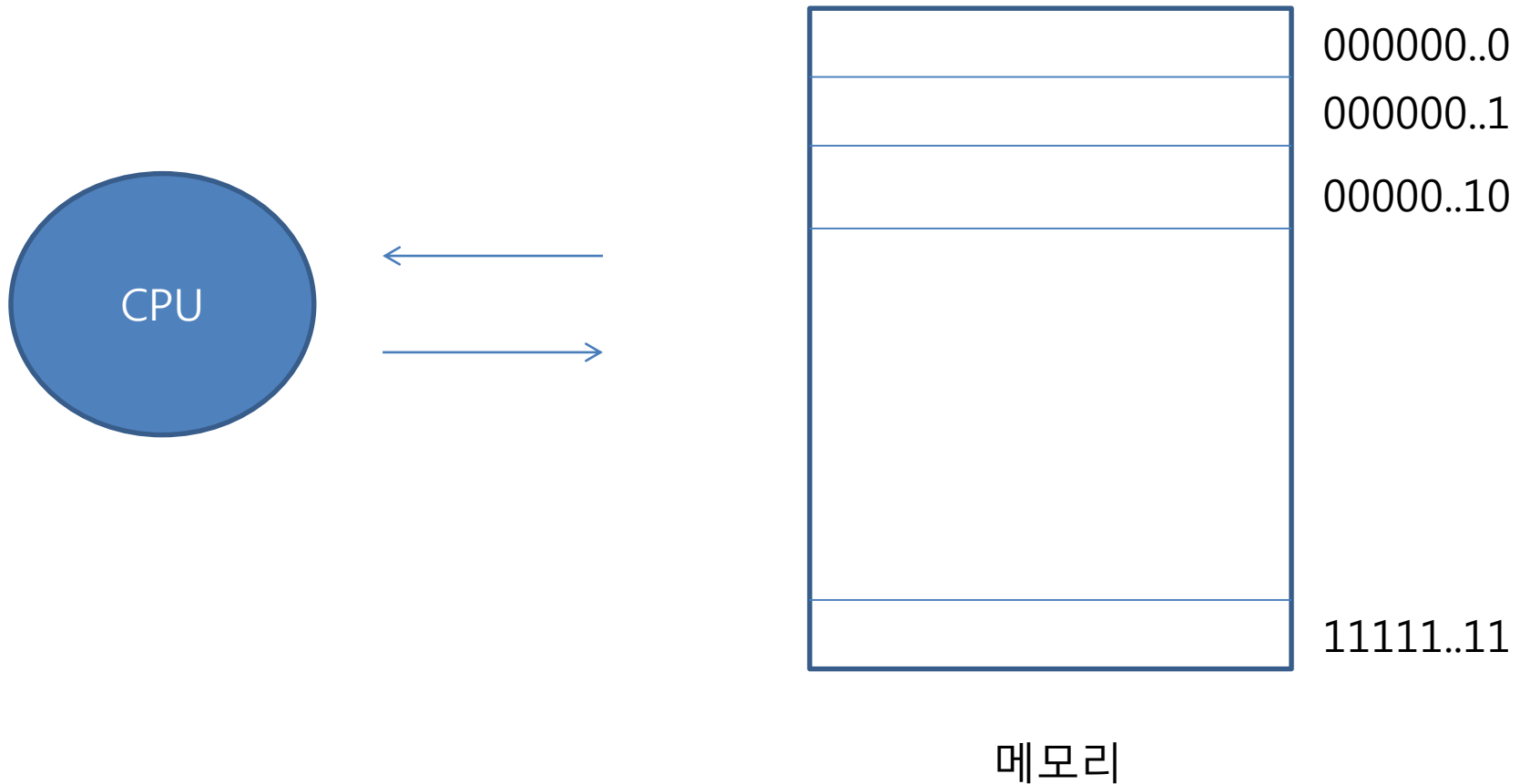
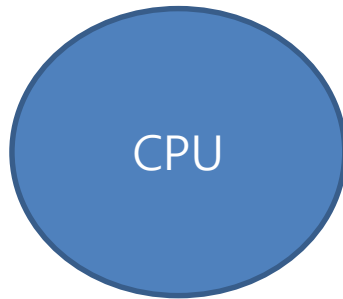


