Topic 05: Develop Code

Java Operators

Lecture Contents

- Java Operator Types
  - Arithmetic Operators
  - Assignment Operators
  - Increment/Decrement Operators
  - Comparison Operators
  - Logical Operators
- Operator Precedence

Operator Types

- ***Arithmetic***
- ***Assignment***
- ***Comparison***
- ***Logical***
- ***Bitwise***

Operator Types

Arithmetic			**+**	**–**	*****	**/**	**%**
Assignment	=		+=	–=	*=	/=	%=
Increment			++	– –			
Comparison	!=	==	>	>=	<	<=	
Logical	!		||		&&		
Bitwise	~ ~=		| |=		& &=		^ ^=

Java Arithmetic Operators

- We should know the word "***operator***" from mathematics
- Java ***arithmetic operators***:
  - Add: `x + y`
  - Subtract: `x - y`
  - Multiply: `x * y`
  - Divide: `x / y`
  - Modulus: `x % y`

Java Arithmetic Operators

- We used arithmetic operators when we examined primitive types

```
System.out.println(3 + 5);
System.out.println(3.0 / 5.0);
System.out.println(3.0 + 5);
System.out.println(3 / 5);
```

```
8
0.6
8.0
0
```

Operator Types

✓ *Arithmetic*			+	–	*	/	%
Assignment	**=**		**+=**	**–=**	***=**	**/=**	**%=**
Increment			++	– –			
Comparison	!=	==	>	>=	<	<=	
Logical	!		||		&&		
Bitwise	~ ~=		| |=		& &=		^ ^=

Java Assignment Operators

- The ***assignment*** *operator* is used to set the value of a variable.

Java Assignment Operators

- An ***assignment operator*** is used to set the value of a variable.
- We used an *assignment operator* when we ***initialized*** variables:

```
public static void main(String[] args) {
    int myNumber = 13;
    System.out.println(myNumber);
}

public static void main(String[] args) {
    int myNumber;
    MyNumber = 13;
    System.out.println(myNumber);
}
```

Java Assignment Operators

- The ***assignment operator*** is used to set the value of a variable.
- We used the *assignment operator* when we ***assigned*** a new value to a variable after initializing it:

```
public static void main(String[] args) {
    int myNumber = 13;
    myNumber = 7;
    System.out.println(myNumber);
}
```

```
7
```

Java Assignment Operators

- The ***assignment operator*** is used to set the value of a variable.
- We used the ***assignment operator*** when we *assigned* a calculated value to a variable:

```
public static void main(String[] args) {
    int myNumber = 13;
    myNumber = myNumber + 7;
    System.out.println(myNumber);
}
```

Java Assignment Operators

- The ***assignment operator*** is used to set the value of a variable.
- We used the ***assignment operator*** when we *assigned* a calculated value to a variable:
  - Note the difference between how we would treat the expression in mathematics!

```
public static void main(String[] args) {
    int myNumber = 13;
    myNumber = myNumber + 7;
    System.out.println(myNumber);
}
```

```
20
```

Java Assignment Operators

- We have combined *arithmetic-assignment* operators for all the arithmetic operators: `+=` `-=` `*=` `/=` `%=`

```
public static void main(String[] args) {
    int myNumber = 13;
    myNumber *= 3;
    System.out.println(myNumber);
}
```

```
39
```

Java Assignment Operators

- Assignment operators that combine ***arithmetic*** operations:

	Arithmetic Operation and Assignment	Combined Operation
Addition	`x = x + 5`	`x += 5`
Subtraction	`x = x - 7`	`x -= 7`
Multiplication	`x = x * 3`	`x *= 3`
Division	`x = x / 6`	`x /= 6`
Modulus	`x = x % 4`	`x %= 4`

- Note: There are also ***bitwise assignment*** operators.

