



2024학년도 1학기

신입생 Java 교육

교육부장 20 이어진

교육을 시작하기 앞서...

1. 컴 퓨 터 프 로 그 래 밍 1 강 의 기 반
2. 이 론 설 명 + 실 습 문 제 풀 이
3. 모 든 실 습 문 제 는 백 준 에 서
4. 교 육 중 언 제 나 질 문 가 능



교육 커리큘럼

1주차 변수, 연산자 + 조건문(if)

2주차 조건문(switch) + 반복문

3주차 배열

<중간고사>

4주차 객체, 메소드 오버로딩과 생성자

5주차 상속과 오버라이딩

6주차 다형성과 인터페이스

7주차 예외처리


<기말고사>



백준

BAEKJOON>
ONLINE JUDGE

회원가입 | 로그인

문제 ▾ 문제집 대회 ² 채점 현황 랭킹 게시판 그룹 더 보기 ▾ 

문제	문제	출처	ICPC
<ul style="list-style-type: none">> 전체 문제> 문제 출처> 단계별로 풀어보기> 알고리즘 분류> 추가된 문제> 문제 순위	<ul style="list-style-type: none">> 푼 사람이 한 명인 문제> 아무도 못 푼 문제> 최근 제출된 문제> 최근 풀린 문제> 랜덤	<ul style="list-style-type: none">> ICPC> Olympiad> 한국정보올림피아드> 한국정보올림피아드시.도지역본선> 전국 대학생 프로그래밍 대회 동아리 연합> 대학교 대회> 카카오 코드 페스티벌> Coder's High	<ul style="list-style-type: none">> Regionals> World Finals> Korea Regional> Africa and the Middle East Regionals> Europe Regionals> Latin America Regionals> North America Regionals> South Pacific Regionals

<https://www.acmicpc.net/>

백준

BAE/KJOON>
ONLINE JUDGE

문제 ▾ 문제집 ▾ 대회 3 채점 ▾

검색

2557

문제 ✓

출처
블로그
게시판

코딩테스트
이것만
알고 가자

최근 코딩테스트 출제 트렌드
얼마 안 남은 코딩테스트, 실전에서 바로 활용할 수 있는 필수 알고리즘 노트 만들기
스파르타코딩클럽

검색 결과 607개 (0.014초)

2557번 - Hello World
시간 제한: 1초 메모리 제한: 128MB
Hello World!를 출력하시오.
제출 맞은 사람 채점 현황

BAE/KJOON>
ONLINE JUDGE

win929 | 설정 | 로그아웃 | ★

문제 ▾ 문제집 ▾ 대회 3 채점 현황 ▾ 랭킹 ▾ 게시판 ▾ 그룹 ▾ 더 보기 ▾

검색

어드민

검색 결과

문제
문제집 ✓
출처
블로그
게시판

코딩테스트
이것만
알고 가자

최근 코딩테스트 출제 트렌드
얼마 안 남은 코딩테스트, 실전에서 바로 활용할 수 있는 필수 알고리즘 노트 만들기
스파르타코딩클럽

적용하기 >

검색 결과 3개 (0.002초)

[JAVA] 2024 어드민 교육 3주차
만든 사람: win929 문제: 3
[JAVA] 2024 어드민 교육 3주차

[JAVA] 2024 어드민 교육 2주차
만든 사람: win929 문제: 3
[JAVA] 2024 어드민 교육 2주차

[JAVA] 2024 어드민 교육 1주차
만든 사람: win929 문제: 4
[JAVA] 2024 어드민 교육 1주차

<https://www.acmicpc.net/>

연습문제 - 백준 2557번

5 2557번

Hello World 성공



시간 제한	메모리 제한	제출	정답	맞힌 사람	정답 비율
1 초	128 MB	1104724	445330	311643	39.385%

주의사항

문제

Hello World!를 출력하시오.

입력

없음

출력

Hello World!를 출력하시오.

예제 입력 1 복사

예제 출력 1 복사

```
Hello World!
```

5 2557번

제출

맞힌 사람

숏코딩

재채점 결과

채점 현황

내 제출

난이도 기여 ↗

질문 게시판

Hello World

언어

Java 8

언어 설정

소스 코드 공개

☒ 공개

☐ 비공개

☐ 맞았을 때만 공개

소스 코드

```
1 public class Main {
2     public static void main(String[] args) {
3         
4     }
5 }
6
```

해설

문제

Hello World!를 출력하시오.

입력

없음

출력

Hello World!를 출력하시오.