컴퓨터 내부



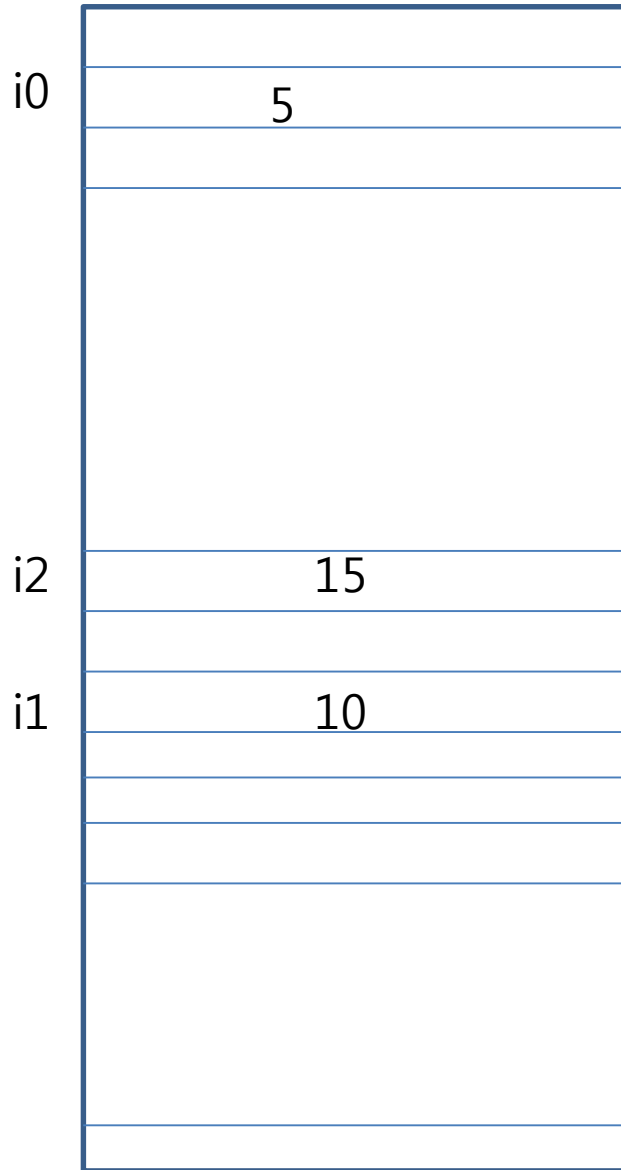
컴퓨터 내부



add	000000..0
10	000000..1
30	00000..10
	11111..11

(32비트)메모리

<일반 변수의 메모리 할당 방법>

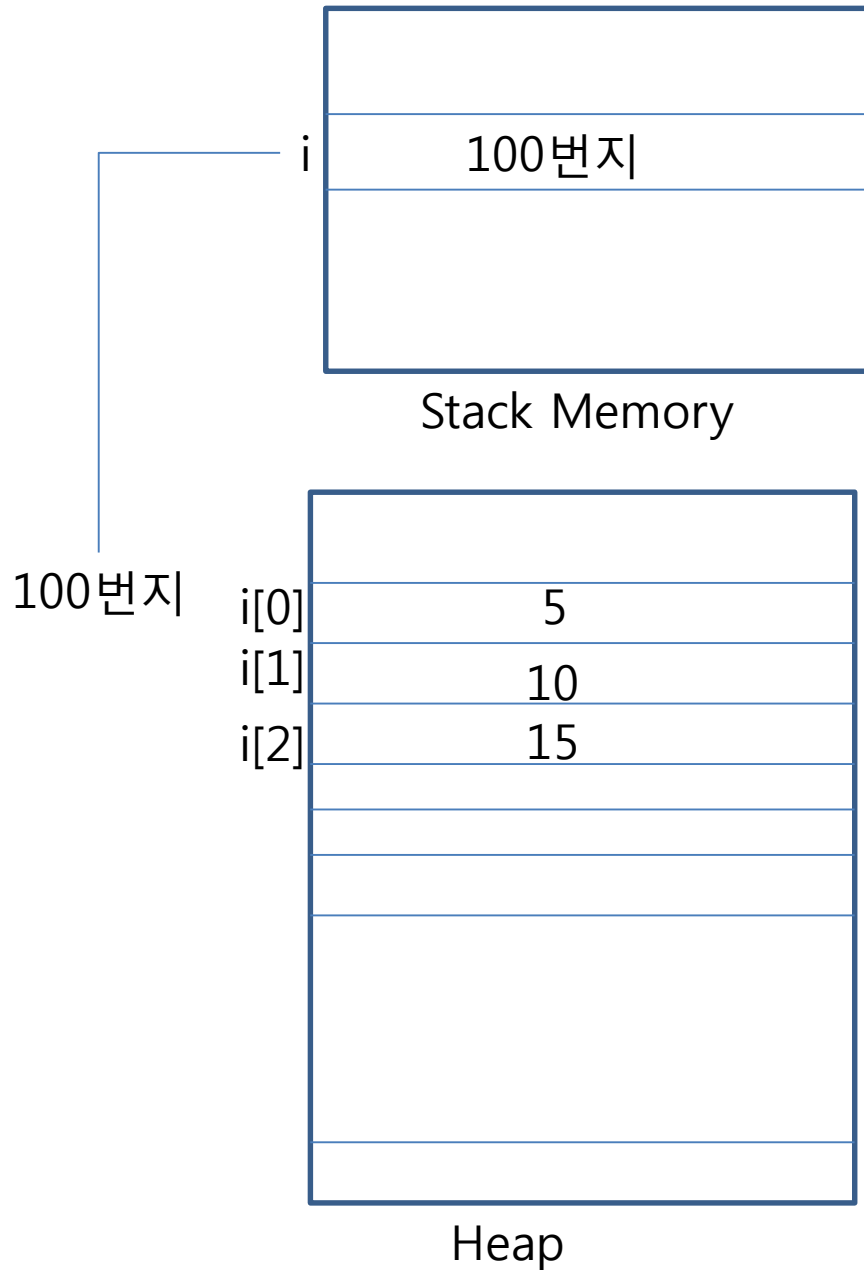


32bit(4바이트)

```
int i0,i1,i2;  
//메모리에 i0,i1,i2  
주소할당
```

```
i0 = 5;  
i1 = 10;  
i2 = 15;  
//할당된 메모리 주소에  
값을 저장한다.
```

<배열 변수 (참조변수) 메모리 할당 방법>



```
int i[] ;
```

// 배열 `i[]`의 레퍼런스 변수 `i`의 주소할당

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

//`i[0]`의 주소 값을 `i`가 가진다.

// 배열 `i[]`의 주소 3개가 연속적으로 할당

```
i[0] = 5;
```

```
i[1] = 10;
```

```
i[2] = 15;
```

//연속적으로 할당된 메모리 주소에 값을 저장한다.

배열 생성과 선언

`int intArray[]` ; 배열에 대한 레퍼런스 변수 `intArray` 생성됨,
데이터타입은 정수형임

`intArray = new int[5]` ; new 연산자를 이용하여 배열 생성하고 `int[0]`의
주소값을 `intArray`에 저장

`int intArray[] = new int[5]` ; 배열의 레퍼런스 변수와 배열을 동시에 만든다.

`int intArray[] = {4,3,2,1,0};`

배열에 대한 레퍼런스 변수 생성되고 배열이 생성되면서 자동으로 초기값이
설정된다. 배열의 인덱스는 0 에서 부터 시작된다

`intArray[0] = 4;`

`intArray[1] = 3;`

`intArray[2] = 2;`