Operator Precedence

Level	Operators	Associativity
16	( )	Left-to-right
15	++, --	Left-to-right
12	*, /, %	Left-to-right
11	+, -	Left-to-right
1	=, +=, -= *=, /=, %=	Right-to-Left

Operator Precedence

```
public static void main(String[] args) {
	int v = 10;
	int w = 20;
	System.out.println(w + v / w);
}
```

Operator Precedence

```
public static void main(String[] args) {
	int v = 10;
	int w = 20;
	System.out.println(w + v / w);
}
```

```
20
```

Operator Precedence

```
public static void main(String[] args) {
	int a = 1;
	int b = 2;
	int c = 3;
	int d = 4;
	int x = a + b * c % d;
	System.out.println(x);
}
```

Operator Precedence

```
public static void main(String[] args) {
	int a = 1;
	int b = 2;
	int c = 3;
	int d = 4;
	int x = a + b * c % d;
	System.out.println(x);
}
```

```
3
```

Operator Precedence

```
public static void main(String[] args) {
	System.out.println(2 / 5 % 3);
	System.out.println(2 / (5 % 3));
	System.out.println(2 / 5 + 1);
}
```

Operator Precedence

```
public static void main(String[] args) {
	System.out.println(2 / 5 % 3);
	System.out.println(2 / (5 % 3));
	System.out.println(2 / 5 + 1);
}
```

```
0
1
1
```

Operator Types

✓ *Arithmetic*			+	–	*	/	%
✓ *Assignment*	=		+=	–=	*=	/=	%=
➔ ***Increment***			**++**	**– –**			
Comparison	!=	==	>	>=	<	<=	
Logical	!		||		&&		
Bitwise	~ ~=		| |=		& &=		^ ^=

Operator Precedence

```
public static void main(String[] args) {
	int i = 5;
	System.out.println(i);
	System.out.println(i++);
	System.out.println(i);
	System.out.println(++i);
}
```

Operator Precedence

```
public static void main(String[] args) {
	int i = 5;
	System.out.println(i);
	System.out.println(i++);
	System.out.println(i);
	System.out.println(++i);
}
```

```
5
5
6
7
```

Operator Precedence

```
public static void main(String[] args) {
	int i = 5;
	i ++;
	System.out.println(i);
}
```

```
6
```

Operator Types

✓ *Arithmetic*			+	–	*	/	%
✓ *Assignment*	=		+=	–=	*=	/=	%=
✓ *Increment*			++	– –			
Comparison	**!=**	**==**	**>**	**>=**	**<**	**<=**	
Logical	!		||		&&		
Bitwise	~ ~=		| |=		& &=		^ ^=

Comparison operators return either `true` or `false`.

As a general rule, only use ***comparison operators*** with ***primitive types*** (`int`, `double`, `boolean`)

Comparison Operators

Symbol	Name
==	equal to
!=	not equal to
>	greater than
>=	greater than or equal to
<	less than
<=	less than or equal to

Operator Precedence

Level	Description	Operators	Associativity
16	parentheses	( )	Left-to-right
15	post inc/dec	++, --	Left-to-right
13	cast	()	Right-to-Left
12	multiplicative	*, /, %	Left-to-right
11	additive	+, -	Left-to-right
9	**relational**	**>, >=, <, <=**	Left-to-right
8	**equality**	**==, !=**	Left-to-right
1	assignment	=, +=, -= *=, /=, %=	Right-to-Left

Comparison Operators

```
int myInt = 6;
boolean myBool = myInt >= 5;
System.out.println("myBool = " + myBool);
```

Comparison Operators

```
int myInt = 6;
boolean myBool = myInt >= 5;
System.out.println("myBool = " + myBool);
```

```
myBool = true
```

Comparison Operators

```
int myInt = 6;
boolean myBool = myInt <= 5;
System.out.println("myBool = " + myBool);
```

Comparison Operators

```
int myInt = 6;
boolean myBool = myInt <= 5;
System.out.println("myBool = " + myBool);
```

```
myBool = false
```

Comparison Operators

```
int myInt = 6;
boolean myBool = myInt == 6.0;
System.out.println("myBool = " + myBool);
```

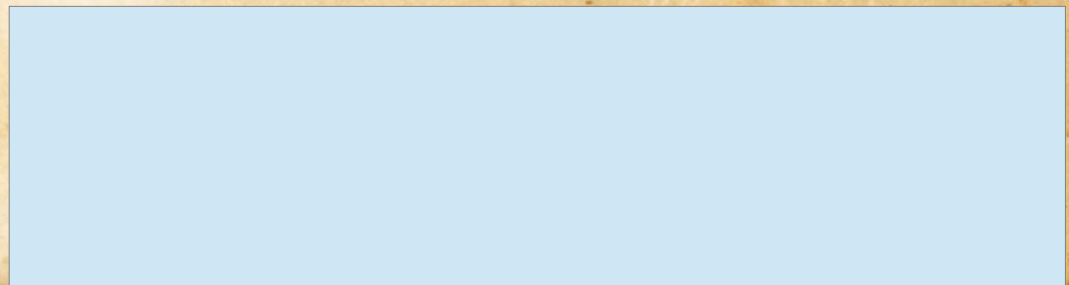
Comparison Operators

```
int myInt = 6;
boolean myBool = myInt == 6.0;
System.out.println("myBool = " + myBool);
```

```
myBool = true
```

