

2 학기

JAVA Class

‘이것이 자바다 3판’

1. 복습 퀴즈

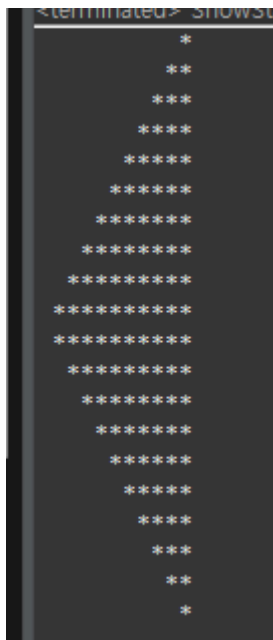
1. 1 ~ 20 까지의 랜덤 숫자를 발생시키는 숫자를 입력 받아 맞추는 프로그램을 작성
2. 입력한 숫자가 랜덤 숫자보다 작으면 입력한 숫자가 랜덤 숫자보다 크면 D 맞으면 Bingo 표시
3. 몇번만에 맞추는지 표시 하시오.

```
Input a number between 1 and 20.  
18  
Down!  
Input a number between 1 and 20.  
15  
Down!  
Input a number between 1 and 20.  
10  
Bingo!  
You passed the Quiz, 3 th try..
```

```
1 package ch04.sec05;  
2 import java.util.Scanner;  
3 public class GuessNumber {  
4  
5     public static void main(String[] args) {  
6         Scanner sc = new Scanner(System.in);  
7         int iAnswer = (int) (Math.random() * 20 + 1);  
8         int iChance = 0;  
9  
10        while (true)  
11        {  
12            System.out.println("Input a number between 1 and 20.");  
13            int input = Integer.parseInt(sc.nextLine());  
14  
15            iChance++;  
16            if (iAnswer < input)  
17            {  
18                System.out.println("Down!");  
19            }  
20            else if (iAnswer > input)  
21            {  
22                System.out.println("Up!");  
23            }  
24            else  
25            {  
26                System.out.println("Bingo!");  
27                break;  
28            }  
29        }  
30  
31        System.out.printf("You passed the Quiz, %d th try..\n", iChance);  
32    }  
33 }  
34
```

2. 복습 퀴즈

1. 아래와 같이 표시되도록 코딩 하시오

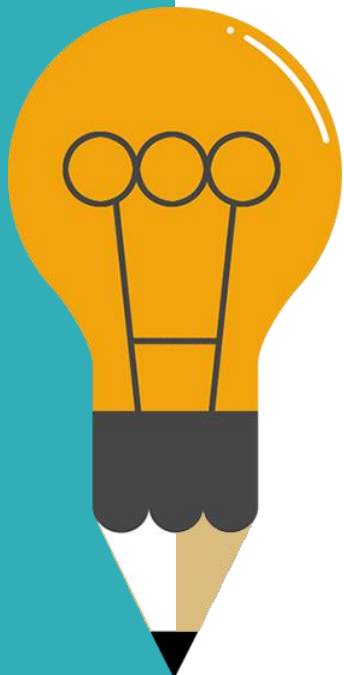


```

1 package ch04.sec04;
2
3 public class ShowStar {
4
5     public static void main(String[] args) {
6         int iMaxVal = 10;
7
8         for(int i=0; i < iMaxVal; i++)
9         {
10             for(int j=iMaxVal; j >= 0; j--)
11             {
12                 if(j <= i)
13                     System.out.printf("*");
14                 else
15                     System.out.printf(" ");
16             }
17             System.out.println("");
18         }
19         for(int i=0; i < iMaxVal; i++)
20         {
21             for(int j=0; j <= iMaxVal; j++)
22             {
23                 if(j > i)
24                     System.out.printf("*");
25                 else
26                     System.out.printf(" ");
27             }
28             System.out.println("");
29         }
30     }
31 }

```

Agenda



01

데이터 타입

데이터 타입 분류 – 기본/참조 타입

02

메모리 영역

JVM 구조

03

참조 타입의 비교 연산과 초기화

== , != , null

04

문자열 타입

비교, 추출, 길이, 대체, 잘라내기, 찾기, 분리

05

배열 타입

생성, 길이, 다차원 배열, 배열 참조

1. 데이터 타입

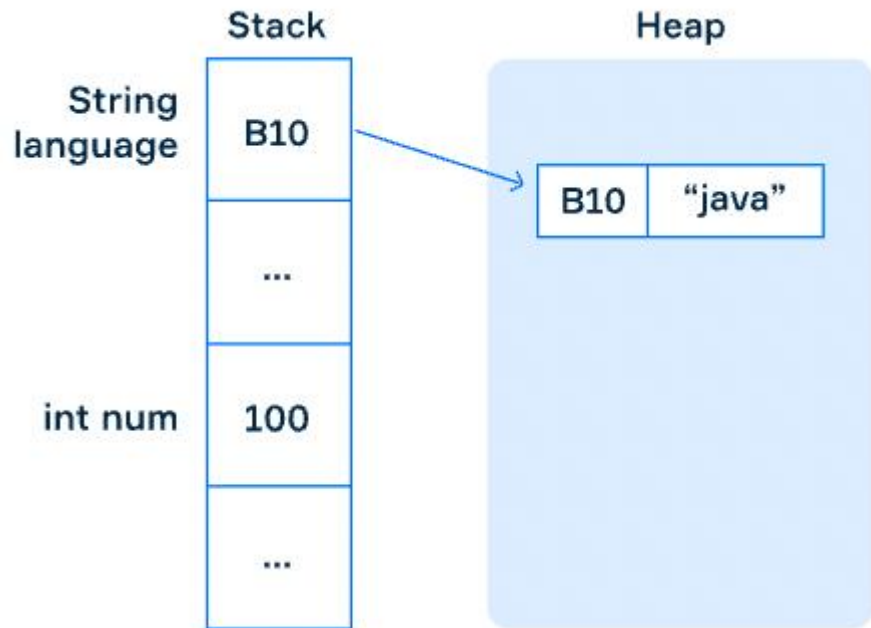
[데이터 타입 분류]

* 기본 타입 - 값 자체를 저장

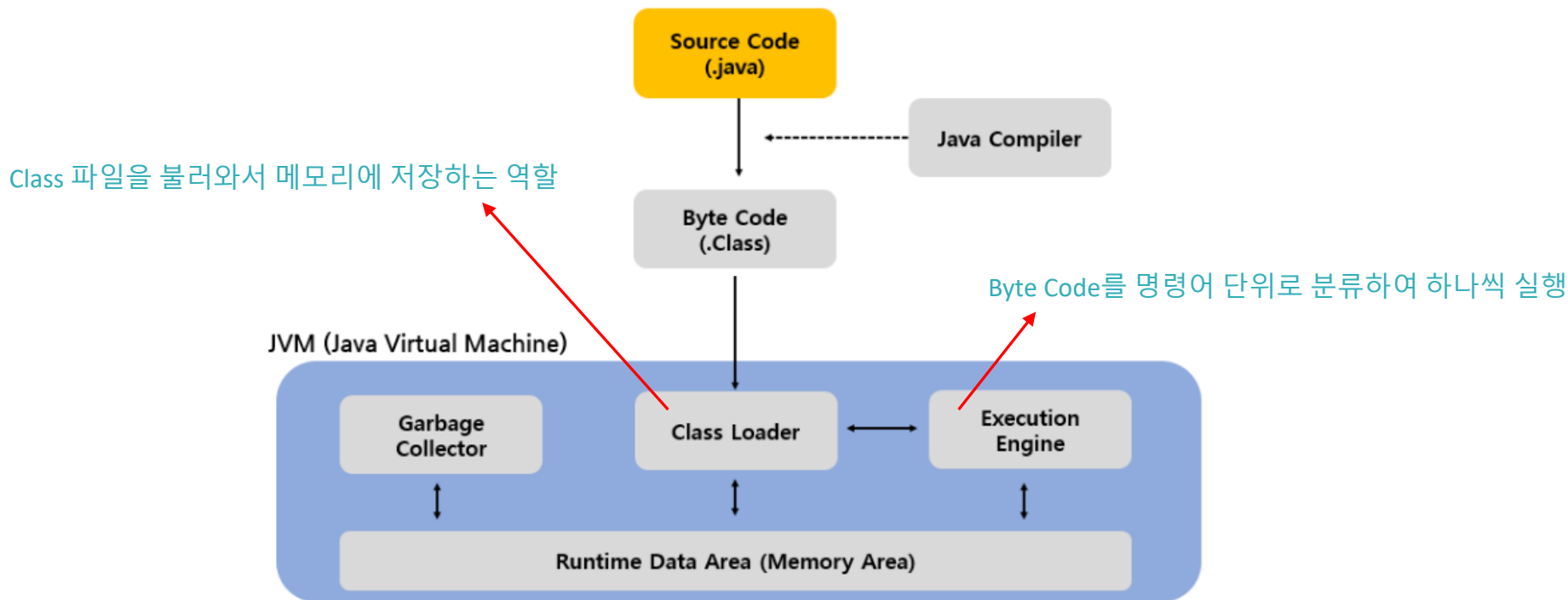
- 정수 타입 : byte, char, short, int, long
- 실수 타입 : float, double
- 논리 타입 : Boolean

* 참조 타입 - 변수 객체가 생성되는 메모리 주소 값 저장

- 배열 타입
- 열거 타입
- 클래스
- 인터페이스



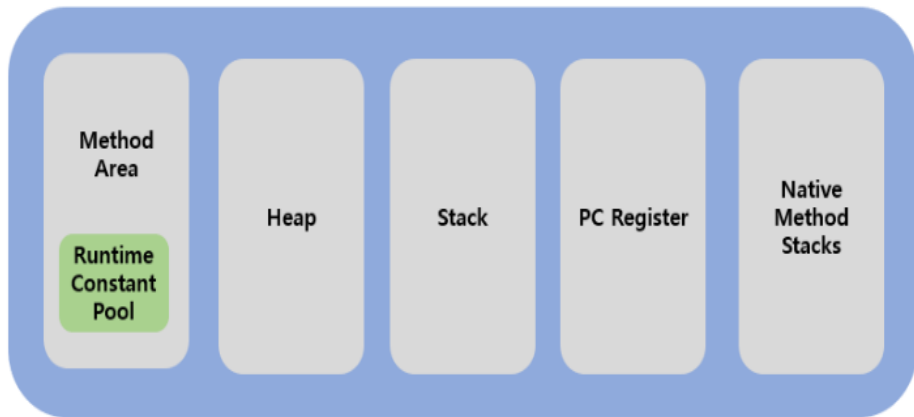
2. 메모리 영역



< JVM 의 전체적인 구조 >

2.1 메모리 영역

Runtime Data Area (Memory Area)



1. Method Area

- Class, 전역변수, Static 변수 및 상수 정보가 저장되는 공간
- 모든 스레드에서 정보가 공유

2. Heap

- 동적으로 생성된 데이터가 저장되는 공간(new, Array 등)
- 참조타입(Reference Type)의 데이터가 저장되는 공간

3. Stack

- 잠시 사용되고 필요가 없어지는 데이터가 저장되는 공간
- 참조타입(Reference Type)일 경우에는 Heap에 저장된 데이터의 주소 값을 Stack에 저장해서 사용