예제 입력 1 복사

예제 출력 1 복사

Hello World!

```
1 public class Main {  
2     public static void main(String[] args) {  
3         System.out.println("Hello World!");  
4     }  
5 }
```

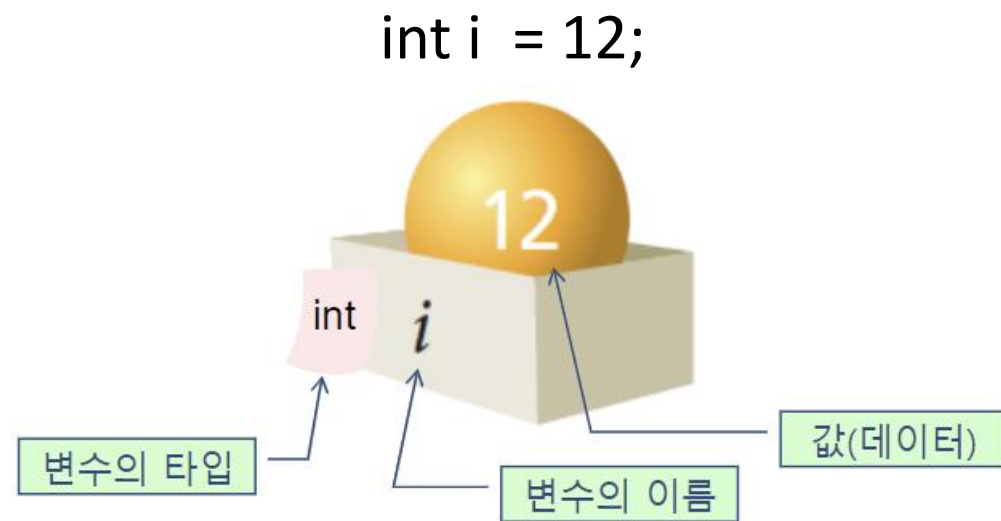

Part 1

변수, 상수, 자료형



Part 1 변수 (Variable) 란 ?

하나의 값을 저장할 수 있는 기억공간(상자)



Part 1 변수의 타입 (Data type)

자료형	데이터	메모리 크기	표현 가능 범위
boolean	참과 거짓	1 바이트	true, false
char	문자	2 바이트	모든 유니코드 문자
byte	정수	1 바이트	-128 ~ 127
short		2 바이트	-32768 ~ 32767
int		4 바이트	-2147483648 ~ 2147483647
long		8 바이트	-9223372036854775808 ~ 9223372036854775807
float	실수	4 바이트	$\pm(1.40 \times 10^{-45} \sim 3.40 \times 10^{38})$
double		8 바이트	$\pm(4.94 \times 10^{-324} \sim 1.79 \times 10^{308})$

Part 1 변수의 타입 (Data type)

자료형	데이터	메모리 크기	
boolean	참과 거짓	1 바이트	true, false
char	문자	2 바이트	모든 유니코드 문자
byte	정수	1 바이트	-128 ~ 127
short		2 바이트	-32768 ~ 32767
int		4 바이트	-2147483648 ~ 2147483647
long		8 바이트	-9223372036854775808 ~ 9223372036854775807
float	실수	4 바이트	$\pm(1.40 \times 10^{-45} \sim 3.40 \times 10^{38})$
double		8 바이트	$\pm(4.94 \times 10^{-324} \sim 1.79 \times 10^{308})$

DEC	HEX	OCT	Char	DEC	HEX	OCT	Char	DEC	HEX	OCT	Char
0	00	000	Ctrl-@ NUL	43	2B	053	+	86	56	126	V
1	01	001	Ctrl-A SOH	44	2C	054	,	87	57	127	W
2	02	002	Ctrl-B STX	45	2D	055	-	88	58	130	X
3	03	003	Ctrl-C ETX	46	2E	056	.	89	59	131	Y
4	04	004	Ctrl-D EOT	47	2F	057	/	90	5A	132	Z
5	05	005	Ctrl-E ENQ	48	30	060	0	91	5B	133	[
6	06	006	Ctrl-F ACK	49	31	061	1	92	5C	134	\
7	07	007	Ctrl-G BEL	50	32	062	2	93	5D	135]
8	08	010	Ctrl-H BS	51	33	063	3	94	5E	136	^
9	09	011	Ctrl-I HT	52	34	064	4	95	5F	137	_
10	0A	012	Ctrl-J LF	53	35	065	5	96	60	140	`
11	0B	013	Ctrl-K VT	54	36	066	6	97	61	141	a
12	0C	014	Ctrl-L FF	55	37	067	7	98	62	142	b
13	0D	015	Ctrl-M CR	56	38	070	8	99	63	143	c
14	0E	016	Ctrl-N SO	57	39	071	9	9A	64	144	d
15	0F	017	Ctrl-O SI	58	3A	072	:	9B	65	145	e
16	10	020	Ctrl-P DLE	59	3B	073	;	9C	66	146	f
17	11	021	Ctrl-Q DC1	60	3C	074	<	9D	67	147	g
18	12	022	Ctrl-R DC2	61	3D	075	=	9E	68	150	h
19	13	023	Ctrl-S DC3	62	3E	076	>	9F	69	151	i
20	14	024	Ctrl-T DC4	63	3F	077	?	9A	6A	152	j
21	15	025	Ctrl-U NAK	64	40	078	@	9B	6B	153	k
22	16	026	Ctrl-V SYN	65	41	101	A	9C	6C	154	l
23	17	027	Ctrl-W ETB	66	42	102	B	9D	6D	155	m
24	18	030	Ctrl-X EOT	67	43	103	C	9E	6E	156	n
25	19	031	Ctrl-Y ABEL	68	44	104	D	9F	6F	157	o
26	1A	032	Ctrl-Z EOI	69	45	105	E	9A	70	160	p
27	1B	033	Ctrl-[ESC	70	46	106	F	9B	71	161	q
28	1C	034	Ctrl-\	71	47	107	G	9C	72	162	r
29	1D	035	Ctrl-]	72	48	110	H	9D	73	163	s
30	1E	036	Ctrl-^	73	49	111	I	9E	74	164	t
31	1F	037	Ctrl-_	74	4A	112	J	9F	75	165	u
32	20	040	Space	75	4B	113	K	9A	76	166	v
33	21	041	!	76	4C	114	L	9B	77	167	w
34	22	042	"	77	4D	115	M	9C	78	170	x
35	23	043	#	78	4E	116	N	9D	79	171	y
36	24	044	\$	79	4F	117	O	9E	7A	172	z
37	25	045	%	80	50	120	P	9F	7B	173	{
38	26	046	&	81	51	121	Q	9A	7C	174	
39	27	047	'	82	52	122	R	9B	7D	175	}
40	28	050	(83	53	123	S	9C	7E	176	~
41	29	051)	84	54	124	T	9D	7F	177	DEL
42	2A	052	*	85	55	125	U	made by Lee Jae-wook			

```
char a1 = 'A';
char a2 = 65;
```

```
System.out.println(a1);
System.out.println(a2);
```