The `if` Statement

```
public static void main(String args[]) {
   int x = 5
   System.out.println("Starting Program...");
   if(x == 5) {
      System.out.println("Hello World!");
   }
   System.out.println("Finished Program!");
}
```



The `if` Statement

```
public static void main(String args[]) {
   int x = 5
   System.out.println("Starting Program...");
   if(x == 5) {  if(true) {
      System.out.println("Hello World!");
   }
   System.out.println("Finished Program!");
}
```

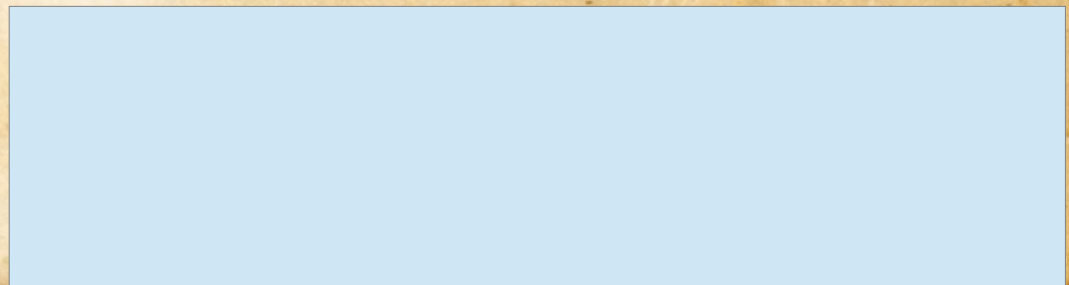
The `if` Statement

```
public static void main(String args[]) {
   int x = 5
   System.out.println("Starting Program...");
   if(x == 5) {  if(true) {
      System.out.println("Hello World!");
   }
   System.out.println("Finished Program!");
}
```

```
Starting Program...
Hello World!
Finished Program!
```

The `if` Statement

```
public static void main(String args[]) {
   int x = 5
   System.out.println("Starting Program...");
   if(x > 5) {
      System.out.println("Hello World!");
   }
   System.out.println("Finished Program!");
}
```



The `if` Statement

```
public static void main(String args[]) {
   int x = 5
   System.out.println("Starting Program...");
   if(x > 5) {  if(false) {
      System.out.println("Hello World!");
   }
   System.out.println("Finished Program!");
}
```

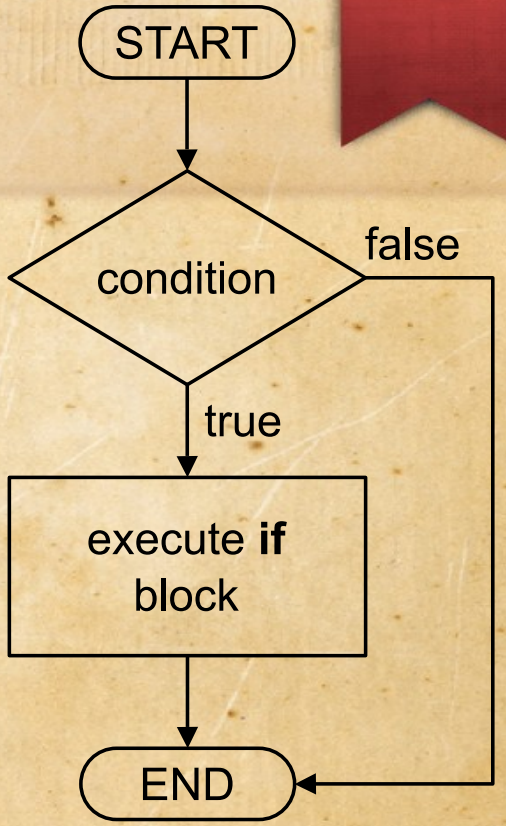
The `if` Statement

```
public static void main(String args[]) {
   int x = 5
   System.out.println("Starting Program...");
   if(x > 5) {  if(false) {
      System.out.println("Hello World!");
   }
   System.out.println("Finished Program!");
}
```

```
Starting Program...
Finished Program!
```