4. PC Register

- 스레드가 생성되면서 생기는 공간

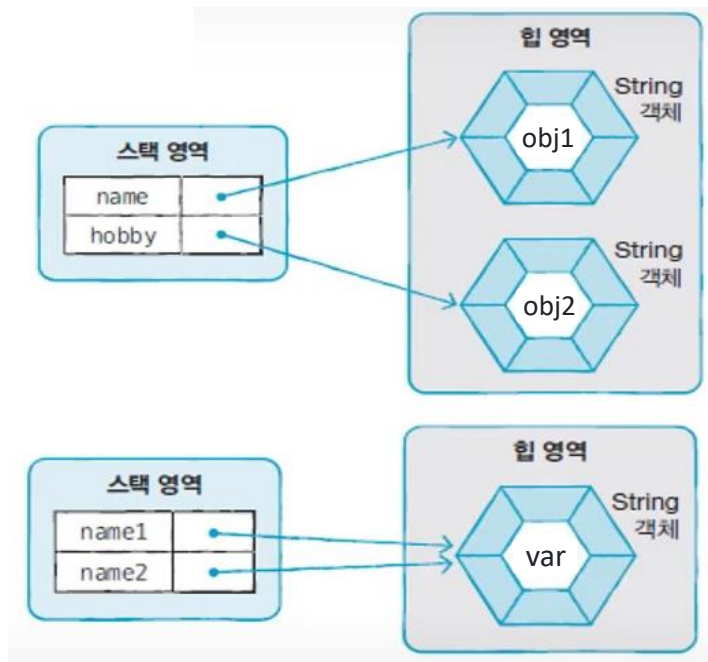
5. Native Method Stack

- Java가 아닌 다른 언어(C, C++)로 구성된 메소드를 실행이 필요할 때 사용되는 공간

2.2 메모리 영역 – Heap, Stack 비교

1. 힙 메모리는 애플리케이션의 모든 부분에서 사용되며, 스택 메모리는 하나의 스레드가 실행될 때 사용.
2. 힙 과 메서드 공간에 저장된 객체는 어디서든지 접근이 가능하지만, 스택 메모리는 다른 스레드가 접근할 수 없다.
3. 언제든지 객체가 생성되면 항상 힙 공간에 저장되며, 스택 메모리는 힙 공간에 있는 객체를 참조만 한다.
→ 스택 메모리는 primitive 타입의 지역변수와 힙 공간에 있는 객체 참조 변수만 갖고 있다.
4. 스택메모리의 생명주기는 매우 짧으며, 힙 메모리는 애플리케이션의 시작부터 끝까지 살아남는다.
5. 스택 메모리가 가득차면 자바에서는 `java.lang.StackOverFlowError`를 발생.
힙 메모리가 가득차면 `java.lang.OutOfMemoryError` : Java Heap Space 에러를 발생
6. 스택 메모리 사이즈는 힙 메모리와 비교했을 때 매우 적다.
7. 스택 메모리는 간단한 메모리 할당 방법(LIFO)를 사용하므로 힙 메모리보다 빠르다.

3. 참조 타입 의 비교 연산



참조타입 변수에서 == , != 연산자는
메모리 주소를 비교

Name == hobby
Name != hobby

name1 == name2
name1 != name2

3.1 참조 타입의 비교 연산

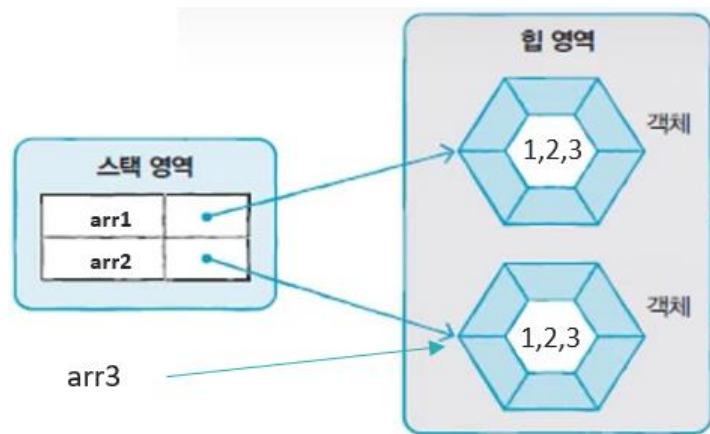
ReferenceVariableCompareExample.java

```
1 package ch05.sec03;
2
3 public class ReferenceVariableCompareExample {
4     public static void main(String[] args) {
5         int[] arr1; //배열 변수 arr1 선언
6         int[] arr2; //배열 변수 arr2 선언
7         int[] arr3; //배열 변수 arr3 선언
8
9         arr1 = new int[] { 1, 2, 3 }; //배열 { 1, 2, 3 }을 생성하고 arr1 변수에 대입
10        arr2 = new int[] { 1, 2, 3 }; //배열 { 1, 2, 3 }을 생성하고 arr2 변수에 대입
11        arr3 = arr2; //배열 변수 arr2의 값을 배열 변수 arr3에 대입
12
13        System.out.println(arr1 == arr2); //arr1과 arr2 변수 값이 동일한 변수인지 검사
14        System.out.println(arr2 == arr3); //arr2와 arr3 변수가 같은 배열을 참조하는지 검사
15    }
16 }
17
18
```

Console Problems Debug Shell

<terminated> ReferenceVariableCompareExample [Java Application] C:\Users\Lewis\p2\pool\plugins\org.eclipse.just

false
true

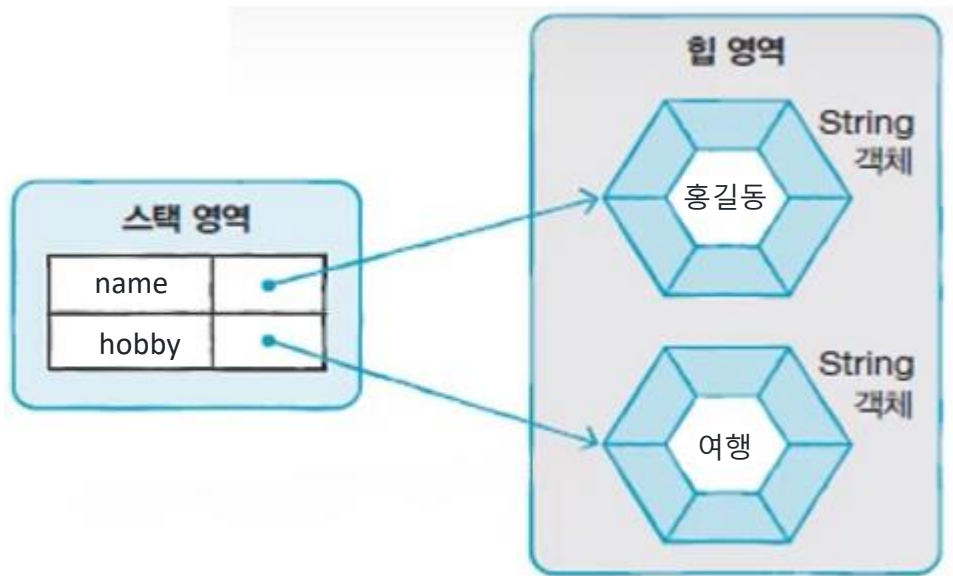


3.2 참조 타입의 초기화

```
1 package ch05.sec04;
2
3 public class NullPointerExceptionExample {
4     public static void main(String[] args) {
5         int[] intArray = null;
6         //intArray[0] = 10; //NullPointerException
7
8         String str1 = null;
9         //System.out.println("총 문자수: " + str1.length() ); //NullPointerException
10        //String str2;
11        String str3 = null;
12
13        //System.out.println("str1 == str2 : " + (str1 == str2)); // Error !!
14        System.out.println("str1 == str2 : " + (str1 == str3));
15    }
16 }
17
```

* null 은 주소값이 없다는 것을 표현하는 키워드

4. 문자열 타입



```
String name = "홍길동";  
String hobby = "여행";
```

```
String name1 = "홍길동";  
String name2 = "홍길동";  
Name1 == name2 ?? True
```

```
String name1 = new String("홍길동");  
String name2 = new String("홍길동");  
Name1 == name2 ?? False
```

4.1 문자열 타입 비교

equals() : String 객체의 내부 문자열만 비교하는 경우

```
EqualsExample.java ×
1 package ch05.sec05;
2
3 public class EqualsExample {
4     public static void main(String[] args) {
5         String strVar1 = "홍길동";
6         String strVar2 = "홍길동";
7
8         if(strVar1 == strVar2) {
9             System.out.println("strVar1과 strVar2는 참조가 같음");
10        } else {
11            System.out.println("strVar1과 strVar2는 참조가 다름");
12        }
13
14        if(strVar1.equals(strVar2)) {
15            System.out.println("strVar1과 strVar2는 문자열이 같음");
16        }
17
18        String strVar3 = new String("홍길동");
19        String strVar4 = new String("홍길동");
20
21        if(strVar3 == strVar4) {
22            System.out.println("strVar3과 strVar4는 참조가 같음");
23        } else {
24            System.out.println("strVar3과 strVar4는 참조가 다름");
25        }
26
27        if(strVar3.equals(strVar4)) {
28            System.out.println("strVar3과 strVar4는 문자열이 같음");
29        }
30    }
31 }
32
```

Console × Problems Debug Shell

<terminated> EqualsExample [Java Application] C:\Users\Lewis\p2\pool\plugins\org.eclipse

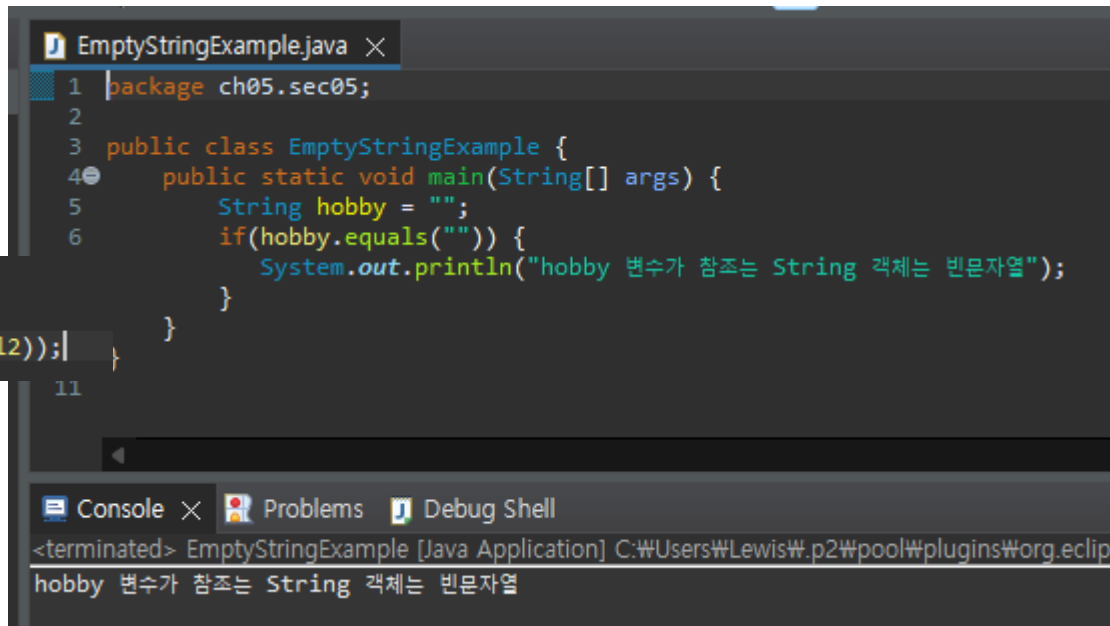
strVar1과 strVar2는 참조가 같음
strVar1과 strVar2는 문자열이 같음
strVar3과 strVar4는 참조가 다름
strVar3과 strVar4는 문자열이 같음

4.1 문자열 타입 비교

1. String 변수에 "" 빈 문자열인 경우에도 비교

2. null != ""

```
String sVal1 = null;  
String sVal2 = "";  
System.out.println("1.Check: " + (sVal1 == sVal2));
```