`intArray[3] = 1;`

`intArray[4] = 0;` 와 같다.

문제) 크기가 5인 정수 배열 intArray를 만들고
intArray[0] = 10, intArray[1] = 20, intArray[2] = 30,
intArray[3] = 40, intArray[4] = 50

를 출력하는 프로그램을 만드세요

(힌트: 배열을 생성하고 선언하고 초기화 하여 for문을 이용하여 출력한다)

<배열에 있는 값을 화면에 출력해 보기 -1 >

```
public class ArrayNumber {  
    public static void main(String[] args) {  
  
        int numArray[] = {10,20,30,40,50};  
  
        for(int i=0; i<5; i++) //NumArray.length사용  
            System.out.print("numArray["+i+"] = "+numArray[i]+" ");  
    }  
}
```

<배열에 있는 값을 화면에 출력해 보기 -2 >

```
public class ArrayNumber {  
  
    public static void main(String[] args) {  
  
        final int COUNT;  
  
        int numArray[] = new int[COUNT];  
  
        numArray[0] = 10;  
        numArray[1] = 20;  
        numArray[2] = 30;  
        numArray[3] = 40;  
        numArray[4] = 50;  
  
        for(int i=0; i<COUNT; i++)  
            System.out.print("numArray["+i+"] = "+numArray[i]);  
  
    }  
}
```