A
A

Part 1

변수의 선언 방법

타입 변수명;

```
int score;    // 선언  
score = 100;  // 초기화
```

```
int score = 100;  
// 선언과 동시에 초기화
```

Part 1

변수의 이름을 짓는 방법

1. 대소문자가 구분되며 길이에 제한이 없다.
 - Admin와 admin는 서로 다른 것으로 간주된다.
2. 예약어(Reserved word)를 사용해서는 안 된다.
 - true는 예약어라 사용할 수 없지만, True는 가능하다.
3. 숫자로 시작해서는 안 된다.
 - top10은 허용하지만, 7up은 허용되지 않는다.
4. 특수문자는 '_'와 '\$'만을 허용한다.
 - \$harp은 허용되지만 S#arp는 허용되지 않는다.

[자바의 예약어]

boolean	if	interface	class	true
char	else	package	volatile	false
byte	final	switch	while	throws
float	private	case	return	native
void	protected	break	throw	implements
short	public	default	try	import
double	static	for	catch	synchronized
int	new	continue	finally	const
long	this	do	transient	enum
abstract	super	extends	instanceof	null

Part 1

변수, 상수, 리터럴

- ▶ 변수(variable) – 하나의 값을 저장하기 위한 공간

```
int score = 100;
```

```
score = 200;
```

```
char ch = 'A';
```

```
String str = "abc";
```

- ▶ 상수(constant) – 한 번만 값을 저장할 수 있는 공간

```
final int MAX = 100;
```

```
MAX = 200; // 에러
```

- ▶ 리터럴(literal) – 그 자체로 값을 의미하는 것
쉽게 말해서 변수에 저장되는 값

실습문제 - 백준 1000번

5 1000번	A+B	성공	다국어	☆	한국어 ▼
시간 제한	메모리 제한	제출	정답	맞힌 사람	정답 비율
2 초	128 MB	1025238	406071	280704	39.365%

힌트

1. 입력한 숫자를 변수에 저장하는 방법은 Scanner를 사용해서 .nextInt()를 사용하면 된다.

해설

문제

두 정수 A와 B를 입력받은 다음, A+B를 출력하는 프로그램을 작성하시오.

입력

첫째 줄에 A와 B가 주어진다. ($0 < A, B < 10$)

출력

첫째 줄에 A+B를 출력한다.

예제 입력 1 복사

```
1 2
```

예제 출력 1 복사

```
3
```

```
1 import java.util.Scanner;
2 import java.util.*;
3 public class Main {
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6
7         int a = sc.nextInt();
8         int b = sc.nextInt();
9
10        System.out.println(a+b);
11
12        sc.close();
13    }
14 }
```

Part 1

형 변환 (Casting)

형 변환이란?

- 값의 타입을 다른 타입으로 변환하는 것이다.
- boolean을 제외한 7개의 기본형은 서로 형변환이 가능하다.

```
float f = 1.6f;
```

```
int i = (int)f;
```

변 환	수 식	결 과
int → char	(char) 65	'A'
char → int	(int) 'A'	65
float → int	(int) 1.6f	1
int → float	(float) 10	10.0f

Part 1

형 변환 (Casting)

형 변환이란?