The screenshot shows an IDE window titled 'EmptyStringExample.java'. The code defines a class 'EmptyStringExample' with a 'main' method. In the 'main' method, a 'String hobby' variable is initialized to an empty string. An 'if' statement checks if 'hobby.equals("")' and prints a message if true. The IDE's 'Console' tab at the bottom shows the output: '<terminated> EmptyStringExample [Java Application] C:\Users\Lewis\p2\pool\plugins\org.eclipse' followed by the printed message 'hobby 변수가 참조는 String 객체는 빈문자열'.

```
EmptyStringExample.java ×  
1 package ch05.sec05;  
2  
3 public class EmptyStringExample {  
4     public static void main(String[] args) {  
5         String hobby = "";  
6         if(hobby.equals("")) {  
            System.out.println("hobby 변수가 참조는 String 객체는 빈문자열");  
        }  
    }  
11  
  
Console × Problems Debug Shell  
<terminated> EmptyStringExample [Java Application] C:\Users\Lewis\p2\pool\plugins\org.eclipse  
hobby 변수가 참조는 String 객체는 빈문자열
```

4.2 문자열 추출

charAt : 특정 위치의 문자열 추출

* 문자열 인덱스는 0 부터 시작

String substring(int start) : 현재 문자열 객체

start 위치부터 끝까지 문자열 발췌

String substrnig(int start, int end) : 현재 문자열

객체에서 start 부터 end 직전까지
문자열 발췌

int indexOf(int ch) : 현재 문자열 객체에서

ch 문자가 첫번째로 발견된 위치를
반환, 없으면 -1 반환

int indexOf(String str) : 현재 문자열 객체에서

str를 찾아서 존재하면 첫째문자
위치를 반환하고 없으면 -1 반환

CharAtExample.java X

```
1 package ch05.sec05;
2
3 public class CharAtExample {
4     public static void main(String[] args) {
5         String ssn = "0106241230123";
6         char sex = ssn.charAt(6);
7         switch (sex) {
8             case '1':
9             case '3':
10                System.out.println("남자 입니다.");
11                break;
12             case '2':
13             case '4':
14                System.out.println("여자 입니다.");
15                break;
16        }
17
18        //=====
19        String str = "CharAtExample_Test1";
20        String s1 = str.substring(12); // 시작값만 주어진다면 그 위치부터 끝까지 추출
21        System.out.printf("s1 : %s\n", s1);
22        String s2 = str.substring(12,15); // 시작값위치부터 끝값-1 위치까지 추출 (끝값위치의 문자는 포함하지않음)
23        System.out.printf("s2 : %s\n", s2);
24
25        //=====
26        String str2 = "CharAtExample_Test2";
27        int a1 = str2.indexOf('E'); // 맨 처음 'E' 값의 위치를 찾을
28        System.out.printf("a1 : %d\n", a1);
29    }
30 }
31
```

Console X Problems Debug Shell

<terminated> CharAtExample [Java Application] C:\Users\Lewis\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_22

남자 입니다.

s1 : e_Test1

s2 : e_T

a1 : 6

4.3 문자열 길이 및 대체

length() : 특정 문자열에서 문자 개수 추출

String sSubject = "Java Progamng";

int iLenght = sSubject.length(); => 14

replace() : 문자열에서 특정 문자열을 다른
문자열로 교체

String sOldStr = "Java Prog";

String sNewStr = sOldStr.replace("Java", "JAVA");

→ JAVA Prog

```
1 package ch05.sec05;
2
3 public class LengthReplaceExample {
4     public static void main(String[] args) {
5         //-----
6         String ssn = "7306241230123";
7         int length = ssn.length();
8         if(length == 13) {
9             System.out.println("주민번호 자리수가 맞습니다.");
10        } else {
11            System.out.println("주민번호 자리수가 틀립니다.");
12        }
13
14        //-----
15        String oldStr = "Java 문자열은 불변입니다. Java 문자열은 String입니다.";
16        String newStr = oldStr.replace("Java", "JAVA");
17
18        System.out.println(oldStr);
19        System.out.println(newStr);
20    }
21 }
22
```

Console Problems Debug Shell

<terminated> LengthReplaceExample [Java Application] C:\Users\#Lewis\p2\pool\plugins\org.eclipse

주민번호 자리수가 맞습니다.
Java 문자열은 불변입니다. Java 문자열은 String입니다.
JAVA 문자열은 불변입니다. JAVA 문자열은 String입니다.

4.4 문자열 분리

split : 문자열을 특정 구분자를 사용하여 따로 분리하여 문자열 추출

```
String sClass = "Java,C,Python,Go";
```

```
String[] arr = sClass.split(",");
```

```
arr[0] == "Java";
```

```
arr[1] == "C";
```

```
arr[2] == "Python";
```

```
arr[3] == "Go";
```

arr 에 있는 총 개수 : arr.length → 4개

```
1 package ch05.sec05;
2
3 public class SplitExample {
4     public static void main(String[] args) {
5         String board = "1,자바학습,참조타입 String을 학습합니다.,홍길동";
6
7         //문자열 분리
8         String[] tokens = board.split(",");
9
10        //인덱스별 읽기
11        System.out.println("번호: " + tokens[0]);
12        System.out.println("제목: " + tokens[1]);
13        System.out.println("내용: " + tokens[2]);
14        System.out.println("쓴이: " + tokens[3]);
15        System.out.println();
16
17        //for문을 이용한 읽기
18        for(int i=0; i<tokens.length; i++) {
19            System.out.println(tokens[i]);
20        }
21    }
22 }
23
```

Console × Problems Debug Shell

<terminated> SplitExample [Java Application] C:\Users\#Lewis#\p2\pool\plugins\org.eclipse

번호: 1

제목: 자바학습

내용: 참조타입 String을 학습합니다.

쓴이: 홍길동

1

자바학습

참조타입 String을 학습합니다.

홍길동

4.5.1 String

- String sTot = "10Point, 23Point, 33Point, 45Point, 55Point, 67Point";
이라는 문자열에서
아래와 같이 표현

<terminated> ScoreCheck

```
1.TotCnt : 6
2.SumValue : 233
3.Average : 38.833332
4.Average : 38.8
```

```
1 package ch05.sec05;
2
3 public class ScoreCheck {
4
5     public static void main(String[] args) {
6         String sTotStr = "10Point, 23Point, 33Point, 45Point, 55Point, 67Point";
7         String[] arEach = sTotStr.split("Point, ");
8
9         int iTotCnt = arEach.length;
10        int iSumVal = 0;
11        float fAverageVal = 0f;
12
13        for(int i=0; i<iTotCnt; i++)
14        {
15            //System.out.println(i + " : " + arEach[i].replace("Point", ""));
16            iSumVal += Integer.parseInt(arEach[i].replace("Point", ""));
17        }
18
19        fAverageVal = (float)(iSumVal) / (float)(iTotCnt);
20        System.out.println("1.TotCnt : " + iTotCnt);
21        System.out.println("2.SumValue : " + iSumVal);
22        System.out.println("3.Average : " + fAverageVal);
23        System.out.printf("4.Average : %10.2f\n", fAverageVal);
24    }
25 }
26
```

```

1 package ch05.sec05;
2
3 public class EachScoreCheck {
4
5     public static void main(String[] args) {
6         String sStudent1 = "math:85, calc:93, science:71";
7         String sStudent2 = "calc:71, math:88, science:92";
8
9         String[] arEach1 = sStudent1.split(", ");
10        String[] arEach2 = sStudent2.split(", ");
11
12        String sMathString = "math";
13        String sCalcString = "calc";
14        String sScienceString = "science";
15
16        int iMathSum = 0;
17        int iCalcSum = 0;
18        int iScienceSum = 0;
19        int iCalcCnt1 = arEach1.length;
20        int iCalcCnt2 = arEach1.length;
21
22        for(int i=0; i<iCalcCnt1; i++)
23        {
24            String[] arEach = arEach1[i].split(":");
25            if(arEach[0].indexOf(sMathString) >= 0)
26            {
27                iMathSum += Integer.parseInt(arEach[1]);
28            }
29            else if(arEach[0].indexOf(sCalcString) >= 0)
30            {
31                iCalcSum += Integer.parseInt(arEach[1]);
32            }
33            else if(arEach[0].indexOf(sScienceString) >= 0)
34            {
35                iScienceSum += Integer.parseInt(arEach[1]);
36            }
37        }
38    }

```

```

38
39        for(int i=0; i<iCalcCnt2; i++)
40        {
41            String[] arEach = arEach2[i].split(":");
42            if(arEach[0].indexOf(sMathString) >= 0)
43            {
44                iMathSum += Integer.parseInt(arEach[1]);
45            }
46            else if(arEach[0].indexOf(sCalcString) >= 0)
47            {
48                iCalcSum += Integer.parseInt(arEach[1]);
49            }
50            else if(arEach[0].indexOf(sScienceString) >= 0)
51            {
52                iScienceSum += Integer.parseInt(arEach[1]);
53            }
54        }
55
56        System.out.println("Tot Math : " + iMathSum + "   Average : "
57                            + (float)(iMathSum)/(2));
58        System.out.println("Tot Calc : " + iCalcSum + "   Average : "
59                            + (float)(iCalcSum)/(2));
60        System.out.println("Tot Science : " + iScienceSum + "   Average : "
61                            + (float)(iScienceSum)/(2));
62    }
63 }
64

```

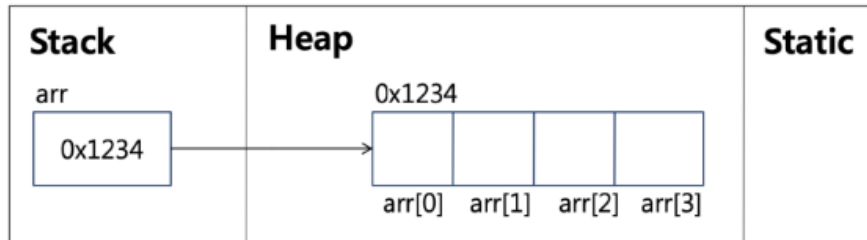
5 배열 타입

- 배열이란 동일한 데이터 타입을 가지는 데이터 구조
→ 하나의 변수로 여러 인스턴스를 표현하는 것

40	55	23	87	21	11	94	Elements
0	1	2	3	4	5	6	Index

Elements[0] == 40;
Elements[5] == 11;

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



- 배열의 장점
 1. 요소에 대한 효율적인 액세스
 2. 빠른 데이터 검색 - 연속된 메모리 위치에 저장(메모리 효율성)
 3. 구현의 용이함

5.1 배열 변수 선언

- 배열 선언 방법 : 변수타입[] 변수명;
- 변수타입 변수[];

- 배열 선언 및 초기화
 int[] iArray = null;
 String[] sArray = null;
 double[] dArray = null;

-> 배열 값이 null 인 상태에서 사용하면 NullPointerException 이 발생

5.2 값 목록으로 배열 생성

1. 변수타입[] 변수명 = { 값1, 값2, 값3..... };
2. 변수타입[] 변수명;
 변수명 = { 값1, 값2, 값3..... };
 → 컴파일 에러 : 배열 변수 선언과 동시에 사용해야함
 or 변수명 = new 타입[] { 값1, 값2, 값3..... };

```
1 package ch05.sec06;
2
3 public class ArrayCreateByVauelListExample1 {
4     public static void main(String[] args) {
5         //배열 변수 선언과 배열 생성
6         String[] season = { "Spring", "Summer", "Fall", "Winter" };
7
8         //배열의 항목값 읽기
9         System.out.println("season[0] : " + season[0]);
10        System.out.println("season[1] : " + season[1]);
11        System.out.println("season[2] : " + season[2]);
12        System.out.println("season[3] : " + season[3]);
13
14        //인덱스 1 항목의 값 변경
15        season[1] = "여름";
16        System.out.println("season[1] : " + season[1]);
17        System.out.println();
18
19        //배열 변수 선언과 배열 생성
20        int[] scores = { 83, 90, 87 };
21
22        //총합과 평균 구하기
23        int sum = 0;
24        for(int i=0; i<3; i++) {
25            sum += scores[i];
26        }
27        System.out.println("총합 : " + sum);
28        double avg = (double) sum / 3;
29        System.out.println("평균 : " + avg);
30    }
31 }
32
```

