

JAVA

꼭 필요한 가장 중요한 문법 !!

변수 : 값을 저장하기 위한 기억공간
데이터를 저장하기 위한 그릇(상자)으로 생각하면 된다.

프로그램에서 데이터는 크게

숫자 : 90, 80.23

문자 : 'a'

문자열 : "홍길동"

90

ii
↙ 변수명

```
int ii;  
double dd;  
char cc;  
String ss;
```

```
ii=90;  
dd=80.23 ;  
cc='a';  
ss="홍길동";
```

데이터를 담을 때
데이터에 알맞은 변수(그릇)를 사용해야 한다.

즉 ii 변수는 정수만 담는다
dd변수는 실수만을 담는다
cc변수는 문자만을 담는다
ss변수는 문자열만을 담는다

변수명으로 어떤 이름이 좋을까?

(변수명 규칙은 영문소문자로 시작하는 변수명이면 ok) 그러나 의미 있게 주는 것이 좋다

```
int kor_score=90; //국어성적이라면  
double kor_avg =80.23; //국어평균이라면  
char grade ='a' ; //학점이라면  
String name ="홍길동"; //이름이라면
```

```
System.out.println( "국어성적= " + kor_score );  
System.out.println( "국어평균= " + kor_avg );  
System.out.println( "학점= " + grade );  
System.out.println( "이름=" + name );
```



내정보를 변수에 담아 출력하기



사칙연산 프로그램

두 개의 정수의 사칙연산의 결과를 출력하기

//1. 필요한 변수 선언하기

```
int su1;
```

```
int su2;
```

```
int sum , minus , multiple , divide ;
```

//2. 변수에 값을 지정합니다. (여기서는 임의 두 수를 저장합니다.)

```
su1=6;
```

```
su2=2;
```

//3. 계산을 합니다. (사칙연산을 수행합니다.)

```
sum = su1+ su2 ;
```

```
minus = su1 - su2 ;
```

```
multiple = su1 * su2 ;
```

```
divide = su1 / su2;
```

//4. 결과를 출력합니다.(화면에 결과 출력)

```
System.out.println( “두 수의 합 = ” + sum );
```

```
System.out.println( “두 수의 차= ” + minus );
```

```
System.out.println( “두 수의 곱= ” + multiple );
```

```
System.out.println( “두 수의 나누기의 몫 = ” + divide );
```

사칙연산에서 하나 더 추가합니다.
자바는 나머지연산자 %를 제공합니다.

```
int mod = 10% 2;  
System.out.println( mod );    // 값을 예상해 볼까요 ?  
◆ 나머지 연산자 % 는 짝수, 홀수 판단문제에서 사용됩니다.
```

■ 성적을 처리해 봅니다.

학생이름과 국어, 영어, 수학 점수를 가지고 합계와 평균을 구합니다.

1. 필요한 변수를 선언합니다

```
String name; // 이름을 저장할 변수  
int kor;      // 국어 점수를 저장할 변수  
int eng;      // 영어 점수를 저장할 변수  
int math;     //수학 점수를 저장할 변수  
int sum ;     // 합계를 저장할 변수  
double avg;   // 평균을 저장할 변수
```



2. 변수에 데이터(값) 를 저장합니다.

```
name="오징어"  
kor=90;  
eng=100;  
math=89;
```

3. 총점과 평균을 구합니다.

```
sum = kor+ eng+ math ;  
avg = sum/ 3.0;
```

4. 계산된 결과를 이름과 함께 출력합니다.

```
System.out.println("====성적표====");  
System.out.println( name );  
System.out.println( sum );  
System.out.println( avg );
```

만약에 너가 짝수라면
만약에 너가 남자라면
만약에 너가 총점이 90점 이상 이라면
만약에 너가 평균이 70점 이상 이라면

if



//변수의 값이 짝수라면 출력하는 프로그램

```
int su=6;
```

```
if( su % 2 == 0)    // su변수의 값이 짝수라면
```

```
{
```

```
    System.out.println(“넌 짝수야” + su);
```

```
    System.out.println(“ 난 짝수가 좋아 ~^^”);
```

```
}
```