- 값의 타입을 다른 타입으로 변경
- boolean을 제외한 7가지 기본 타입

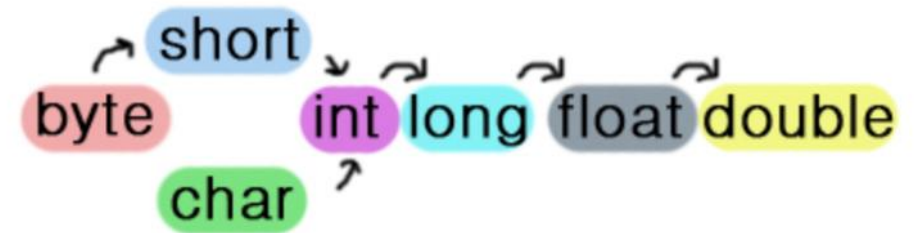
float

int

변 환		
int → char		
char → int	(int) 'A'	65
float → int	(int) 1.6f	1
int → float	(float) 10	10.0f

자료형	데이터	메모리 크기	표현 가능 범위
boolean	참과 거짓	1 바이트	true, false
char	문자	2 바이트	모든 유니코드 문자
byte	정수	1 바이트	-128 ~ 127
short		2 바이트	-32768 ~ 32767
int		4 바이트	-2147483648 ~ 2147483647
long		8 바이트	-9223372036854775808 ~ 9223372036854775807
float	실수	4 바이트	$\pm(1.40 \times 10^{-45} \sim 3.40 \times 10^{38})$
double		8 바이트	$\pm(4.94 \times 10^{-324} \sim 1.79 \times 10^{308})$

Part 1 묵시적 형 변환

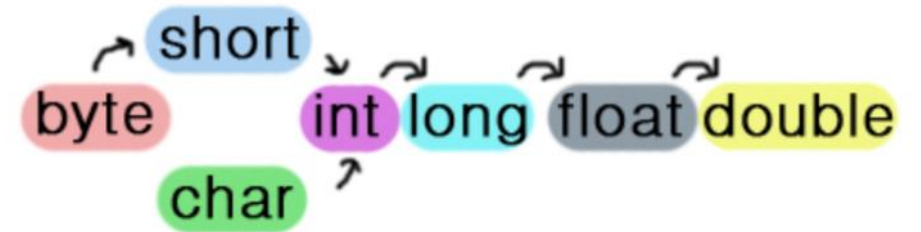


형변환 연산자를 사용하지 않아도
자동적으로 이루어지는 경우

```
int a = 5;  
double b = a;
```



Part 1 묵시적 형 변환



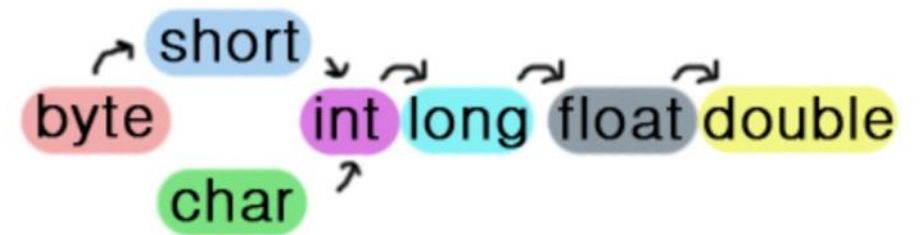
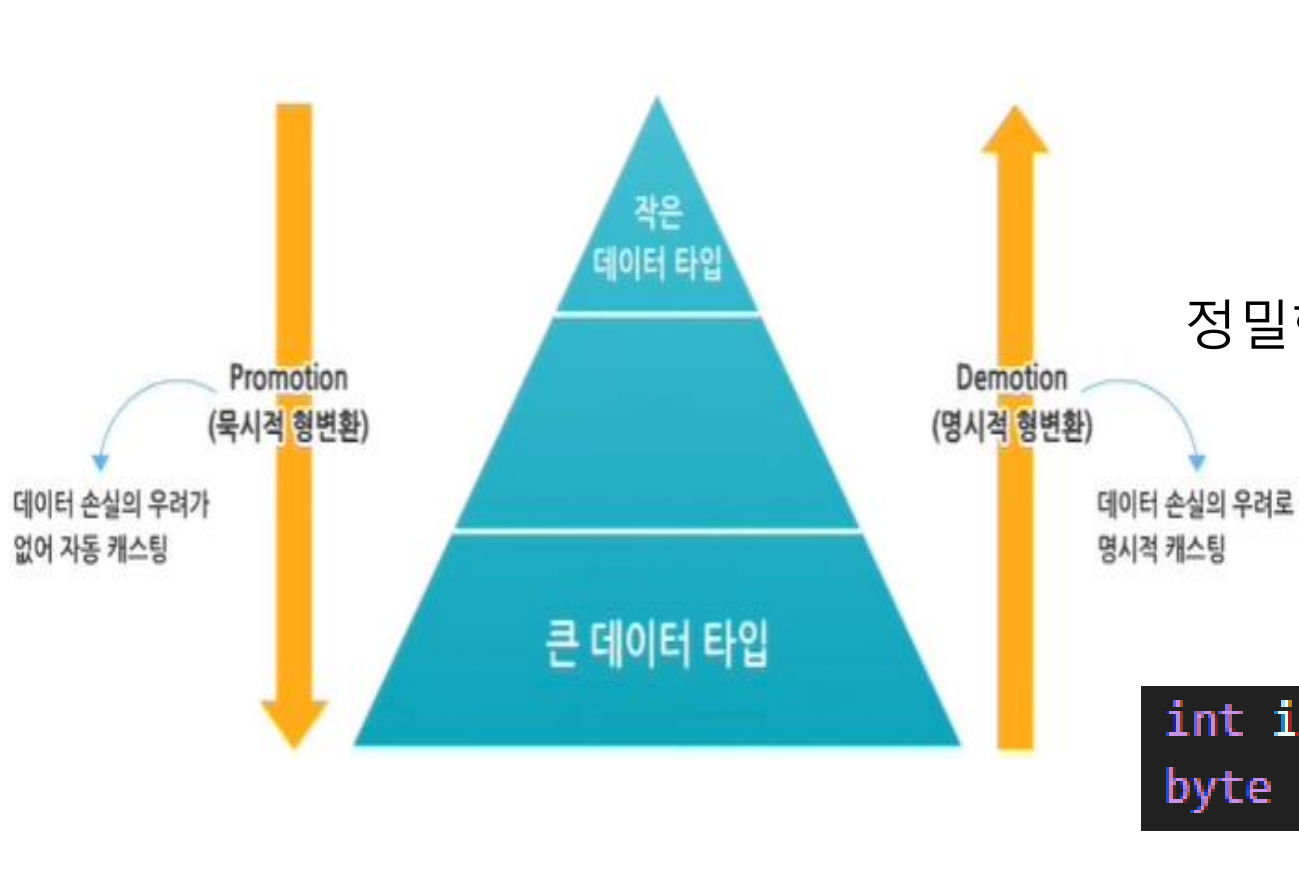
형변환 연산자를 사용하지 않아도
자동적으로 이루어지는 경우

```
int a = 5;  
double b = a;
```