Console × Problems Debug Shell

<terminated> ArrayCreateByVauelListExample1 [Java Application] C:\Users\#Lewis#\p2\pool\#plugins\

```
season[0] : Spring
season[1] : Summer
season[2] : Fall
season[3] : Winter
season[1] : 여름

총합 : 260
평균 : 86.66666666666667
```

5.3 new 연산자로 배열 생성

1. 값의 목록은 없지만 생성될 값을 위한 객체 생성
2. 변수타입[] 변수명 = new 변수타입[길이];
3. 변수타입[] 변수명 = null;
 변수명 = new 변수타입[길이];

타입	타입	타입	타입	타입	타입	타입
기본값	기본값	<terminated> ArrayCreateByNewExample [Java Application] C:\W				
0	1	arr1[0] : 0, arr1[1] : 0, arr1[2] : 0, arr1[0] : 10, arr1[1] : 20, arr1[2] : 30,				
		arr2[0] : 0.0, arr2[1] : 0.0, arr2[2] : 0.0, arr2[0] : 0.1, arr2[1] : 0.2, arr2[2] : 0.3,				
		arr3[0] : null, arr3[1] : null, arr3[2] : null, arr3[0] : 1월, arr3[1] : 2월, arr3[2] : 3월,				

```
package ch05.sec06;
2
3 public class ArrayCreateByNewExample {
4     public static void main(String[] args) {
5         //배열 변수 선언과 배열 생성
6         int[] arr1 = new int[3];
7         //배열 항목의 초기값 출력
8         for(int i=0; i<3; i++) {
9             System.out.print("arr1[" + i + "] : " + arr1[i] + ", ");
10        }
11        System.out.println();
12        //배열 항목의 값 변경
13        arr1[0] = 10;
14        arr1[1] = 20;
15        arr1[2] = 30;
16        //배열 항목의 변경값 출력
17        for(int i=0; i<3; i++) {
18            System.out.print("arr1[" + i + "] : " + arr1[i] + ", ");
19        }
20        System.out.println("\n");
21
22        //배열 변수 선언과 배열 생성
23        double[] arr2 = new double[3];
24        //배열 항목의 초기값 출력
25        for(int i=0; i<3; i++) {
26            System.out.print("arr2[" + i + "] : " + arr2[i] + ", ");
27        }
28        System.out.println();
29        //배열 항목의 값 변경
30        arr2[0] = 0.1;
31        arr2[1] = 0.2;
32        arr2[2] = 0.3;
33        //배열 항목의 변경값 출력
34        for(int i=0; i<3; i++) {
35            System.out.print("arr2[" + i + "] : " + arr2[i] + ", ");
36        }
37        System.out.println("\n");
38
39        //배열 변수 선언과 배열 생성
40        String[] arr3 = new String[3];
41        //배열 항목의 초기값 출력
42        for(int i=0; i<3; i++) {
43            System.out.print("arr3[" + i + "] : " + arr3[i] + ", ");
44        }
45        System.out.println();
46        //배열 항목의 값 변경
47        arr3[0] = "1월";
48        arr3[1] = "2월";
49        arr3[2] = "3월";
50        //배열 항목의 변경값 출력
51        for(int i=0; i<3; i++) {
52            System.out.print("arr3[" + i + "] : " + arr3[i] + ", ");
53        }
54    }
55 }
56 }
```

5.4 배열 길이

1. 배열에 저장할 수 있는 항목 수
2. 변수타입[] 변수명 = { 값1, 값2, 값3};
변수명.length;
→ 3
3. 변수명.length = 5; → 컴파일 오류

```
1 package ch05.sec06;
2
3 public class ArrayLengthExample {
4     public static void main(String[] args) {
5         //배열 변수 선언과 배열 대입
6         int[] scores = { 84, 90, 96 };
7
8         //배열 항목의 총합 구하기
9         int sum = 0;
10        for(int i=0; i<scores.length; i++) {
11            sum += scores[i];
12        }
13        System.out.println("총합 : " + sum);
14
15        //배열 항목의 평균 구하기
16        double avg = (double) sum / scores.length;
17        System.out.println("평균 : " + avg);
18    }
19 }
20
```

Console × Problems Debug Shell

<terminated> ArrayLengthExample [Java Application] C:\Users\WLewis\p2\W

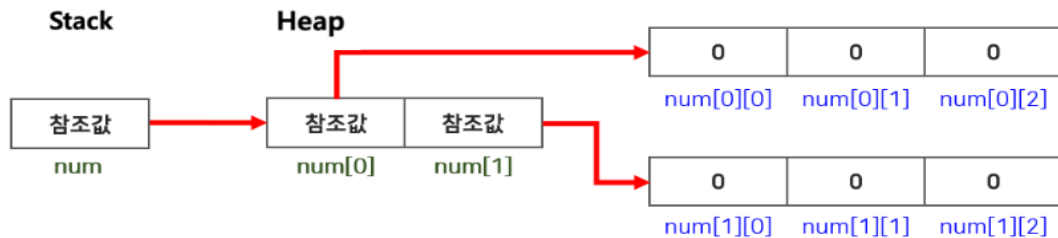
총합 : 270

평균 : 90.0

5.4 다차원 배열

* 2차원 배열

1. 자료형이 같은 1차원 배열들의 묶음
2. 할당 된 공간 마다 인덱스 번호 두개 부여 (행, 열)
3. 값 목록으로 배열 생성
타입[][] 변수 = { {}, {}..... };
int[][] iScores = { { 80, 90, 70 }
 , { 76, 86, 96 } };
iScores[0][1] ➔ 90;
iScores[1][2] ➔ 96;
4. new 연산자로 배열 생성
타입[][] 변수 = new 타입[1차원수][2차원수];
int[][] iScores = new int[2][3];



5.4.1 2차원 배열

3. 값 목록으로 배열 생성

```
타입[][] 변수 = { {}, {}..... };  
int[][] iScores = { { 80, 90, 70 }  
                    , { 76, 86, 96 } };  
iScores[0][1] → 90;  
iScores[1][2] → 96;
```

```
1 package ch05.sec07;  
2  
3 public class MultidimensionalArrayByValueListExample {  
4     public static void main(String[] args) {  
5         //2차원 배열 생성  
6         int[][] scores = {  
7             { 80, 90, 96 },  
8             { 76, 88 }  
9         };  
10  
11         //배열의 길이  
12         System.out.println("1차원 배열 길이(반의 수): " + scores.length);  
13         System.out.println("2차원 배열 길이(첫번째 반의 학생 수): " + scores[0].length);  
14         System.out.println("2차원 배열 길이(두번째 반의 학생 수): " + scores[1].length);  
15  
16         //첫번째 반의 세번째 학생의 점수 읽기  
17         System.out.println("scores[0][2]: " + scores[0][2]);  
18  
19         //두번째 반의 두번째 학생의 점수 읽기  
20         System.out.println("scores[1][1]: " + scores[1][1]);  
21  
22         //첫번째 반의 평균 점수 구하기  
23         int class1Sum = 0;  
24         for(int i=0; i<scores[0].length; i++) {  
25             class1Sum += scores[0][i];  
26         }  
27         double class1Avg = (double) class1Sum / scores[0].length;  
28         System.out.println("첫번째 반의 평균 점수: " + class1Avg);  
29  
30         //첫번째 반의 평균 점수 구하기  
31         int class2Sum = 0;  
32         for(int i=0; i<scores[1].length; i++) {  
33             class2Sum += scores[1][i];  
34         }  
35         double class2Avg = (double) class2Sum / scores[1].length;  
36         System.out.println("두번째 반의 평균 점수: " + class2Avg);  
37  
38         //전체 학생의 평균 점수 구하기  
39         int totalStudent = 0;  
40         int totalSum = 0;  
41         for(int i=0; i<scores.length; i++) {  
42             totalStudent += scores[i].length;  
43             for(int k=0; k<scores[i].length; k++) { //해당 반의 학생 수만큼 반복  
44                 totalSum += scores[i][k];  
45             }  
46         }  
47         double totalAvg = (double) totalSum / totalStudent;  
48         System.out.println("전체 학생의 평균 점수: " + totalAvg);  
49     }  
50 }  
51 }
```

```

1 package ch05.sec07;
2
3 public class MultidimensionalArrayByNewExample {
4     public static void main(String[] args) {
5         // 각 반의 학생 수가 3명으로 동일할 경우 점수 저장을 위한 2차원 배열 생성
6         int[][] mathScores = new int[2][3];
7         // 배열 항목 초기값 출력
8         for (int i = 0; i < mathScores.length; i++) { // 반의 수만큼 반복
9             for (int k = 0; k < mathScores[i].length; k++) { // 해당 반의 학생 수만큼 반복
10                 System.out.println("mathScores[" + i + "][" + k + "]: " + mathScores[i][k]);
11             }
12         }
13         System.out.println();
14         // 배열 항목 값 변경
15         mathScores[0][0] = 80;
16         mathScores[0][1] = 83;
17         mathScores[0][2] = 85;
18         mathScores[1][0] = 86;
19         mathScores[1][1] = 90;
20         mathScores[1][2] = 92;
21         // 전체 학생의 수학 평균 구하기
22         int totalStudent = 0;
23         int totalMathSum = 0;
24         for (int i = 0; i < mathScores.length; i++) { // 반의 수만큼 반복
25             totalStudent += mathScores[i].length; // 반의 학생수 합산
26             for (int k = 0; k < mathScores[i].length; k++) { // 해당 반의 학생 수만큼 반복
27                 totalMathSum += mathScores[i][k]; // 학생 점수 합산
28             }
29         }
30         double totalMathAvg = (double) totalMathSum / totalStudent;
31         System.out.println("전체 학생의 수학 평균 점수: " + totalMathAvg);
32         System.out.println();
33     }
34 }

```

```

35 // 각 반의 학생 수가 다를 경우 점수 저장을 위한 2차원 배열 생성
36 int[][] englishScores = new int[2][];
37 englishScores[0] = new int[2];
38 englishScores[1] = new int[3];
39 // 배열 항목 초기값 출력
40 for (int i = 0; i < englishScores.length; i++) { // 반의 수만큼 반복
41     for (int k = 0; k < englishScores[i].length; k++) { // 해당 반의 학생 수만큼 반복
42         System.out.println("englishScores[" + i + "][" + k + "]: " + englishScores[i][k]);
43     }
44 }
45 System.out.println();
46 // 배열 항목 값 변경
47 englishScores[0][0] = 90;
48 englishScores[0][1] = 91;
49 englishScores[1][0] = 92;
50 englishScores[1][1] = 93;
51 englishScores[1][2] = 94;
52 // 전체 학생의 영어 평균 구하기
53 totalStudent = 0;
54 int totalEnglishSum = 0;
55 for (int i = 0; i < englishScores.length; i++) { // 반의 수만큼 반복
56     totalStudent += englishScores[i].length; // 반의 학생수 합산
57     for (int k = 0; k < englishScores[i].length; k++) { // 해당 반의 학생 수만큼 반복
58         totalEnglishSum += englishScores[i][k]; // 학생 점수 합산
59     }
60 }
61 double totalEnglishAvg = (double) totalEnglishSum / totalStudent;
62 System.out.println("전체 학생의 영어 평균 점수: " + totalEnglishAvg);
63 }
64 }