배열의 크기와 인덱스

배열도 하나의 객체로 구현 하였고 **length**라는 필드는 배열의 크기를 나타낸다

```
public class ArrayNumber {  
  
    public static void main(String[] args) {  
  
        int numArray[] = new int[5];  
  
        numArray[0] = 10;  
        numArray[1] = 20;  
        numArray[2] = 30;  
        numArray[3] = 40;  
        numArray[4] = 50;  
  
        for(int i=0; i<numArray.length; i++)  
            System.out.print("numArray["+i+"] = "+numArray[i]);  
  
    }  
}
```

예제) 키보드로 부터 5개의 정수를 입력받아
배열에 저장하고 ,
배열에 있는 내용을 화면에 출력하시요.

예제) 키보드로 부터 5개의 문자열을 입력받아
배열에 저장하고 ,
배열에 있는 내용을 화면에 출력하시요.

예제 1)

키보드에서 입력받은 양수5개를 배열에 저장하고, 저장된 배열을 화면에 출력한 후 제일 큰 수를 화면에 출력하는 프로그램을 작성하여라

```
import java.util.Scanner;
public class ArrayAccess{
    public static main(String[] args) {
        Scanner in = new Scanner(System.in);
        //정수형의 배열선언과 생성

        int max = 0; // 가장 큰수 선언 및초기화

        //for문을 이용하여 키보드로부터입력받은 정수를 배열에
        // 저장 및 출력
        //if문을 사용하여 입력받은 수와 max의 수를 비교하여
        //큰 수를 max에 저장한다.

        System.out.print("가장큰수는 : "+max+"입니다.");
    }
}
```

```
import java.util.Scanner;
```

```
public class ArrayMax{
```

```
    public static main(String[] args) {
```

```
        Scanner in = new Scanner(System.in);
```

```
        int intArray[] = new int[5];
```

```
        int max = 0; // 가장 큰수 선언 및 초기화
```

```
        for(int i=0; i<intArray.length;i++) {
```

```
            intArray[i] = in.nextInt();
```

```
        }
```

```
        for(int i=0; i<intArray.length;i++) {
```

```
            System.out.println(intArray[i]);
```

```
        }
```

```
        for(int i=0; i<intArray.length;i++) {
```

```
            if(intArray[i] > max) max = intArray[i];
```

```
        }
```

```
        System.out.print("가장큰수는 : "+max+"입니다.");
```

```
    }
```

```
}
```

예제) 반학생의 이름을 입력하는 프로그램이다. 5명이라고 가정을 하고 아래와 같이 메뉴를 구성하여 프로그래밍 하여라.

```
Menu : Input <1>, Searching<2>, All View<3>, Exit<4> >>>1
Input name >>> hong
Menu : Input <1>, Searching<2>, All View<3>, Exit<4> >>>1
Input name >>> kim
Menu : Input <1>, Searching<2>, All View<3>, Exit<4> >>>1
Input name >>> choi
Menu : Input <1>, Searching<2>, All View<3>, Exit<4> >>>3
name[0] = hong
name[1] = kim
name[2] = choi
name[3] = null
name[4] = null
Menu : Input <1>, Searching<2>, All View<3>, Exit<4> >>>
```

```

import java.util.Scanner;

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

        String name[] = new String[5];
        int count=0;
        int menu;
        Scanner sc = new Scanner(System.in);

        do {
            System.out.print("Menu : Input <1>, Searching<2>, All View<3>, Exit<4> >>>");

            menu = sc.nextInt();
            switch(menu){
                case 1 :
                    System.out.print("Input name >>> ");
                    name[count]=sc.next();
                    count++;
                    break;
                case 2 : System.out.print("Searching Name >>>");
                    String str = sc.next();
                    for(int i=0; i <count; i++) {
                        if(str.equals(name[i])) {
                            System.out.println(str+" searched ...");
                            break;
                        }
                    }
                    break;
                case 3 :
                    for(int i=0; i<count; i++) { //count .. name.length
                        System.out.println("name["+i+"] = "+name[i]);
                    }
                    break;

            }

        }while(menu != 4);
        System.out.println("System End");    } }

```

연습문제

5개의 정수를 입력 받아 배열을 이용하여 오름차순으로 정렬하세요.