int

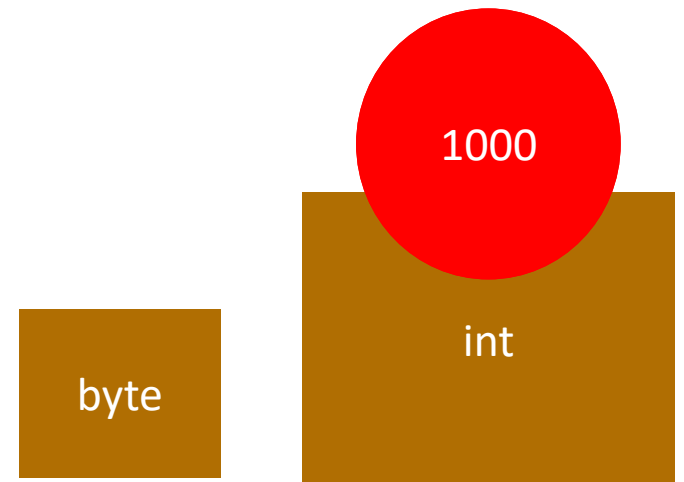
5
double

Part 1 명시적 형 변환

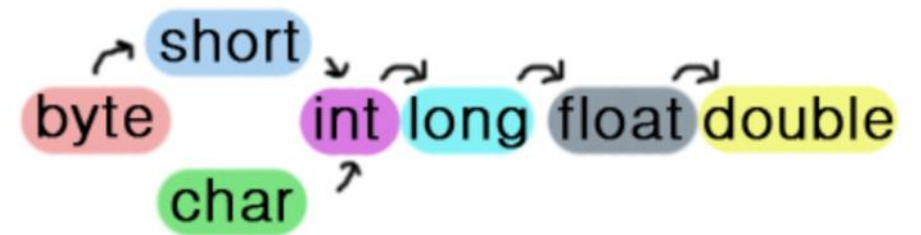
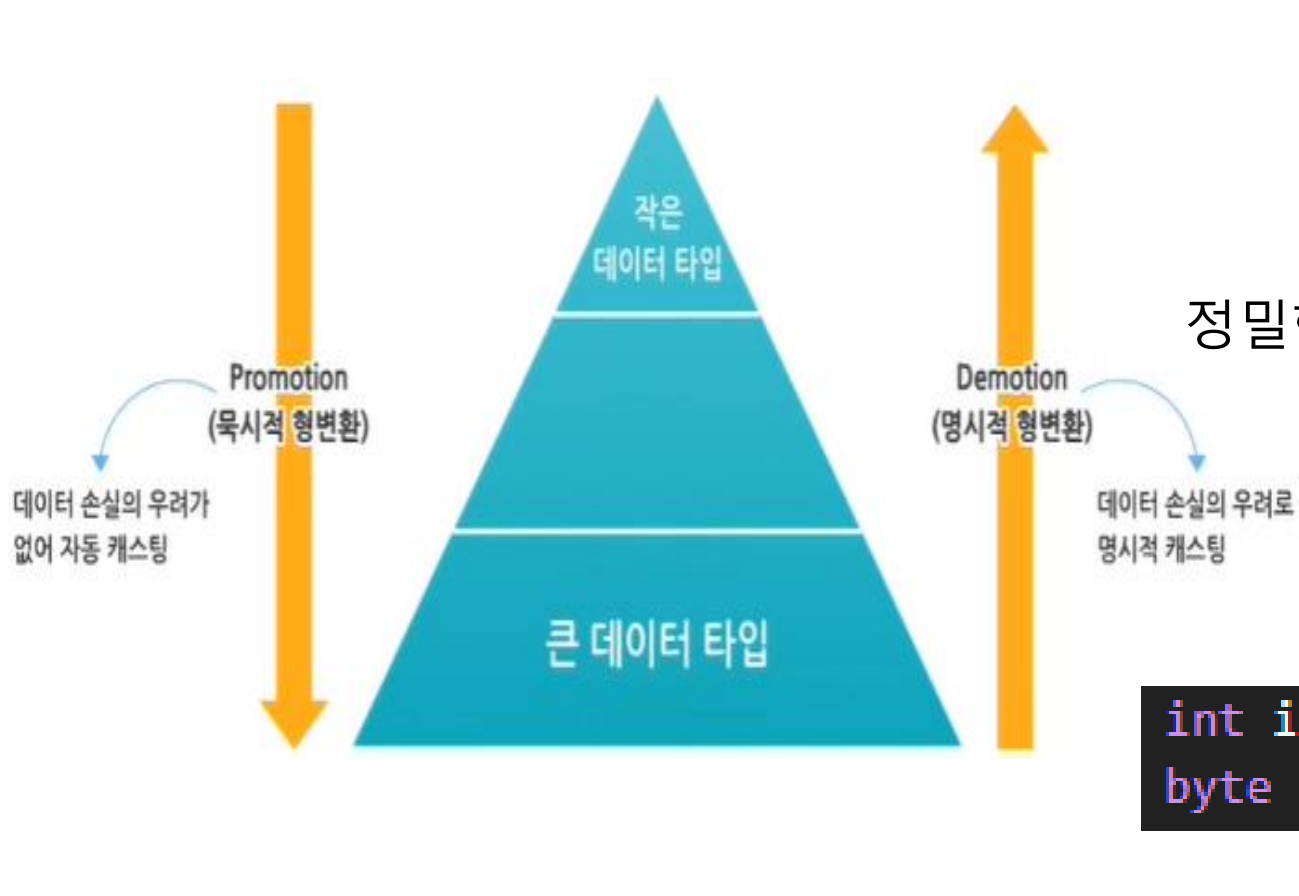