```

5.5.1 배열 연습문제

길이가 10인 배열을 선언하고 임의의 수(1부터 10까지)를 생성하여 배열에 등록 하시오.
그 중 7이 몇 번째의 값인지 출력하는 소스를 구현 하시오.

```
<terminated> ArrayTest [Java Appl
```

```
5 5 5 8 5 7 9 9 7 1
7 is 6 th Number.
```

```
<terminated> ArrayTest [Java Appl
```

```
10 7 6 10 7 6 7 10 1 10
7 is 2 th Number.
7 is 5 th Number.
7 is 7 th Number.
```

```
1 package ch05.sec08;
2
3 public class ArrayTest
4 {
5     public static void main(String[] args)
6     {
7         boolean bExist7 = false;
8         int iTotCnt = 10;
9         Integer[] arrRandomVal = new Integer[iTotCnt];
10
11         for(int i=0; i<iTotCnt; i++)
12         {
13             arrRandomVal[i] = (int)(Math.random()*10) + 1;
14             System.out.printf("%d ", arrRandomVal[i]);
15         }
16         System.out.println("");
17
18         for(int i=0; i<iTotCnt; i++)
19         {
20             if(arrRandomVal[i] == 7)
21             {
22                 System.out.println("7 is " + (i+1) + " th Number.");
23
24                 bExist7 = true;
25                 break;
26             }
27         }
28
29         if(!bExist7)
30         {
31             System.out.println("There is no 7 Number.!!!!");
32         }
33     }
34 }
```

5.5.2 배열 연습문제

학생 3명이 있습니다

1번 학생의 국어, 영어, 수학 점수는 90, 80, 72

2번 학생의 국어, 영어, 수학 점수는 94, 85, 92

3번 학생의 국어, 영어, 수학 점수는 87, 91, 78

점 입니다.

학생들의 과목별 총점 및 평균, 개인별 총점 및 평균을
프로그램을 코딩 하시오.

```
1 Student   KOR : 90  MATH : 80  ENG : 72  [
2 Student   KOR : 94  MATH : 85  ENG : 92  [
3 Student   KOR : 87  MATH : 91  ENG : 78  [
   KOR SUM : 271,  Avr :  90.33
   MATH SUM : 256,  Avr :  85.33
   ENG SUM  : 242,  Avr :  80.67
```

```
1 package ch05.sec09;
2
3 public class StudentScoreArray {
4
5     public static void main(String[] args) {
6         String[] arrSubjectNm = {"KOR", "MATH", "ENG"};
7         int[][] arrStudents = {{90, 80, 72}
8                                 , {94, 85, 92}
9                                 , {87, 91, 78}
10                                };
11
12         int iTotStudentCnt = arrStudents.length;
13         int[] arrEachStudentSumVal = new int[iTotStudentCnt];
14         int iTotSubjectCnt = arrSubjectNm.length;
15         int[] arrEachSubjectSumVal = new int[iTotSubjectCnt];
16
17         for(int i = 0; i<iTotStudentCnt; i++)
18         {
19             System.out.printf("%d Student", (i+1));
20             for(int j = 0; j<iTotSubjectCnt; j++)
21             {
22                 System.out.printf("%6s : %d", arrSubjectNm[j], arrStudents[i][j]);
23                 arrEachStudentSumVal[i] += arrStudents[i][j];
24                 arrEachSubjectSumVal[j] += arrStudents[j][i];
25             }
26
27             System.out.printf(" [ SumVal : %d, Avr : %6.2f ]\n"
28                               , arrEachStudentSumVal[i]
29                               , (float)arrEachStudentSumVal[i] / iTotSubjectCnt);
30         }
31
32         for(int i=0; i<iTotSubjectCnt; i++)
33         {
34             System.out.printf("%5s SUM : %d,  Avr : %7.2f\n"
35                               , arrSubjectNm[i]
36                               , arrEachSubjectSumVal[i]
37                               , (float)(arrEachSubjectSumVal[i])/iTotSubjectCnt);
38         }
39     }
40 }
41
```

5.6 배열 복사

1. 배열은 한 번 생성하면 길이 변경 불가
2. 다른 배열을 생성하여 값을 복사
3. 얕은 복사 – Shallow Copy
=> 변경 시 원본 배열에 바로 적용
4. 깊은 복사 – Deep Copy
=> 독립된 형태로 생성됨
5. 2차원 이상의 경우 DeepCopy를 하기 위해선 for 문으로 값을 넣어야 한다.



5.6.1 얕은 복사

– Shallow Copy

```
<terminated> ShallowCop
```

```
a Array : 1 8 3
```

```
b Array : 1 8 3
```

```
-----  
a Array : 10 8 3
```

```
b Array : 10 8 3
```

```
1 package ch05.sec09;  
2  
3 public class ShallowCopy {  
4  
5     public static void main(String[] args) {  
6         int a[] = {1, 8, 3};  
7         int b[] = a;  
8  
9         System.out.print("a Array : ");  
10        for (int i = 0; i < a.length; i++)  
11            System.out.print(a[i] + " ");  
12  
13        System.out.print("\nb Array : ");  
14        for (int i = 0; i < b.length; i++)  
15            System.out.print(b[i] + " ");  
16  
17        System.out.print("\n-----\n");  
18        b[0] = 10;  
19        System.out.print("a Array : ");  
20        for (int i = 0; i < a.length; i++)  
21            System.out.print(a[i] + " ");  
22  
23        System.out.print("\nb Array : ");  
24        for (int i = 0; i < b.length; i++)  
25            System.out.print(b[i] + " ");  
26    }  
27  
28 }  
29
```

5.6.2 깊은 복사

– Deep Copy

* `b = a.clone();`

```
<terminated> DeepCopy [Java Applicatio  
a Array : 1 8 3  
b Array : 1 8 3  
-----  
a Array : 1 8 3  
b Array : 10 8 3
```

```
1 package ch05.sec09;  
2  
3 public class DeepCopy {  
4  
5     public static void main(String[] args) {  
6         int a[] = {1, 8, 3};  
7         int b[] = a.clone();  
8  
9         System.out.print("a Array : ");  
10        for (int i = 0; i < a.length; i++)  
11            System.out.print(a[i] + " ");  
12  
13        System.out.print("\nb Array : ");  
14        for (int i = 0; i < b.length; i++)  
15            System.out.print(b[i] + " ");  
16  
17        System.out.print("\n-----\n");  
18        b[0] = 10;  
19        System.out.print("a Array : ");  
20        for (int i = 0; i < a.length; i++)  
21            System.out.print(a[i] + " ");  
22  
23        System.out.print("\nb Array : ");  
24        for (int i = 0; i < b.length; i++)  
25            System.out.print(b[i] + " ");  
26    }  
27 }  
28
```


5.6.3 깊은 복사

– Deep Copy

1. 기존 배열의 일부만 덮어쓰우기
2. Clone 보다 빠르게
3. 배열의 길이 확인 필수!

System.arraycopy(A, sourcePos, B, destPos, len);

- Object A : 복사할 대상
- int sourcePos : 복사할 배열(A)의 시작 index
- Object B : 복사될 곳
- int destPos : 복사될 곳(B)의 시작 index
- int len : 복사할 길이(원소 개수)

<terminated> DeepCopy [Java Application] C:\Users+

a Array : 10 20 30 40 50 60 70 80 90 100

b Array : 0 1 2 3 4 40 50 60 70 9

a Array : 10 20 30 40 50 60 70 80 90 100

b Array : 10 1 2 3 4 40 50 60 70 9

```
1 package ch05.sec09;
2
3 public class DeepCopy {
4
5     public static void main(String[] args) {
6         int a[] = { 10, 20, 30, 40, 50, 60, 70, 80, 90, 100};
7         int b[] = { 0, 1, 2, 3, 4, 5, 6, 7, 8, 9};
8
9         int sourcePos = 3;
10        int destPos = 5;
11        int len = 4;
12
13        System.arraycopy(a, sourcePos, b, destPos, len);
14
15        System.out.print("a Array : ");
16        for (int i = 0; i < a.length; i++)
17            System.out.print(a[i] + " ");
18
19        System.out.print("\nb Array : ");
20        for (int i = 0; i < b.length; i++)
21            System.out.print(b[i] + " ");
22
23        System.out.print("\n-----\n");
24        b[0] = 10;
25        System.out.print("a Array : ");
26        for (int i = 0; i < a.length; i++)
27            System.out.print(a[i] + " ");
28
29        System.out.print("\nb Array : ");
30        for (int i = 0; i < b.length; i++)
31            System.out.print(b[i] + " ");
32    }
33 }
34
```

5.6.4 깊은 복사

– Deep Copy

1. 원하는 길이 설정

B = Arrays.copyOf(A, newLength);

• int[] A : 복사할 배열

• int newLength : 새로 만들 배열(b)의 길이

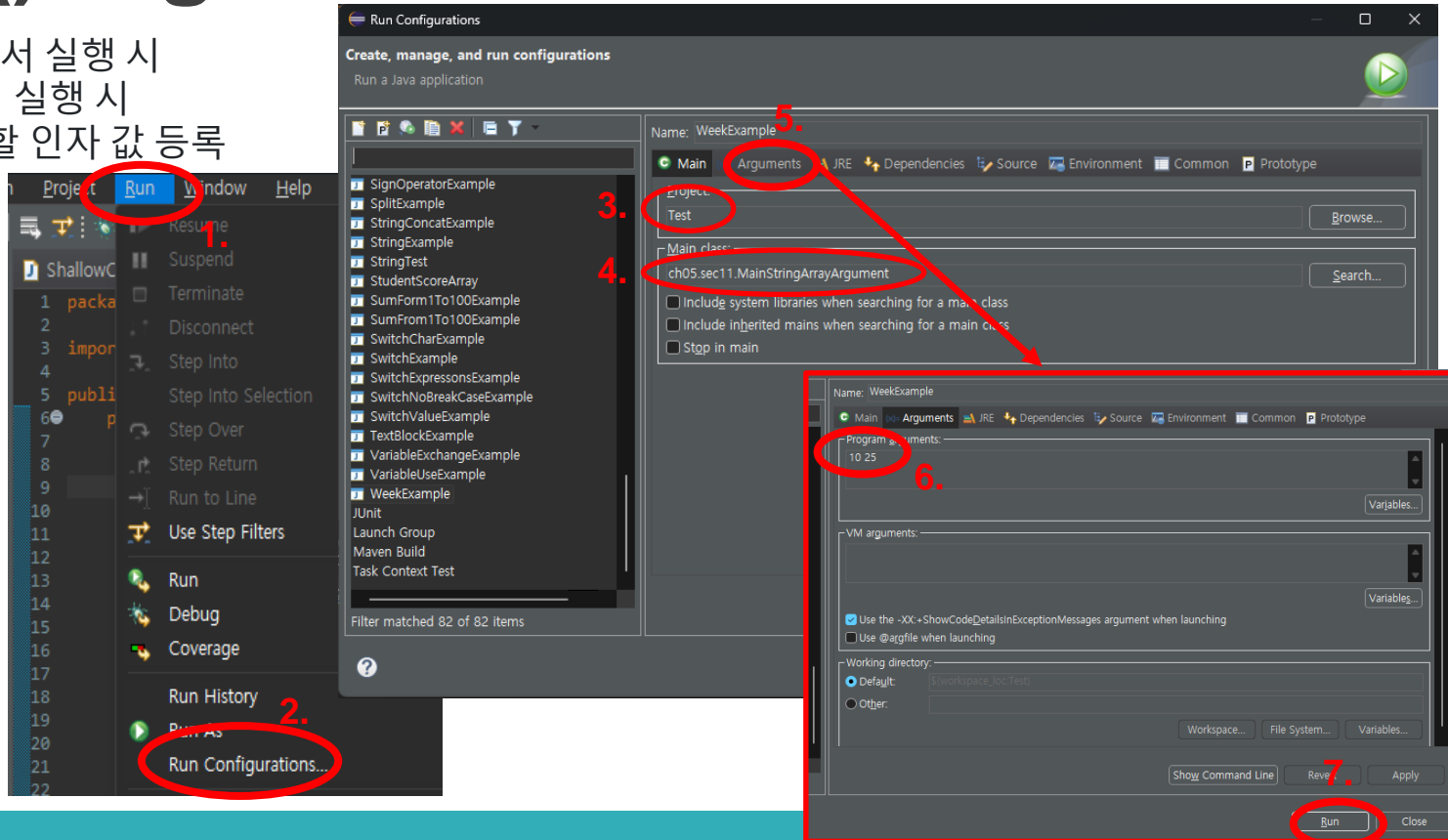
```
<terminated> DeepCopy [Java Application]
a Array : 1 2 3
b Array : 1 2 3 0 0
-----
a Array : 1 2 3
b Array : 10 2 3 0 0
```

```
1 package ch05.sec09;
2 import java.util.Arrays;
3 public class DeepCopy {
4
5     public static void main(String[] args) {
6         int[] a = new int[] {1, 2, 3};
7         int[] b = Arrays.copyOf(a, 5);
8
9         System.out.print("a Array : ");
10        for (int i = 0; i < a.length; i++)
11            System.out.print(a[i] + " ");
12
13        System.out.print("\nb Array : ");
14        for (int i = 0; i < b.length; i++)
15            System.out.print(b[i] + " ");
16
17        System.out.print("\n-----\n");
18
19        b[0] = 10;
20        System.out.print("a Array : ");
21        for (int i = 0; i < a.length; i++)
22            System.out.print(a[i] + " ");
23
24        System.out.print("\nb Array : ");
25        for (int i = 0; i < b.length; i++)
26            System.out.print(b[i] + " ");
27    }
28 }
29
```