The `if` Statement

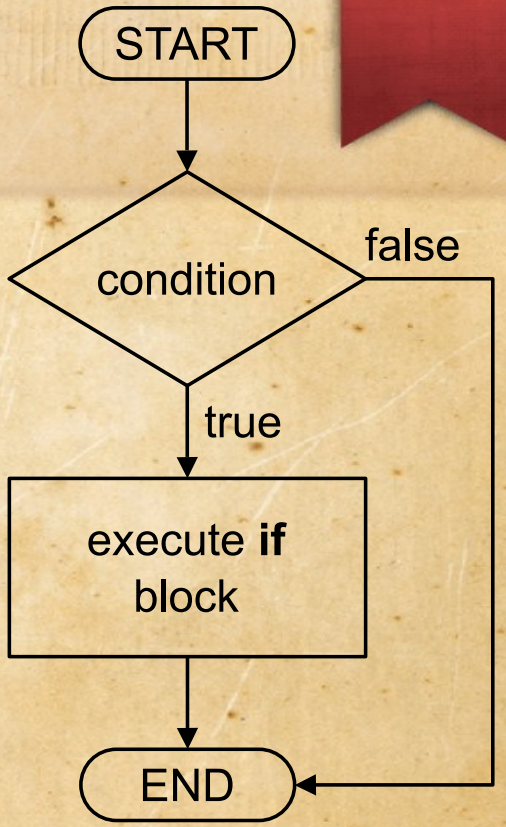
```
public static void main(String args[]) {
   int x = 5
   System.out.println("Starting Program...");
   if(x > 5) {  if(false) {
      System.out.println("Hello World!");
   }
   System.out.println("Finished Program!");
}
```




```
Starting Program...
Finished Program!
```

The `if` Statement

```
public static void main(String args[]) {
   int x = 5
   System.out.println("Starting Program...");
   if(x > 5) {  if(false) {
      System.out.println("Hello World!");
   }
   System.out.println("Finished Program!");
}
```



Note: a ***condition*** is an expression that evaluates to either **`true`** or **`false`**.

```
Starting Program...
Finished Program!
```

The `if..else..` Statement

```
public static void main(String args[]) {
   System.out.println("Starting Program...");
   if(false) {
      System.out.println("Hello World!");
   } else {
      System.out.println(“Goodbye cruel world…”);
   }
   System.out.println("Finished Program!");
}
```

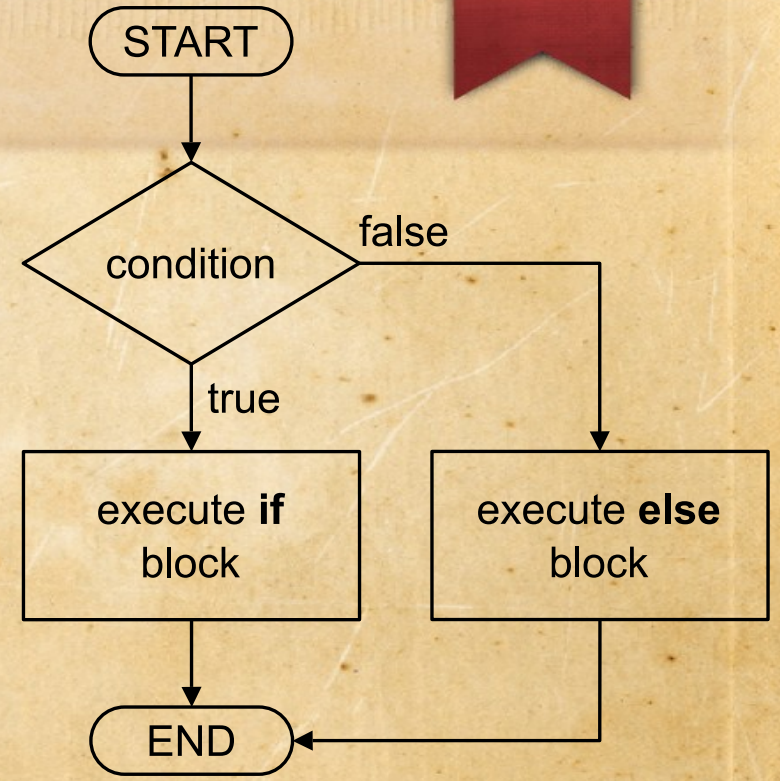
The `if..else..` Statement

```
public static void main(String args[]) {
   System.out.println("Starting Program...");
   if(false) {
      System.out.println("Hello World!");
   } else {
      System.out.println(“Goodbye cruel world…”);
   }
   System.out.println("Finished Program!");
}
```

```
Starting Program…
Goodbye cruel world...
Finished Program!
```