//총점이 90이상인 경우만 사탕을 줍니다.

```
int sum=93;
```

```
if( sum >= 90){
```

```
    System.out.println( “넌 총점이 90점 이상이구나 ”);
```

```
    System.out.println(“ 맛있는 사탕을 줄게”);
```

```
}
```

```
double avg =83.3;
```

```
if( avg >=80 ){
```

```
    System.out.println( “넌 80점이상이니 학점이 B 야”);
```

```
}
```

만약에 변수의 값이 짝수라면
그렇지 않다면 홀수입니다.

```
int su=19;
if( su%2 ==0 )
{
    System.out.println( “짝수이군요”);
}else{           //위의 조건이 아니라면
    System.out.println(“ 홀수이군요”);
}
```

반복: 똑같은 명령을 여러 번 반복해야 할 때 사용함

◆줄넘기를 세 번 하고 싶을 때

```
System.out.println("줄넘기를 한다");  
System.out.println("줄넘기를 한다");  
System.out.println("줄넘기를 한다.");
```



```
for(int i=1; i<=3; i++){  
    System.out.println("줄넘기를 한다");    //줄넘기를 한대를 세 번 반복  
}
```

◆ “좋아해” 5번 출력하기

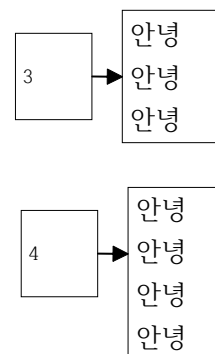
```
int count=1;  
while( count <= 5 ){    //조건이 만족될 동안  
    System.out.println("좋아해") ;  
    count= count +1;  
}
```

변수명	값	반복조건
count	1	참
	2	참
	3	참
	4	참
	5	참
	6	거짓

```
for(int i=1 ; i <=5 ; i++){  
    System.out.println("좋아해") ;  
}
```

문제

1) 반복횟수를 변수에 지정하고 그 횟수 만큼 “안녕” 출력하기

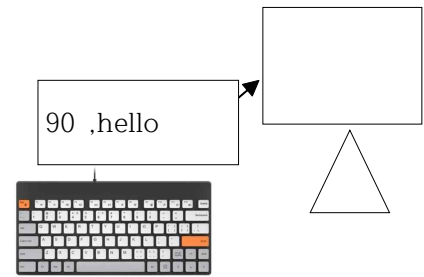


2) 숫자(1~10 사이의 값)를 변수에 지정하고 (저장하고)
int su=10; (예시)

그 수가 짝수이면 4번 “짝수”라고 출력하고
그 수가 홀수이면 3번 “홀수”라고 출력하기

```
Scanner sc = new Scanner(System.in) ;
int score = sc.nextInt() ;
String name = sc.next();
System.out.println("점수 =" + score ) ;
System.out.println("이름은=" + name );

sc.close();
```



내정보를 입력 받아
출력하는 프로그램을 작성해 보세요
내정보는 임의로 5가지 정도 해 보세요
(필요한 변수는 선언해서 사용하세요 !)



■ 성적처리 프로그램

이름, 국어, 영어, 수학
성적을 입력 받아 (Scanner 사용하세요!)
이름, 총점, 평균, 학점을 출력하시오
(평균이 90이상이면 A 80점 이상이면 B 70점 이상이면 C 나머지는 D 가 출력되도록 하세요 !)





문자열 처리하는 프로그램

```
String name = "2012-09-10";  
String year = name.substring(0 ,4) ;  
String month = name.substring(5, 7);  
String day  = name.substring(7, 9) ;
```

```
System.out.println( year );  
System.out.println( month );  
System.out.println( day );
```

//문자열 비교

```
String message="ok" ;  
boolean result = message.equals("ok") ;  
System.out.println( result );
```

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

```
while( true){  
    String value  = sc.next() ;  
    if( value.equals("n") ) break;  
}
```

■ 입력받은 문자열을 5번 출력하시오~

방과후 5 일차 : 사용자정의 자료형 class

자료형 복습해 볼까요?

int , double , char ,String

```
age=18;  
height=181.8;  
name="성기훈";
```

```
int age;  
double height;  
String name;
```

학생정보를 저장할 변수



3개의 변수를 사용하네요? , 더 많을 수도 있겠죠?