5.7 main() args 사용

1. 윈도우 - cmd 창에서 실행 시
2. 맥OS - 터미널에서 실행 시
→ 최초 실행 시 사용할 인자 값 등록

Java Sum 10 20
-> args[0] → 10;
args[1] → 20;



5.7.1 main() args 실습

* 입력 받은 2개의 값 덧셈 프로그램

1. 입력 값 1개만 입력한 경우
2. 입력 값 2개 입력한 경우

```
1 package ch05.sec11;
2
3 public class MainStringArrayArgument {
4     public static void main(String[] args) {
5         if(args.length != 2) {
6             System.out.println("프로그램 입력 값이 부족");
7             System.exit(0);
8         }
9
10        String strNum1 = args[0];
11        String strNum2 = args[1];
12
13        int num1 = Integer.parseInt(strNum1);
14        int num2 = Integer.parseInt(strNum2);
15
16        int result = num1 + num2;
17        System.out.println(num1 + " + " + num2 + " = " + result);
18    }
19 }
```

Console × Problems Debug Shell Coverage

<terminated> WeekExample [Java Application] C:\Users\Lewis\p2\pool\plugins\org.eclipse.ju

10 + 32 = 42

5.8 열거(Enum) 타입

1. 한정된 값(요일이나 계절 등)의 고정된 데이터의 집합을 표현
2. 코드를 더 간결하고 의미 있는 형태로 작성

```
1 package ch05.sec12;
2
3 enum Season {
4     SPRING,
5     SUMMER,
6     AUTUMN,
7     WINTER
8 }
9
10 public class SeasonExe {
11     public String sAbbr;
12     public Season sFavoriteSeason;
13
14     public static void main(String[] args) {
15         SeasonExe season = new SeasonExe();
16         season.sAbbr = "Win";
17         season.sFavoriteSeason = Season.WINTER;
18
19         System.out.println("ABBR : " + season.sAbbr);
20         System.out.println("Full Name: " + season.sFavoriteSeason);
21     }
22 }
23
24
```

Console Problems Debug Shell Coverage

<terminated> SeasonExe [Java Application] C:\Users\Lewis\p2\pool\plugins

ABBR : Win

Full Name: WINTER

타입

- Week 열거형 선언

- Calendar 사용하기 위한 선언 추가

```
import java.util.Calendar;
```

- Calendar.DAY_OF_WEEK : 1 ~ 7 요일 표시
Calendar.YEAR, MONTH, DATE 등

p.197 Week.java

```
1 package ch05.sec12;
2
3 public enum Week {
4     MONDAY,
5     TUESDAY,
6     WEDNESDAY,
7     THURSDAY,
8     FRIDAY,
9     SATURDAY,
10    SUNDAY
11 }
12
```

```

1 package ch05.sec12;
2
3 import java.util.Calendar;
4
5 public class WeekExample {
6     public static void main(String[] args) {
7         //Week 열거 타입 변수 선언
8         Week today = null;
9
10        //Calendar 열거
11        Calendar cal = Calendar.getInstance();
12
13        //오늘의 요일 얻기(1~7)
14        int week = cal.get(Calendar.DAY_OF_WEEK);
15
16        //숫자를 열거 상수로 변환해서 변수에 대입
17        switch(week) {
18            case 1: today = Week.SUNDAY; break;
19            case 2: today = Week.MONDAY; break;
20            case 3: today = Week.TUESDAY; break;
21            case 4: today = Week.WEDNESDAY; break;
22            case 5: today = Week.THURSDAY; break;
23            case 6: today = Week.FRIDAY; break;
24            case 7: today = Week.SATURDAY; break;
25        }
26
27        //열거 타입 변수를 사용
28        if(today == Week.SUNDAY) {
29            System.out.println("일요일에는 축구를 합니다." + Week.SUNDAY);
30        } else {
31            System.out.println("열심히 자바 공부합니다." + Week.SUNDAY + "--" + today);
32        }
33    }
34 }

```



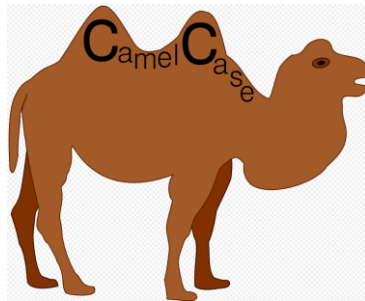
Section End

1 ~ 5 장 복습

2. 변수

2.1 변수명 생성

- 첫 글자는 문자로 중간에 \$, _ 포함 가능
- 변수명은 카멜케이스로 이루어진 영문자
- 항상 의미있는 이름을 사용



2.2 기본변수 타입

변수 구분	기본 타입	메모리 크기
정수	byte, char , short, int, long	1 byte 2 byte 4 byte 8 byte
실수	float, double	4 byte 8 byte
논리 값	boolean	1 byte
문자열	string	

2. 변수

2.2 자동 타입 변환

- 변수의 허용 범위가 작은 타입이 허용 범위가 큰 타입으로 자동 대체

Byte < short, char < int < long < float < double

2.4 문자열 변환

String -> byte : Byte.parseByte();

String -> short : Short.parseShort();

String -> int : Integer.parseInt();

String -> long : Long.parseLong();

String -> float : Float.parseFloat();

String -> double : Double.parseDouble();

String -> boolean : Boolean.parseBoolean();

↔ String sVal = String.valueOf(기본타입값);

2. 변수

2.5 변수값 출력

- `System.out.printf("나이 : %d", 25);`

<code>%b</code>	boolean 형식으로 출력
<code>%d</code>	정수 형식으로 출력
<code>%o</code>	8진수 정수의 형식으로 출력
<code>%x</code> 또는 <code>%X</code>	16진수 정수의 형식으로 출력
<code>%f</code>	소수점 형식으로 출력
<code>%c</code>	문자형식으로 출력
<code>%s</code>	문자열 형식으로 출력
<code>%n</code>	줄바꿈 기능
<code>%e</code> 또는 <code>%E</code>	지수 표현식의 형식으로 출력

2. 변수

2.6 키보드 입력

- 패키지 импорт : `import java.util.Scanner;`
- 객체 생성 : `Scanner scanner = new Scanner(System.in);`
- 데이터 입력 : `String sInputData = scanner.nextLine();`
`int iValue = scanner.nextInt();`

```
2
3 import java.util.Scanner;
4
5 public class ScannerExample {
6     public static void main(String[] args) throws Exception {
7         Scanner scanner = new Scanner(System.in);
8     }
```

2. 변수 - 연습문제

1. int iValue1 = 5;

int iValue2 = 8;

입력된 두 변수를 표시 하고 두 변수의 값을 번

```
<terminated> VarSwaping [Java Application]
1.iValue1 : 5
1.iValue2 : 8
Change Value-----
2.iValue1 : 8
2.iValue2 : 5
```

```
1 package ch02;
2
3 public class VarSwaping {
4
5     public static void main(String[] args) {
6         int iValue1 = 5;
7         int iValue2 = 8;
8
9         System.out.println("1.iValue1 : " + iValue1);
10        System.out.println("1.iValue2 : " + iValue2);
11
12        System.out.println("Change Value-----");
13        int iTemp = 0;
14        iTemp = iValue1;
15        iValue1 = iValue2;
16        iValue2 = iTemp;
17
18        System.out.println("2.iValue1 : " + iValue1);
19        System.out.println("2.iValue2 : " + iValue2);
20    }
21 }
22
```

2. 변수 - 연습문제

2. 1 ~ 10 까지 입력 받은 정수 값 3개를 각각 iValue1,에 저장하고 가장 큰 수 부터 출력 하시오.

```
<terminated> inputvaluesorting.java Applicati  
1. Insert 1 ~ 10 Value :  
5  
2. Insert 1 ~ 10 Value :  
8  
3. Insert 1 ~ 10 Value :  
3  
[First].8, [Second].5, [Third].3
```

```
1 package ch02;  
2 import java.util.Scanner;  
3 public class InputValueSorting {  
4  
5     public static void main(String[] args) {  
6         Scanner scanner = new Scanner(System.in);  
7  
8         int iSort1 = 0; // 가장 큰 수  
9         int iSort2 = 0;  
10        int iSort3 = 0;  
11        int iTempVal = 0;  
12  
13        System.out.println("1. Insert 1 ~ 10 Value :");  
14        int iValue1 = scanner.nextInt();  
15        iSort1 = iValue1;  
16  
17        System.out.println("2. Insert 1 ~ 10 Value :");  
18        int iValue2 = scanner.nextInt();  
19  
20        if(iSort1 < iValue2)  
21        {  
22            iTempVal = iSort1;  
23            iSort1 = iValue2;  
24            iSort2 = iTempVal;  
25        }  
26        else  
27        {  
28            iSort2 = iValue2;  
29        }  
30        //System.out.printf("2-[1].%d, [2].%d, [3].%d \n", iSort1, iSort2, iSort3);  
31  
32        System.out.println("3. Insert 1 ~ 10 Value :");  
33        int iValue3 = scanner.nextInt();  
34  
35        if(iSort2 < iValue3)  
36        {  
37            iTempVal = iSort2;  
38            iSort2 = iValue3;  
39  
40            iSort3 = iTempVal;  
41        }  
42        else  
43        {  
44            iSort3 = iValue3;  
45        }  
46  
47        if(iSort1 < iValue3)  
48        {  
49            iTempVal = iSort1;  
50            iSort1 = iValue3;  
51  
52            iSort2 = iTempVal;  
53        }  
54  
55        System.out.printf("[First].%d, [Second].%d, [Third].%d \n", iSort1, iSort2, iSort3);  
56    }  
57 }  
58
```

3. 연산자

3.1 연산자란 ?