정밀한 숫자나 큰 숫자를 나타내는 정보를 잃을 수 있다

```
int i = 1000;  
byte b = (byte)i;
```

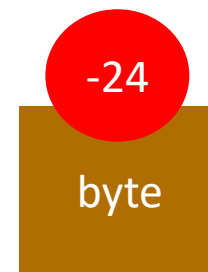


Part 1 명시적 형 변환



정밀한 숫자나 큰 숫자를 나타내는 정보를 잃을 수 있다

```
int i = 1000;  
byte b = (byte)i;
```



Part 2

연산자



Part 2

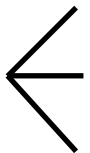
연산자 (Operator) 란 ?

- ▶ 연산자(Operator)
 - 어떠한 기능을 수행하는 기호(+,-,*,/ 등)
- ▶ 피연산자(Operand)
 - 연산자의 작업 대상(변수,상수,리터럴,수식)

$$a + b$$

Part 2

연산자의 종류

- ▶ 단항 연산자 : + - ++ -- ~ ! (타입) ex) -5, ++5
- ▶ 이항 연산자 
 - 산술 : + - * / % << >> >>> ex) 2 + 3, 4 % 2
 - 비교 : > < >= <= == != ex) 1 < 3, 2 != 3
 - 논리 : && || & ^ | ex) true && true, true || false
- ▶ 삼항 연산자 : ? : ex) 1 < 2 ? true : false
- ▶ 대입 연산자 : = ex) a = 1

Part 2

연산자의 우선순위

종 류	연산방향	연 산 자	우선순위
단항 연산자	←	++ -- + - ~ ! (타입)	높음
산술 연산자	→	* / %	
	→	+ -	
	→	<< >> >>>	
비교 연산자	→	< > <= >= instanceof	
	→	== !=	
논리 연산자	→	&	
	→	^	
	→		
	→	&&	
	→		
삼항 연산자	→	?:	
대입 연산자	←	= *= /= %= += -= <<= >>= >>>= &= ^= =	낮음

[표3-1] 연산자의 종류와 우선순위

Part 2

대입 연산자(=)와 산술 연산자(+, -, *, /, %)

연산자	연산자의 기능	결합방향
=	연산자 오른쪽에 있는 값을 연산자 왼쪽에 있는 변수에 대입한다. 예) val = 20;	←
+	두 피연산자의 값을 더한다. 예) val = 4 + 3;	→
-	왼쪽의 피연산자 값에서 오른쪽의 피연산자 값을 뺀다. 예) val = 4 - 3;	→
*	두 피연산자의 값을 곱한다. 예) val = 4 * 3;	→
/	왼쪽의 피연산자 값을 오른쪽의 피연산자 값으로 나눈다. 예) val = 7 / 3;	→
%	왼쪽의 피연산자 값을 오른쪽의 피연산자 값으로 나눴을 때 얼게 되는 나머지를 반환한다. 예) val = 7 % 3	→

실습문제 - 백준 10869번

5 10869번

사칙연산

성공



시간 제한	메모리 제한	제출	정답	맞힌 사람	정답 비율
1 초	256 MB	507848	231158	195481	45.653%

해설

문제

두 자연수 A와 B가 주어진다. 이때, A+B, A-B, A*B, A/B(몫), A%B(나머지)를 출력하는 프로그램을 작성하시오.

