

Operator - Assignment, Arithmetic

! Return (반환)

- 연산문, 메소드가 있던 자리를 연산의 결과값이나 메소드가 return하는 값으로 대체하는것

```
int num1 = 5+2;  
// int num1 = 7; // 5+2 연산문이 연산결과 7로 대체되었다.
```

```
int sum(int n1, int n2){  
    return n1+n2;  
}
```

```
int num2 = sum(10,20);  
// int num2 = 30; // sum 메소드가 return값 30으로 대체되었다.
```

연산자 우선순위와 결합방향

- 우선순위 : Java의 연산자들 사이에는 우선순위가 있다. (우선순위 Table 참고)
[], ()안의 연산이 우선순위가 제일 높다.

- 결합방향

→	왼쪽에서 오른쪽으로 실행
←	오른쪽에서 왼쪽으로 실행

Assignment, Arithmetic Operator

이항연산자

=	연산자의 오른쪽에 있는 값을 왼쪽의 변수에 저장한다.	←
+	두 피연산자 값을 더한다.	→
-	왼쪽의 피연산자 값에서 오른쪽 피연산자 값을 뺀다.	→
*	두 피연산자의 값을 곱한다.	→
/	왼쪽의 피연산자 값을 오른쪽 피연산자의 값으로 나누고 몫을 반환한다.	→
%	왼쪽의 피연산자 값을 오른쪽 피연산자의 값으로 나누고 나머지를 반환한다.	→

- 나누기 연산시 주의할 점
 - $5 \div 2 =$ 몫 2, 나머지 1 (정수 ÷ 정수이면 정수형 연산, 결과도 정수)
 - $5.0 \div 2.0 =$ 몫 2.5 (실수 ÷ 실수이면 실수형 연산, 몫만 실수로 반환)

복합대입연산자

+=	A += B	A = A + B
-=	A -= B	A = A - B
*=	A *= B	A = A * B
/=	A /= B	A = A / B
%=	A %= B	A = A % B