- 주어진 식을 계산하여 결과를 얻어내는 과정을 연산이라고 하며, 연산을 수행하는 기호를 연산자라고 한다.

3.2 연산자 종류

1. 부호/증감 연산자 `++a, a++`
2. 산술 연산자 `0`으로 나누는 연산 주의
3. 비교 연산자
4. 논리 연산자
5. 비트 논리/이동 연산자
6. 대입 연산자
7. 삼항(조건) 연산자 조건식? 반환값1: 반환값2

3. 연산자 - 연습문제

```
1. int iValue1 = 5;  
   int iValue2 = 8;  
   int iValue3 = 3;  
   int iRsltVal = (iValue1++) + (++iValue2) % (--iValue3) / 2 + 4;  
  
   iRsltVal == ??
```

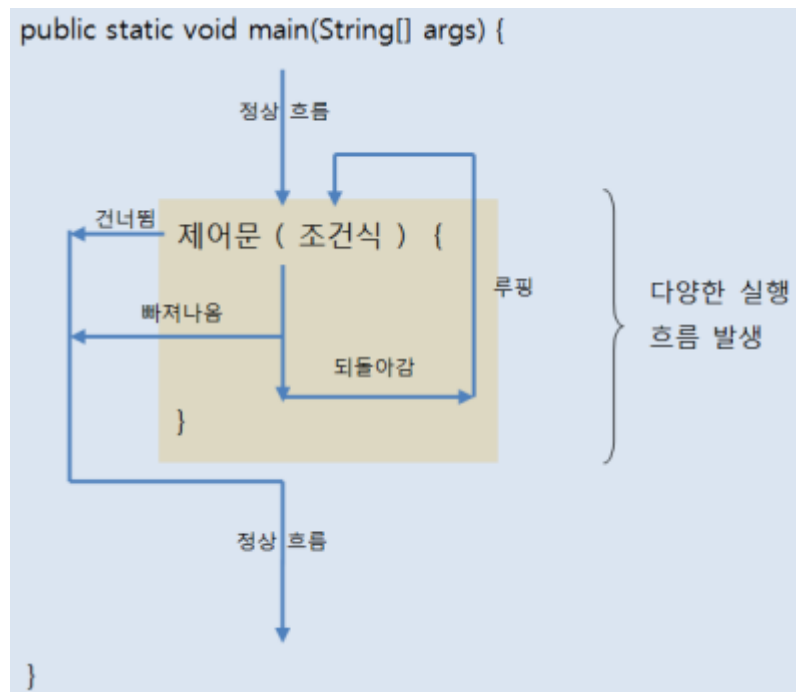
3. 연산자 - 연습문제

2. 사과 12개씩 넣을 수 있는 박스가 100 장이 있고
금일 수확한 사과가 573 개가 있다.
박스에 다 넣고 나면 빈 박스는 몇 개인가?

```
1.iLeftBoxCnt : 47  
2.(iTotApple % iEachBundle) : 9  
3.Tot left Box Counr : 52
```

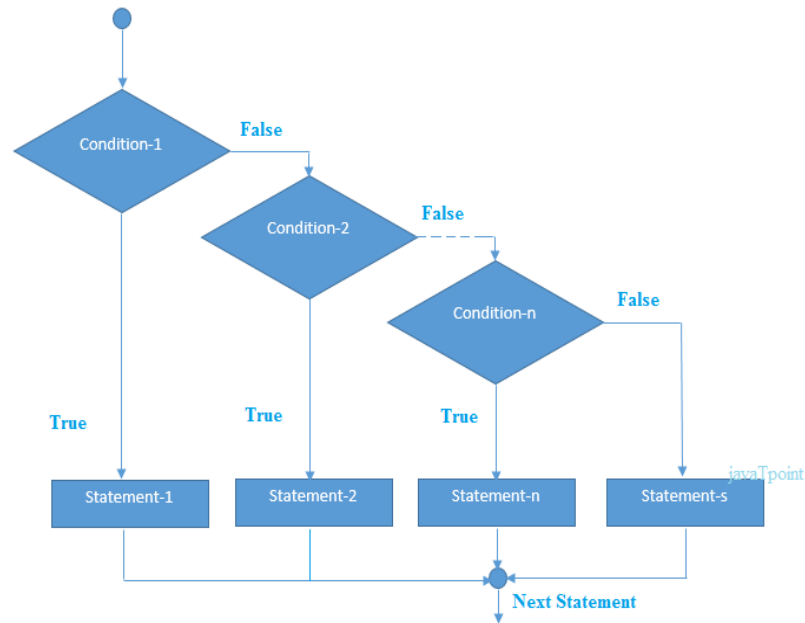
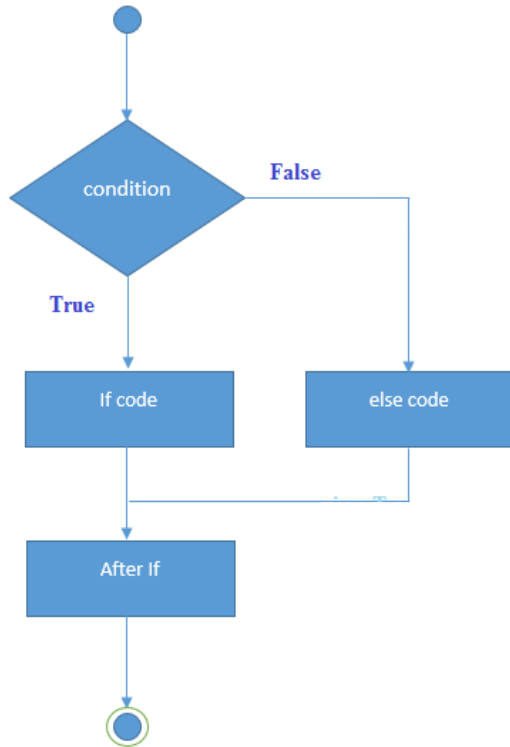
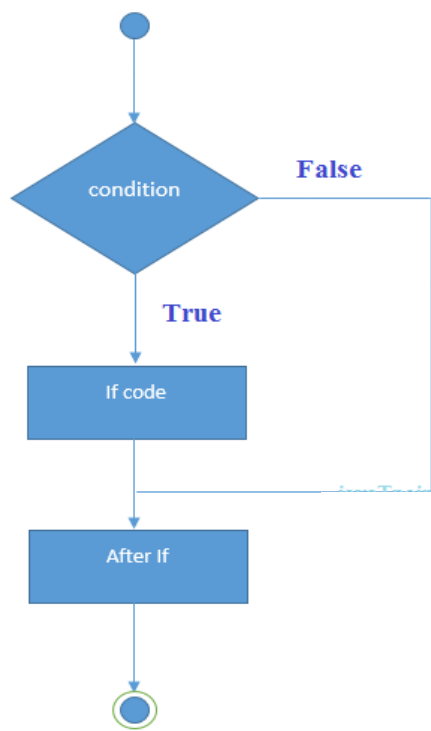
```
1 package ch05.sec12;  
2  
3 public class Cls03Test {  
4  
5     public static void main(String[] args) {  
6         int iTotBoxCnt = 100;  
7         int iEachBundle = 12;  
8  
9         int iTotApple = 573;  
10  
11        int iUsedBoxCnt = iTotApple / iEachBundle;  
12        System.out.println("1.iLeftBoxCnt : " + iUsedBoxCnt);  
13        System.out.println("2.(iTotApple % iEachBundle) : " + (iTotApple % iEachBundle));  
14        if( (iTotApple % iEachBundle) > 0)  
15            ++iUsedBoxCnt;  
16  
17        System.out.println("3.Tot left Box Counr : " + (iTotBoxCnt - iUsedBoxCnt));  
18    }  
19 }  
20 }
```


4.제어문

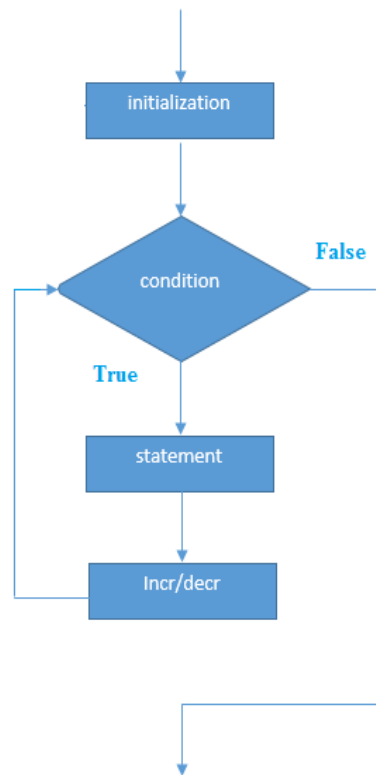
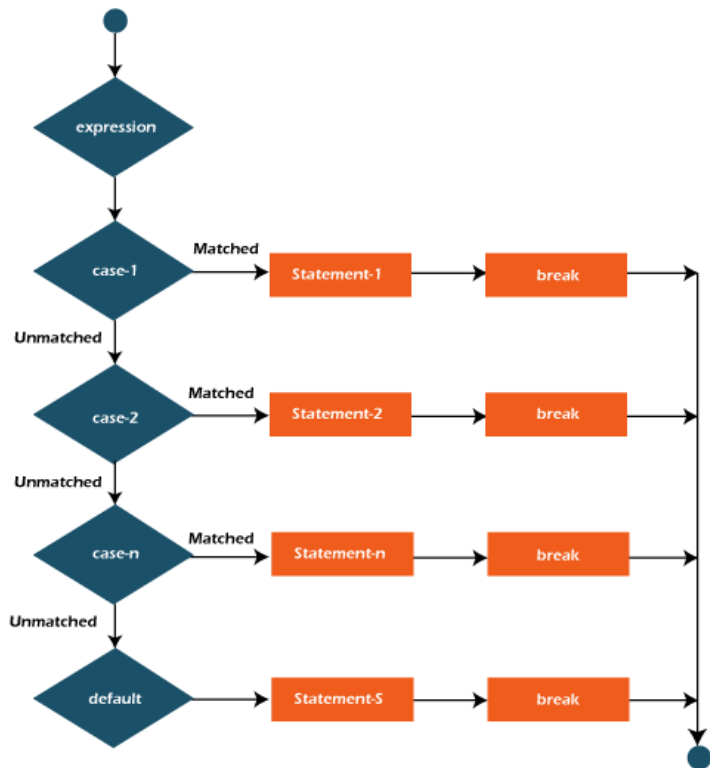