입력

두 자연수 A와 B가 주어진다. ($1 \leq A, B \leq 10,000$)

출력

첫째 줄에 A+B, 둘째 줄에 A-B, 셋째 줄에 A*B, 넷째 줄에 A/B, 다섯째 줄에 A%B를 출력한다.

예제 입력 1 복사

7 3

예제 출력 1 복사

10
4
21
2
1

```
1 import java.util.Scanner;
2
3 public class Main {
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6
7         int a = sc.nextInt();
8         int b = sc.nextInt();
9
10        System.out.println(a+b);
11        System.out.println(a-b);
12        System.out.println(a*b);
13        System.out.println(a/b);
14        System.out.println(a%b);
15
16        sc.close();
17    }
18 }
```

Part 2

복 합 대 입 연 산 자

$a = a + b$	⇨ 동일 연산 ⇨	$a += b$
$a = a - b$	⇨ 동일 연산 ⇨	$a -= b$
$a = a * b$	⇨ 동일 연산 ⇨	$a *= b$
$a = a / b$	⇨ 동일 연산 ⇨	$a /= b$
$a = a \% b$	⇨ 동일 연산 ⇨	$a \% = b$

해석의 원칙은 동일

$\&=$, $\wedge=$, $|=$, $\ll=$, $\gg=$, $\gg\gg=$

Part 2

관계 연산자

연산자	연산자의 기능	결합방향
<	예) $n1 < n2$ n1이 n2보다 작은가?	→
>	예) $n1 > n2$ n1이 n2보다 큰가?	→
<=	예) $n1 <= n2$ n1이 n2보다 같거나 작은가?	→
>=	예) $n1 >= n2$ n1이 n2보다 같거나 큰가?	→
==	예) $n1 == n2$ n1과 n2가 같은가?	→
!=	예) $n1 != n2$ n1과 n2가 다른가?	→

Part 2

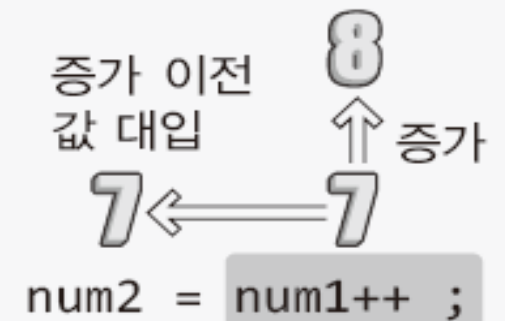
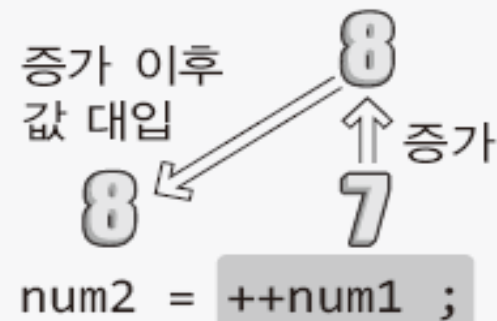
논리 연산자

연산자	연산자의 기능	결합방향
&&	예) A && B A와 B 모두 true이면 연산결과는 true (논리 AND)	→
	예) A B A와 B 둘 중 하나라도 true이면 연산결과는 true (논리 OR)	→
!	예) !A 연산결과는 A가 true이면 false, A가 false이면 true (논리 NOT)	←

Part 2

증가, 감소 연산자

연산자	연산자의 기능	결합방향
++ (prefix)	피연산자에 저장된 값을 1 증가 예) val = ++n;	←
-- (prefix)	피연산자에 저장된 값을 1 감소 예) val = --n;	←
연산자	연산자의 기능	결합방향
++ (postfix)	피연산자에 저장된 값을 1 증가 예) val = n++;	←
-- (postfix)	피연산자에 저장된 값을 1 감소 예) val = n--;	←