결과)

정수입력 : 50

정수입력 : 100

정수입력 : 5

정수입력 : 3

정수입력 : 7

numArray[0] = 3 numArray[1] = 5 numArray[2] = 7

numArray[3] = 50 numArray[4] = 100

50

100

5

3

7

힌트)

키보드로 부터 입력받은 50 100 5 3 7 을 배열에 넣는다

배열의 첫번째 값을 가장 작은 값 min으로 둔다 $\text{min} = \text{numArray}[0]$

min 값과 그다음 번의 배열값과 크기를 비교하여 작다면 min의 값을 비교한 배열값으로 하고 배열의 인덱스를 저장한다

$\text{min} = 50$ 이고

if ($\text{numArray}[i] < \text{min}$)

$\text{min} = \text{numArray}$

$\text{min_num} = i$

배열을 끝까지 비교한 후 배열의 가장 첫번째 값에 가장 작은 값을 두고 첫번째 배열의 값은 가장 작은 값을 가지고 있던 배열에 저장한다

$\text{numArray}[\text{min_num}] = \text{numArray}[j]$

$\text{numArray}[j] = \text{min}$

50과 3이 자리 바꿈한다.

3 100 5 50 7로 바뀐 배열값을 다시 두번째 배열값 100을 min으로 두고 배열끝까지 가장 작은 값을 비교하여 앞에 작업과 동일한 작업을 한다.

3 5 100 50 7

3 5 7 50 100

3 5 7 50 100

```
public class SortEx {

    public static void main(String[] args) {
        int numArray[] = {-50,-100,-5,-3,-7,-10,-2,-3};

        for(int j= 0; j<numArray.length;j++) {
            int min = numArray[j];
            int min_num = j;

            for(int i=j+1;i<numArray.length;i++) {

                if(min < numArray[i]) {
                    min = numArray[i];
                    min_num=i;
                }
            }
            numArray[min_num]=numArray[j];
            numArray[j]=min;
        }

        for(int j= 0; j<numArray.length;j++){
            System.out.print(numArray[j]+" ");
        }

    }
}
```

2차원 배열

2차원 배열의 초기화

```
int intArray[][] = {{0,1,2},{3,4,5},{6,7,8}}
```

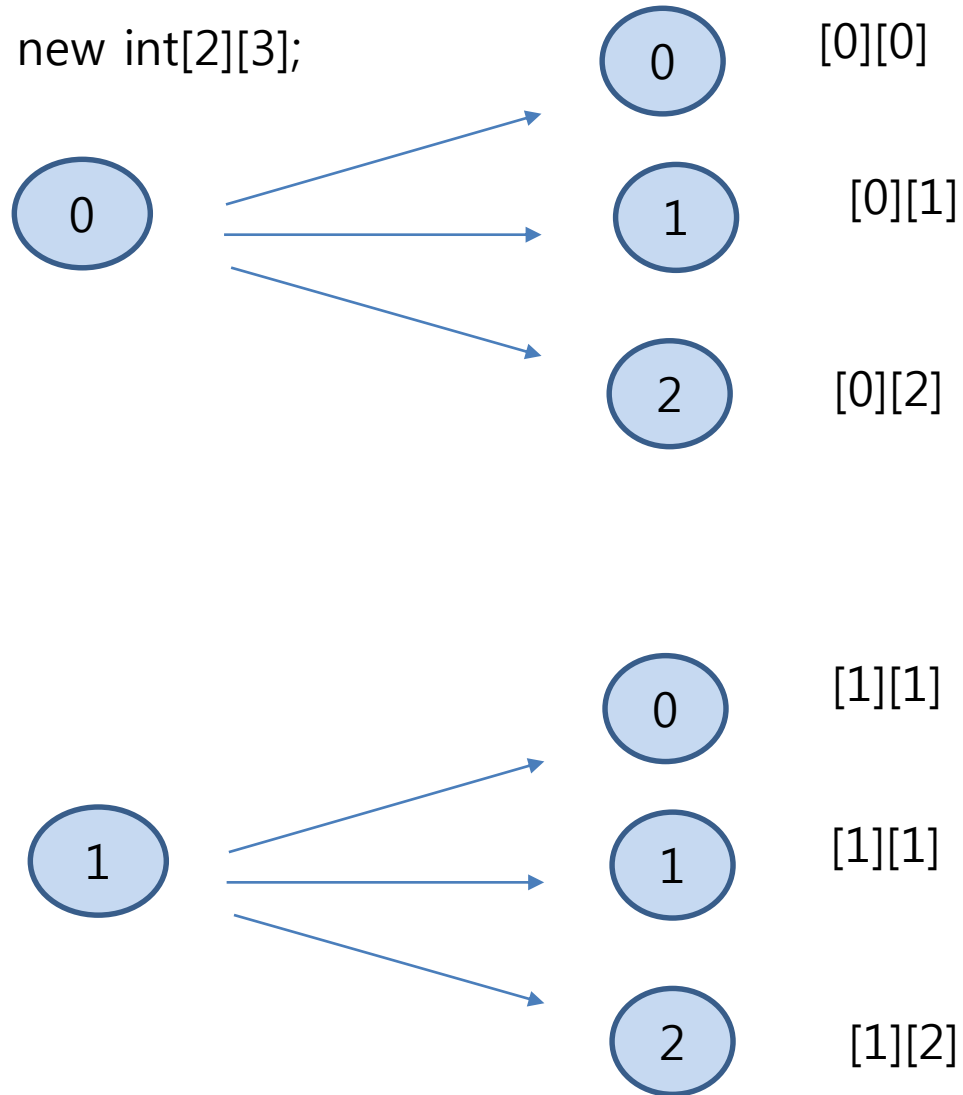
```
intArray[0][0] = 0  intArray[0][1] = 1  intArray[0][2] = 2  
intArray[1][0] = 3  intArray[1][1] = 4  intArray[1][2] = 5  
intArray[2][0] = 6  intArray[2][1] = 7  intArray[2][2] = 8
```