The `if..else..` Statement

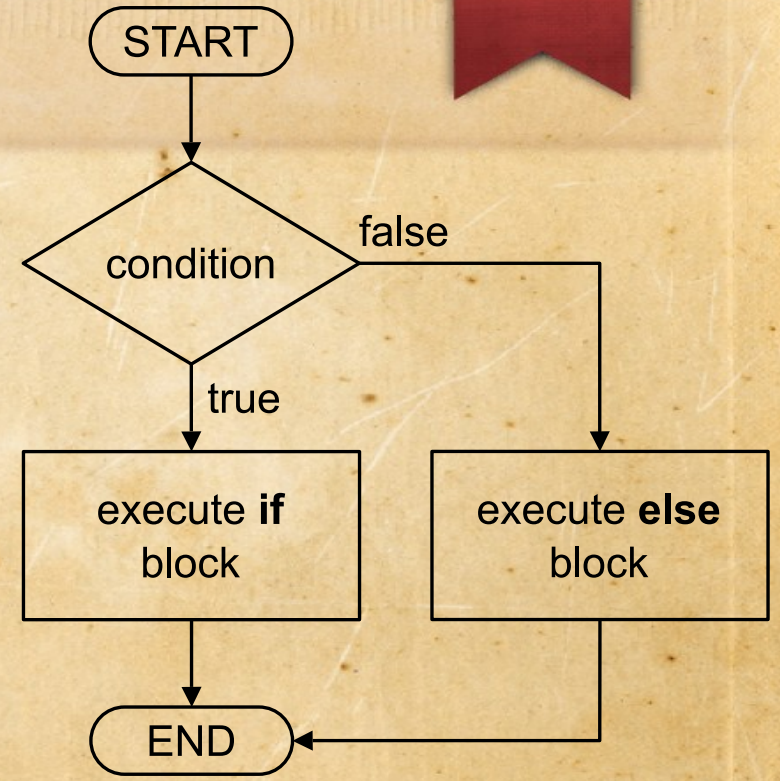
```
public static void main(String args[]) {
   System.out.println("Starting Program...");
   if(false) {
      System.out.println("Hello World!");
   } else {
      System.out.println(“Goodbye cruel world…”);
   }
   System.out.println("Finished Program!");
}
```




```
Starting Program…
Goodbye cruel world...
Finished Program!
```

The `if..else..` Statement

```
public static void main(String args[]) {
   System.out.println("Starting Program...");
   if(false) {
      System.out.println("Hello World!");
   } else {
      System.out.println(“Goodbye cruel world…”);
   }
   System.out.println("Finished Program!");
}
```



Note: a ***condition*** is an expression that evaluates to either **`true`** or **`false`**.

```
Starting Program…
Goodbye cruel world...
Finished Program!
```

Operator Types

✓ *Arithmetic*			+	–	*	/	%
✓ *Assignment*	=		+=	–=	*=	/=	%=
✓ *Increment*			++	– –			
Comparison	**!=**	**==**	**>**	**>=**	**<**	**<=**	
Logical	!		||		&&		
Bitwise	~ ~=		| |=		& &=		^ ^=

Comparison operators return either `true` or `false`.

Comparison Operators

```
public static void main(String args[]) {
   System.out.println("Starting Program...");
   if (5 == 5.0) {
      System.out.println("Hello World!");
   }else {
      System.out.println("Goodbye cruel world...");
   }
   System.out.println("Finished Program!");
}
```

Comparison Operators

```
public static void main(String args[]) {
   System.out.println("Starting Program...");
   if (5 == 5.0) {
      System.out.println("Hello World!");
   }else {
      System.out.println("Goodbye cruel world...");
   }
   System.out.println("Finished Program!");
}
```

```
Starting Program...
Hello World!
Finished Program!
```

Comparison Operators

```
public static void main(String args[]) {
   System.out.println("Starting Program...");
   if (5 == "5") {
      System.out.println("Hello World!");
   }else {
      System.out.println("Goodbye cruel world...");
   }
   System.out.println("Finished Program!");
}
```