개발자가 원하는 방향으로 코드 실행 흐름 제어

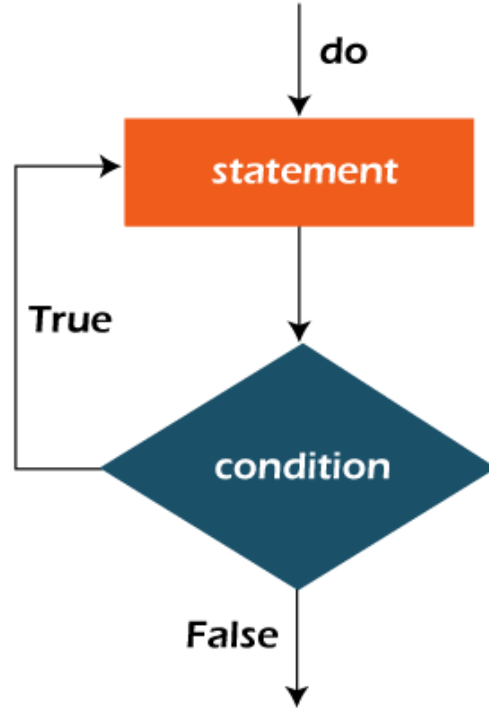
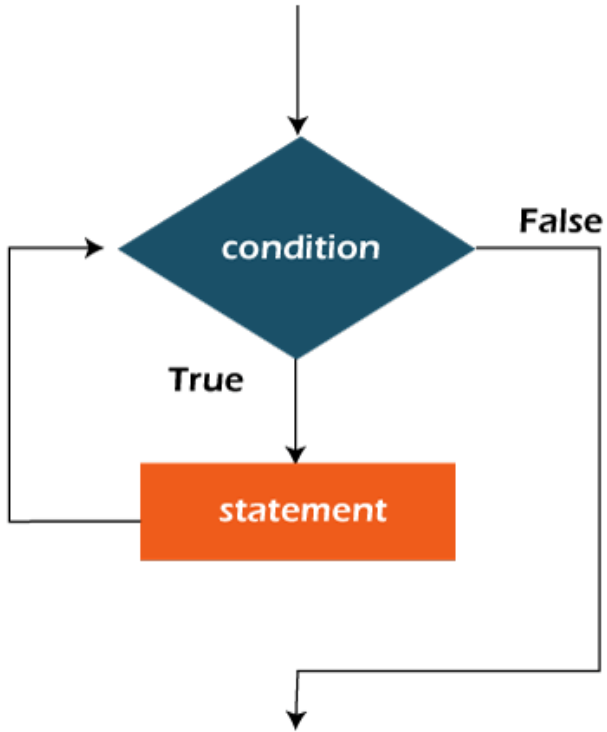
4.1 제어문 종류



4.1 제어문 종류



4.1 제어문 종류



4.2 제어문 - 연습문제

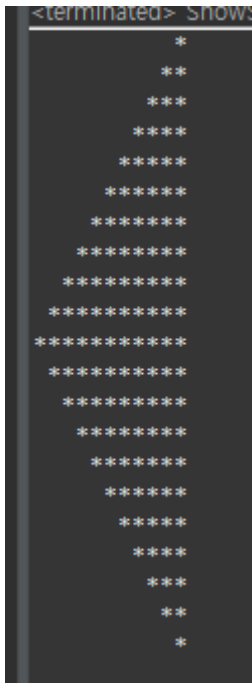
1. 9 9 단을 출력하는 프로그래밍을
단 6단을 제외하고 출력

```
4 * 7 = 28
4 * 8 = 32
4 * 9 = 36
[5 Dan]
5 * 1 = 5
5 * 2 = 10
5 * 3 = 15
5 * 4 = 20
5 * 5 = 25
5 * 6 = 30
5 * 7 = 35
5 * 8 = 40
5 * 9 = 45
[7 Dan]
7 * 1 = 7
7 * 2 = 14
7 * 3 = 21
7 * 4 = 28
7 * 5 = 35
7 * 6 = 42
7 * 7 = 49
7 * 8 = 56
7 * 9 = 63
[8 Dan]
```

```
1 package ch05.Test;
2
3 public class CalcTest {
4
5     public static void main(String[] args) {
6         for(int i=2; i<10; i++)
7         {
8             if(i != 6)
9             {
10                 System.out.printf("[%d Dan]\n", i);
11                 for(int j=1; j<10; j++)
12                 {
13                     System.out.printf("%d * %d = %d \n", i, j , i*j);
14                 }
15             }
16         }
17     }
18 }
19
```

4.2 제어문 - 연습문제

1. 아래와 같이 표시되도록 코딩 하시오
2. 라인당 최대 별의 수는 11 개, 총 21 라인



```

1 package ch04.sec04;
2
3 public class ShowStar {
4
5     public static void main(String[] args) {
6         int iMaxVal = 21;
7         int iHalfVal = iMaxVal / 2;
8         int iEachStar = 0;
9
10        for(int i=0; i < iMaxVal; i++)
11        {
12            iEachStar = i;
13
14            if(iEachStar <= iHalfVal)
15            {
16                for(int j=iHalfVal; j >= 0; j--)
17                {
18                    if(j <= iEachStar)
19                    {
20                        System.out.printf("*");
21                    }
22                    else
23                    {
24                        System.out.printf(" ");
25                    }
26                }
27            }
28            else
29            {
30                iEachStar -= iHalfVal;
31
32                for(int j=0; j <= iHalfVal; j++)
33                {
34                    if(j >= iEachStar)
35                    {
36                        System.out.printf("*");
37                    }
38                    else
39                    {
40                        System.out.printf(" ");
41                    }
42                }
43            }
44            System.out.println("");
45        }
46    }
47 }
48

```

4.2 제어문 - 연습문제

1. 아래와 같이 표시되도록 코딩 하시오
(한줄에 들어가는 총 별의 수는 25개)

```
<terminated> ShowStars.java Application
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
```

```

1 package ch05.sec12;
2
3 public class ShowStars {
4
5     public static void main(String[] args) {
6         int iTotStar = 25;
7
8         for(int i=0; i<iTotStar; i++)
9         {
10             for(int j=0; j<iTotStar; j++)
11             {
12                 if(j >= (iTotStar - i))
13                     break;
14
15                 if(j < i)
16                     System.out.printf(" ");
17                 else
18                     System.out.printf("*");
19             }
20             System.out.println("");
21         }
22     }
23 }
24

```

5.참조타입 변수

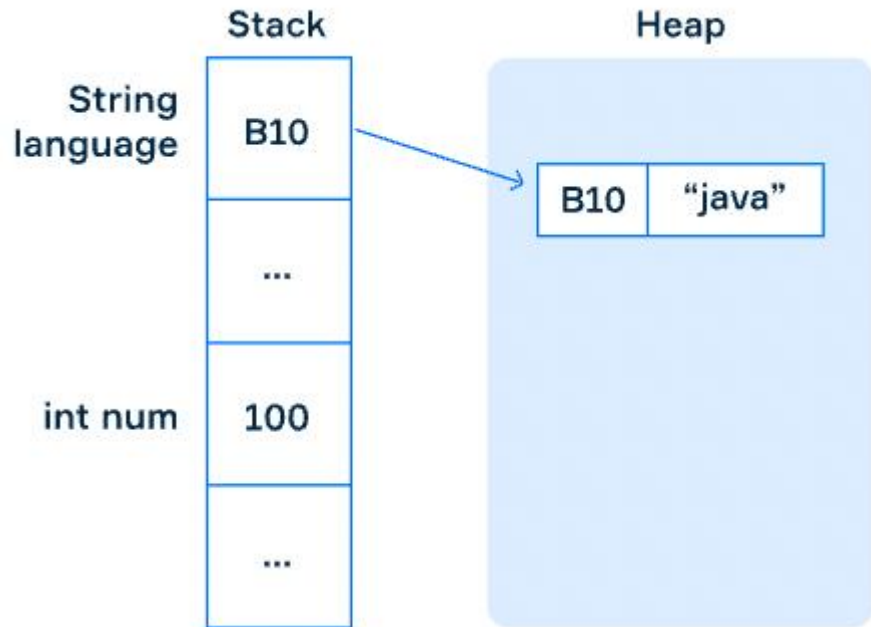
[데이터 타입 분류]

* 기본 타입 - 값 자체를 저장

- 정수 타입 : byte, char, short, int, long
- 실수 타입 : float, double
- 논리 타입 : Boolean

* 참조 타입 - 변수 객체가 생성되는 메모리 주소 값 저장

- String
- 배열 타입
- 열거 타입
- 클래스
- 인터페이스



5.1 String 변수

* String 관련 함수

1. **equals()** : 객체의 주소와 상관없이 해당 변수의 내부 문자열만 비교
2. **charAt(int n)** : 특정 위치의 문자열 추출
* 문자열 인덱스는 0 부터 시작
3. **String substring(int start)** : 현재 문자열 객체 start위치부터 끝까지 문자열 발취
4. **String substrnig(int start, int end)** : 현재 문자열 객체에서 start 부터 end 직전까지 문자열 발취
5. **int indexOf(int ch)** : 현재 문자열 객체에서 ch문자가 첫번째로 발견된 위치를 반환, 없으면 -1반환
6. **int indexOf(String str)** : 현재 문자열 객체에서 str를 찾아서 존재하면 첫째문자 위치를 반환, 없으면 -1반환
7. **length()** : 특정 문자열에서 문자 개수 추출
8. **replace()** : 문자열에서 특정 문자열을 다른 문자열로 교체
9. **split(String str)** : 문자열을 특정 구분자를 사용하여 따로 분리하여 문자열 추출

5.2 배열 변수

- 배열이란 동일한 데이터 타입을 가지는 데이터 구조
→ 하나의 변수로 여러 인스턴스를 표현하는 것

```
변수타입[] 변수명 = { 값1, 값2, 값3..... };  
int[] iScores = {1, 2};
```

```
변수타입[] 변수명 = new 변수타입[길이];  
int[] iScores = new int[2];
```

```
변수명.length;
```

```
타입[][] 변수 = new 타입[1차원수][2차원수];  
int[][] iScores = new int[2][3];  
iScores.length == 2  
iScores[0].length == 3
```

5.3 참조타입 - 연습문제

1. 10개의 임의의 정수 (1~10) 를 중복되지 않게 배

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

```
1 package ch05.Test;
2
3 public class RandomNumber {
4
5     public static void main(String[] args) {
6         int iTotCnt = 10;
7         int[] arRandomNo = new int[iTotCnt];
8         int iCreateCnt = 0;
9         int iRandomNo = 0;
10        boolean bSameCheck = false;
11
12        while(true)
13        {
14            bSameCheck = false;
15            iRandomNo = (int)(Math.random() * iTotCnt)+1;
16
17            for(int i=0; i<iTotCnt; i++)
18            {
19                if(arRandomNo[i] == iRandomNo)
20                {
21                    bSameCheck = true;
22                    break;
23                }
24            }
25
26            if(bSameCheck)
27                continue;
28
29            arRandomNo[iCreateCnt++] = iRandomNo;
30
31            System.out.printf("%d ", iRandomNo);
32
33            if(iCreateCnt >= iTotCnt)
34                break;
35        }
36    }
37 }
38
```

5.3

1.

```
1 package ch05.Test;
2 public class RandomNumber {
3     public static void main(String[] args) {
4         int iTotCnt = 10;
5         int[] arRandomNo = new int[iTotCnt];
6         int iCreateCnt = 0;
7         int iRandomNo = 0;
8         boolean bSameCheck = false;
9
10        int iTempVal = 0;
11
12        while(true)
13        {
14            bSameCheck = false;
15            iRandomNo = (int)(Math.random() * iTotCnt)+1;
16
17            for(int i=0; i<iTotCnt; i++)
18            {
19                if(arRandomNo[i] == iRandomNo)
20                {
21                    bSameCheck = true;
22                    break;
23                }
24            }
25
26            if(bSameCheck)
27                continue;
28
29            arRandomNo[iCreateCnt++] = iRandomNo;
30            System.out.printf("%d ", iRandomNo);
31
32            if(iCreateCnt >= iTotCnt)
33                break;
34        }
35    }
36}
```

에 등록하고 오름차순 소팅

```
34 }
35 System.out.println("");
36 for(int i=0; i<iTotCnt; i++)
37 {
38     for(int j=i+1; j<iTotCnt; j++)
39     {
40         if(arRandomNo[i] < arRandomNo[j])
41         {
42             iTempVal = arRandomNo[i];
43             arRandomNo[i] = arRandomNo[j];
44             arRandomNo[j] = iTempVal;
45         }
46     }
47 }
48
49 for(int i=0; i<iTotCnt; i++)
50 {
51     System.out.printf("%d ", arRandomNo[i]);
52 }
53 }
54 }
```