한 학생의 정보를 담을 수 있는 하나의 자료형이 제공된다면 편리하겠지요?

바로 class 가 구조화된 데이터를 저장할 수 있는 자료형입니다. (사용자:즉 프로그래머 가 만드는 자료형, 사용자정의자료형)
구조화되었다는 것은 학생정보안에는 다른 여러개로 구성되었습니다.

이렇게 그 안에 무언가를 품고 있을 때 구조화되었다고 합니다. 어렵게 생각하지 마세요 !!!!

학생		
18	181.9	성기훈
나이	키	이름

학생 데이터는
여러개의 정보로 구성되었습니다.
이런 데이터를 구조화된 데이터
라고 말합니다.
어렵지 않아요^^

```

class Student{
    private int age;
    private double height ;
    private String name;

    //생성자
    public Student() { }
    //생성자 , 변수 생성시 변수의 값 초기화 담당
    public Student( int age, double height , String name){
        this.age = age;
        this.height = height;
        this.name = name;
    }

    //Setter setter
    //변수에 값넣기 , 변경이 가능함
    public void setAge(int age) {
        this.age = age;
    }

    public void setHeight( double height) {
        this.height = height;
    }

    public void setName(String name) {
        this.name = name;
    }

    //게터 getter

    public int getAge(){
        return age;
    }

    public double getHeight(){
        return height;
    }

    public String getName(){
        return name;
    }
}

```

```

public class Test{

    public static void main(String[] args) {
        // 생성자를 통한 변수 값 세팅
        Student student1 = new Student(18, 181.9 , "성기훈") ;
        System.out.println( student1.getAge() );
        System.out.println( student1.getHeight() );
        System.out.println ( student1.getName() );

        // Setter 사용하여 변수 값 세팅하기 setter

        Student student2 = new Student( );
        student2.setAge(16);
        student2.setHeight(160.2);
        student2.setName("이연주");

        System.out.println( student1.getAge() );
        System.out.println( student1.getHeight() );
        System.out.println ( student1.getName() );

    }
}

```

사칙연산을 위한 클래스입니다. // 두 수를 저장할 수 있는 변수를 정한다. (멤버변수라고 말함)

```
class Calculator{
```

```
private int su1;
```

```
private int su2;
```

```
// 생성자
```

```
//세터
```

```
//게터 만들기
```

```
}
```

■ 위의 Calculator 클래스를 사용하여 두 수를 저장하여 두 수를 출력해 보세요

■ 자신이 담고 싶은 정보를 저장할 수 있는 클래스를 만들고 값을 세팅하고 출력해보세요

예) 환자정보를 담을 수 있는 클래스

환자정보 ; 환자이름, 전화번호, 주소, 혈액형

■ Calculator 클래스에 덧셈기능, 뺄셈 기능 추가해 보자

```
class Calculator {
    private int su1;
    private int su2;

    //생성자
    public Calculator() {}
    public Calculator(int su1, int su2) {
        this.su1 = su1;
        this.su2 = su2;
    }
    //세터 (필요하면 추가한다.)

    // 기능추가 ( 뺄셈기능 덧셈기능)

    public int sum( ) {
        int total=0;
        total = su1 + su2 ;
        return total;
    }

    public int minus( ){
        int sub=0;
        sub = su1- su2;
        return sub;
    }
    //
```

//getter (필요하면 추가한다.)

