, 83이 차례로 저장된 engScore 배열을 생성하시오
 - 5명 학생의 수학, 영어 점수의 평균이 저장된 avgScore 배열을 생성하시오
 - 수학 평균과 영어 평균이 저장된 avgSubject 배열을 생성하시오
 - avgScore의 내용을 출력하시오
 - avgSubject의 내용을 출력하시오

배열 - 실습 E1

다음은 어떤 코드와 그 실행 결과이다

- 이 코드는 반복문을 사용하여 department 배열 내 각 문자를 공백 한 칸씩 간격을 띄어 출력한다고 한다. 이 코드를 완성하시오.

```
public class Test {  
    public static void main(String[] args) {  
        char department[] = {'소','프','트','웨','어','학','과'};  
  
    }  
}
```

실행결과

소 프 트 웨 어 학 과

배열 - 실습 E2

다음은 어떤 코드와 그 실행 결과이다

- 이 코드는 반복문을 사용하여 department 배열 내 각 문자를 역순 출력한다고 한다. 이 코드를 완성하시오.

```
public class Test {  
    public static void main(String[] args) {  
        char department[] = {'소','프','트','웨','어','학','과'};  
  
    }  
}
```

실행결과

과학어웨트프소

배열 - 실습 E3

다음은 어떤 코드와 그 실행 결과이다

- 이 코드는 반복문을 사용하여 department 배열 내 각 문자를 출력할 때 문자들 사이에 대쉬문자를 삽입하여 출력한다고 한다. 이 코드를 완성하시오.

```
public class Test {  
    public static void main(String[] args) {  
        char department[] = {'소','프','트','웨','어','학','과'};  
  
    }  
}
```

실행결과

소-프-트-웨-어-학-과

배열 - 실습 F

다음은 배열 s에 저장된 숫자 문자열에 대응하는 숫자들의 평균을 출력하는 코드이다. 이 코드를 완성하시오.

```
public class Test {  
    public static void main(String[] args) {  
        String s[]= {"100", "60", "90", "70", "80"};  
  
    }  
}
```

명령행 인자(command line argument)

```
public class Test {  
    public static void main(String[] args) {  
        System.out.println(args[0]);  
        System.out.println(args[1]);  
    }  
}
```

아래 코드는 어떤 작업을 수행하는가?

```
public class Test {  
    public static void main(String[] args) {  
        for (int i = 0; i < args.length; i++) {  
            System.out.println(args[i]);  
        }  
    }  
}
```

실행예시

```
java Test apple pear  
apple  
pear
```

명령행 인자

- 명령행 인자(들)은 문자열의 배열로 변환되어 클래스 Test의 main 메소드 파라미터 args로 전달됨

배열 - 실습 G1

✚ 클래스 Test는 아래 실행 예시와 같이 명령행 인자로 주어진 두 숫자 문자열에 대응하는 수의 합을 출력한다고 한다. 클래스 Test를 작성하시오.

```
java Test 11 22  
33
```

명령프롬프트에서의 실행 예시

배열 - 실습 G2

- 클래스 Test는 아래 실행 예시와 같이 명령행 인자로 주어진 임의의 개수의 숫자 문자열에 대응하는 모든 수들의 합을 출력한다고 한다. 클래스 Test를 작성하시오.

명령프롬프트에서의
실행 예시

java Test 11 22
33


java Test 11 22 33
66

java Test
0

References

 <http://docs.oracle.com/javase/7/docs/api/>

 <https://docs.oracle.com/javase/tutorial/java/>

 김윤명. (2008). 뇌를 자극하는 Java 프로그래밍. 한빛미디어.

 남궁성. 자바의 정석. 도우출판.

 황기태, 김효수 (2015). 명품 Java Programming. 생능출판사.