Comparison Operators

```
public static void main(String args[]) {
   System.out.println("Starting Program...");
   if (5 == "5") {
      System.out.println("Hello World!");
   }else {
      System.out.println("Goodbye cruel world...");
   }
   System.out.println("Finished Program!");
}
```

```
ERROR:
Incompatible operand types int and
String
```

Comparison Operators

```
public static void main(String args[]) {
   System.out.println("Starting Program...");
   if (65 == 'A') {
      System.out.println("Hello World!");
   }else {
      System.out.println("Goodbye cruel world...");
   }
   System.out.println("Finished Program!");
}
```

Comparison Operators

```
public static void main(String args[]) {
   System.out.println("Starting Program...");
   if (65 == 'A') {
      System.out.println("Hello World!");
   }else {
      System.out.println("Goodbye cruel world...");
   }
   System.out.println("Finished Program!");
}
```

```
Starting Program...
Hello World!
Finished Program!
```

Comparison Operators – Strings

```
public static void main(String args[]) {
   if ("Hello" == "Hello") {
      System.out.println("Hello == Hello");
   }
}
```

Comparison Operators – Strings

```
public static void main(String args[]) {
   if ("Hello" == "Hello") {
      System.out.println("Hello == Hello");
   }
}
```

```
Hello == Hello
```

Comparison Operators – Strings

```
public static void main(String args[]) {
   String s = "Hell";
   s += "o";
   System.out.println("s == " + s);
   if (s == "Hello") {
      System.out.println("Confirmed. s == Hello");
   } else {
      System.out.println("What?! s != Hello");
   }
}
```

Comparison Operators – Strings

```
public static void main(String args[]) {
   String s = "Hell";
   s += "o";
   System.out.println("s == " + s);
   if (s == "Hello") {
      System.out.println("Confirmed. s == Hello");
   } else {
      System.out.println("What?! s != Hello");
   }
}
```

```
s == Hello
What?! s != Hello
```

Comparison Operators – Strings

```
public static void main(String args[]) {
   String s = "Hell";
   s += "o";
   System.out.println("s == " + s);
   if (s == "Hello") {
      System.out.println("Confirmed. s == Hello");
   } else {
      System.out.println("What?! s != Hello");
   }
}
```

Note: to be safe simply do ***NOT*** compare the values stored in objects using the equality operator! (==)

```
s == Hello
What?! s != Hello
```

Operator Types

✓ *Arithmetic*			+	–	*	/	%
✓ *Assignment*	=		+=	–=	*=	/=	%=
✓ *Increment*			++	– –			
✓ *Comparison*	!=	==	>	>=	<	<=	
Logical	!		||		&&		
Bitwise	~ ~=		| |=		& &=		^ ^=

Logical Operators

!	**not**
&&	**and**
||	**or**

- Note the ***logical operators*** require double && and double **||** . Single & and single **|** are used as ***bitwise operators***

Truth Tables

Input	!
0	**1**
1	**0**

Inputs		&&
0	0	**0**
0	1	**0**
1	0	**0**
1	1	**1**

Inputs		||
0	0	**0**
0	1	**1**
1	0	**1**
1	1	**1**

Logical Operators

- Logical operators combine `boolean` values

```
if( x > 0 && x % 2 == 0) {
    System.out.println("x is both positive and even.");
}

if( x > 0 || x % 2 == 0) {
    System.out.println("x is either positive or it is even");
    System.out.println("  ... or maybe it is both.");
}

if( !(x % 2 == 0) ) {
    System.out.println("Saying 'x is even' is not true!");
}
```

Operator Types

✓ *Arithmetic*			+	–	*	/	%
✓ *Assignment*	=		+=	–=	*=	/=	%=
✓ *Increment*			++	– –			
✓ *Comparison*	!=	==	>	>=	<	<=	
✓ *Logical*	!		||		&&		
Bitwise	~ ~=		| |=		& &=		^ ^=

Operator Precedence

Level	Description	Operators	Associativity
16	parentheses	( )	Left-to-right
15	post inc/dec	++, --	Left-to-right
14	**logical NOT**	**!**	Right-to-Left
13	cast	()	Right-to-Left
12	multiplicative	*, /, %	Left-to-right
11	additive	+, -	Left-to-right
9	relational	>, >=, <, <=	Left-to-right
8	equality	==, !=	Left-to-right
4	**logical AND**	**&&**	Left-to-right
3	**logical OR**	**||**	Left-to-right
1	assignment	=, +=, -= *=, /=, %=	Right-to-Left

Topic 05: Develop Code

Java Operators