//정보를 한 번에 출력하는 매서드 이용해 보자

```
public String toString() {
    String str= su1 + " ," + su2 ;
    return str;
}
```

}

■ 곱셈기능 추가해 보자

■ 나눗셈 기능(몫 구하기) 추가해 보자

```
public Calculator_Test{

    public static void main(String[] args){
        //생성자를 통한 멤버변수에 값 세팅
        Calculator c = new Calculator( 10, 2) ;
        int hap = c.sum();
        System.out.println( "합=" + hap ) ;

        int cha= c.minus() ;
        System.out.println( "차=" + cha );

        // c객체의 내용을 출력해보자
        System.out.println( c.toString() );

    }

}
```

```
class Score{
```

```
private String name ;  
private int kor;  
private int eng;  
private char grade; // 학점 , A, B, C,F
```

```
//생성자
```

```
public Score() {}  
public Score( String name, int kor, int eng) {
```

```
    this.name = name ;  
    this.kor = kor ;  
    this.eng = eng ;
```

```
}
```

```
public void calcGrade()  
    int tot = kor + eng ;  
    double avg = tot/2.0;  
    if( avg >=90) {  
        grade='A';  
    }else if( avg>=80) {  
        grade='B';  
    }else if( avg >=70){  
        grade='C';  
    }else{  
        grade='D';  
    }  
}
```

```
//필요하면 세터
```

```
//필요하면 getter 사용
```

```
public String toString() {  
    return name + ", " + kor + ", " + eng + "," + grade;  
}
```

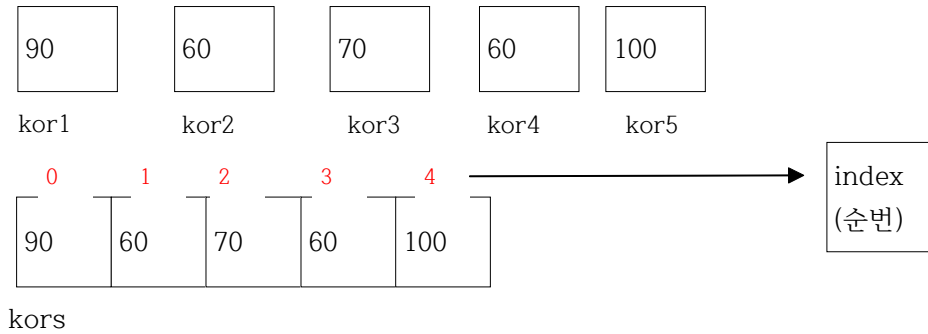
```
}
```

```
class Score_Test{  
public static void main(String[] args) {  
  
    Score s = new Score( '홍길동' , 100,80 );  
    s.calcGrade();  
  
    // s객체변수 의 정보 출력 ,  
  
    System.out.println( s.toString() );  
}  
}
```

■ 배열연습하기

변수가 많이 필요할 때 , 많은 자료를 저장해야 한다면....

예) 5명의 국어 점수를 저장해야 한다면



변수 5개를 사용하는 것과 배열을 사용하는 것 어떤 잇점이 있을까요?

```
int kor1, kor2 , kor3, kor4, kor5 ;

kor1=90;
kor2=60;
kor3=70;
kor4=60;
kor5=100;
합계를 구한다면 ?
int sum = kor1 + kor2 + kor3 + kor4 + kor5 ;
```

배열:

같은크기 (같은 자료형)의 변수를 연속적으로 사용하는 것을 말한다.

배열은 인덱스(순번)를 이용해서 각 저장공간에 접근할 수 있다.

```
int[] arr = new int [5] ;
arr[0] =90;
arr[1]= 60;
arr[2] =70;
arr[3] =60;
arr[4]=100;

int sum = arr[0] + arr[1] + arr[2] + arr[3] + arr[4] ;
```



배열은 반복문으로 해결할 수 있습니다.

```
for( int i=0 ; i <5; i++){  
    sum = sum + arr[ i ];  
}  
  
System.out.println( sum ) ;
```

처리해야 할 점수가 1000개라면
변수 1000개를 사용하시겠어요 배열을 사용하고 싶으세요?
`int[] arr = new int[1000]; // 문제 없죠?`

변수 하나씩 1000개 선언?
미칠거 같네요 π

- 문제 double형 변수 5개를 저장할 수 있는 배열을 선언 해 주세요
- 문제 문자열을 5개 저장할 수 있는 문자열 배열을 선언 해 주세요
- 위의 배열에 임의 값을 저장하시고 배열의 내용을 모두 출력해 주세요
(자기가 좋아하는 숫자나 문자열(좋아하는 연예인 이름 등) 입력하시면 됩니다.)