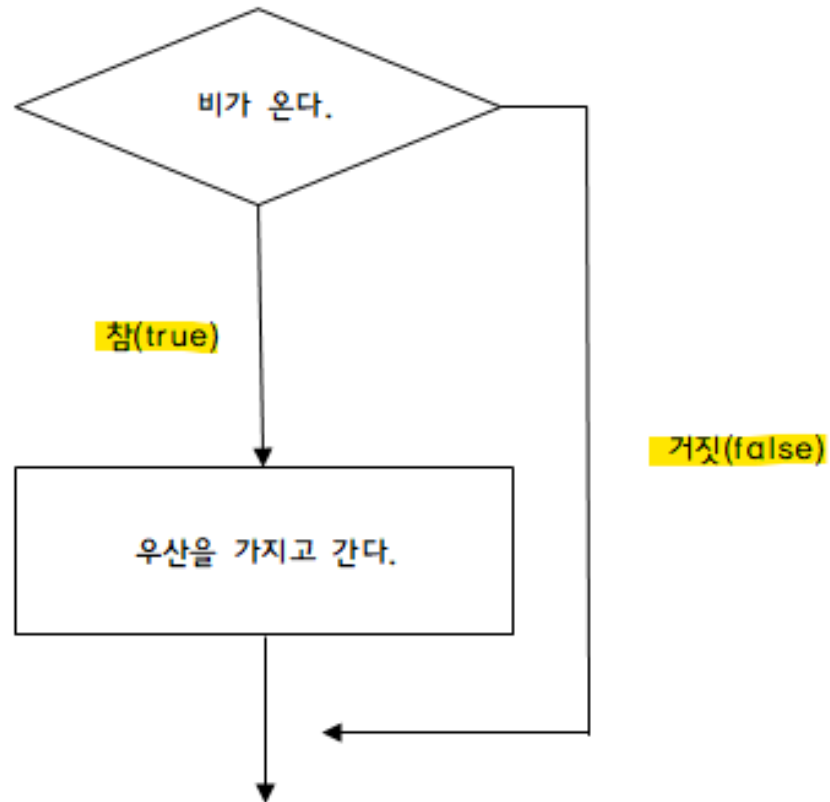


Part 3

조건문



Part 3 조건문 ?



조건을 판단해서 그 상황에 맞게 처리해야 할 경우

Part 3

조건문의 종류 - if, switch

```
if(조건식) {  
    문장들;  
}
```

```
switch(변수 또는 조건식) {  
    case 상수1 :  
        문장들;  
        break;  
    case 상수2 :  
        문장들;  
        break;  
    default :  
        문장들;  
        break;  
}
```

Part 4

if문



Part 4 if 문의 기본 구조 1 (if)

```
if (조건식) {  
    수행될 문장;  
}
```

```
if (A) {    만약 A라면?  
    B;    B 해라  
}
```

Part 4

if 문의 기본 구조 2 (if-else)

```
if (조건식) {  
    수행될 문장;  
}  
else {  
    수행될 문장;  
}
```

```
if (A) { 만약 A라면?  
    B; B 해라  
}  
else { A가 아니라면?  
    C; C 해라  
}
```


Part 4

if 문의 기본 구조 3 (if-else if)

```
if (조건식) {  
    수행될 문장;  
}  
else if (조건식) {  
    수행될 문장;  
}  
else {  
    수행될 문장;  
}
```

```
if (A1) { 만약 A1이라면?  
    B; B 해라  
}  
else if (A2) { A1이 아닌 A2라면?  
    C; C 해라  
}  
else { A1도, A2도 아니라면?  
    D; D 해라  
}
```

else if 여러 개 사용 가능

Part 4

if 문의 기본 구조 4 (중첩 if 문)

```
if (조건식) {  
    수행될 문장;  
    if (조건식) {  
        수행될 문장;  
    }  
}
```

```
if (A) { 만약 A라면?  
    B; B 해라  
    if (A1) { A면서 A1이라면?  
        C; C 해라  
    }  
}
```


Part 4

삼항 연산자

Part 2

연산자의 종류

▶ 단항 연산자 : + - (타입) ++ -- ~ !

▶ 이항 연산자  산술 : + - * / % << >> >>>
비교 : > < >= <= == !=
논리 : && || & ^ |

▶ 삼항 연산자 : ? :

▶ 대입 연산자 : =

Part 4

삼항 연산자

- 조건식의 연산결과가 true이면 '식1'의 결과를 반환하고
false이면 '식2'의 결과를 반환한다.

(조건식) ? 식1 : 식2

```
int x = -10;  
int y;  
if (x >= 0) {  
    y = x;  
} else {  
    y = -x;  
}
```

```
int x = -10;  
int y;  
y = (x >= 0) ? x : -x;
```

y의 값은?
y == 10

실습문제 - 백준 1330번

5 1330번

두 수 비교하기

성공



시간 제한

메모리 제한

제출

정답

맞힌 사람

정답 비율

1 초

512 MB

433187

209609

172264

49.706%

해설

문제

두 정수 A와 B가 주어졌을 때, A와 B를 비교하는 프로그램을 작성하시오.

입력

첫째 줄에 A와 B가 주어진다. A와 B는 공백 한 칸으로 구분되어져 있다.

출력

첫째 줄에 다음 세 가지 중 하나를 출력한다.

- A가 B보다 큰 경우에는 '>'를 출력한다.
- A가 B보다 작은 경우에는 '<'를 출력한다.
- A와 B가 같은 경우에는 '=='를 출력한다.

제한

- $-10,000 \leq A, B \leq 10,000$

예제 입력 1 복사

1 2

예제 출력 1 복사

<

예제 입력 2 복사

10 2

예제 출력 2 복사

>

예제 입력 3 복사

5 5

예제 출력 3 복사

==

```
1 import java.util.Scanner;
2
3 public class Main {
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6
7         int a = sc.nextInt();
8         int b = sc.nextInt();
9
10        if (a > b) {
11            System.out.println(">");
12        } else if (a < b) {
13            System.out.println("<");
14        } else {
15            System.out.println("==");
16        }
17
18        sc.close();
19    }
20 }
```

Q&A



감사합니다