예제)

2차원 배열을 초기화하고 위와 같은 형식으로 배열을 출력하는 프로그램을 만드세요 (행이 바뀌면 한줄 아래 출력한다)

```
system.out.print("intArray["+i+"]["+j+"] = "+intArray[i][j]);
```

```
Int [][] a = new int[2][3];
```



비정방형 배열의 생성과 접근

다음 그림과 같은 비정방형 배열을 만들어 값을 초기화하고 출력하라

10	11	12
20	21	
30	31	32
40	41	

참고)

1행은 10, 2행은 20, 3행은 30, 4행
열은 1씩 증가한다.

```
public class Array2d {  
    public static main(String[] args) {  
  
        int intArray[][] = new int[4][];  
        intArray[0] = new int[3];  
        intArray[1] = new int[2];  
        intArray[2] = new int[3];  
        intArray[3] = new int[2];  
  
        for(int i=0; i<intArray.length;i++)  
            for(int j=0; j<intArray[i].length;j++)  
                intArray[i][j] = (i+1)*10+j;  
  
        for(int i=0; i<intArray.length;i++) {  
            for(int j=0; j<intArray[i].length;j++)  
                System.out.print(intArray[i][j]+" ");  
            System.out.println();  
        }  
    }  
}
```


메소드에서 배열 리턴 (143p 참조)

참고)

메소드인 함수의 구성

```
int plusAB(int a, int b) return a+b;
```

enum type (열거형 상수, 나열 타입)

1. 열거된 상수중 하나의 값을 취한다.
2. 열거된 상수 이외의 값을 사용할 수 없다.

사용법

```
enum 열거데이터형이름 {상수값1, 상수값2,...};
```

```
enum Colors {RED, BLACK, YELLOW};  
Colors color = Colors.RED;
```

for-each문 과의 사용법

```
enum Week {일,월,화,수,목,금,토} //상수값을 스트링으로 표시하지  
않아도 된다.
```

```
for(Week day : Week.values())  
    System.out.print(day+"요일");
```

```
public class EnumEx {  
  
    public static main(String[] args) {  
  
        enum Week {일,월,화,수,목,금,토};  
  
        for(Week day : Week.values())  
            System.out.print(day+"요일");  
  
    }  
}
```

For(Week day : Week.values())는
day = Week.values(0); day = Week.values(1);
day = Week.values(2); day = Week.values(3); day = Week.values(4);
day = Week.values(5); day = Week.values(6); 을 수행한다.

main() 메소드 인자 전달

c:\wjava Hello option1, option2,....

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

        double sum=0.0;

        for(int i=0; i<args.length;i++) {
            sum+=Double.parseDouble(args[i]);
        }

        System.out.println("합계: "+sum ) ;
        System.out.println("평균: "+sum /args.length) ;
    }
}
```