

# Introduction to Java

## Notes Packet #2

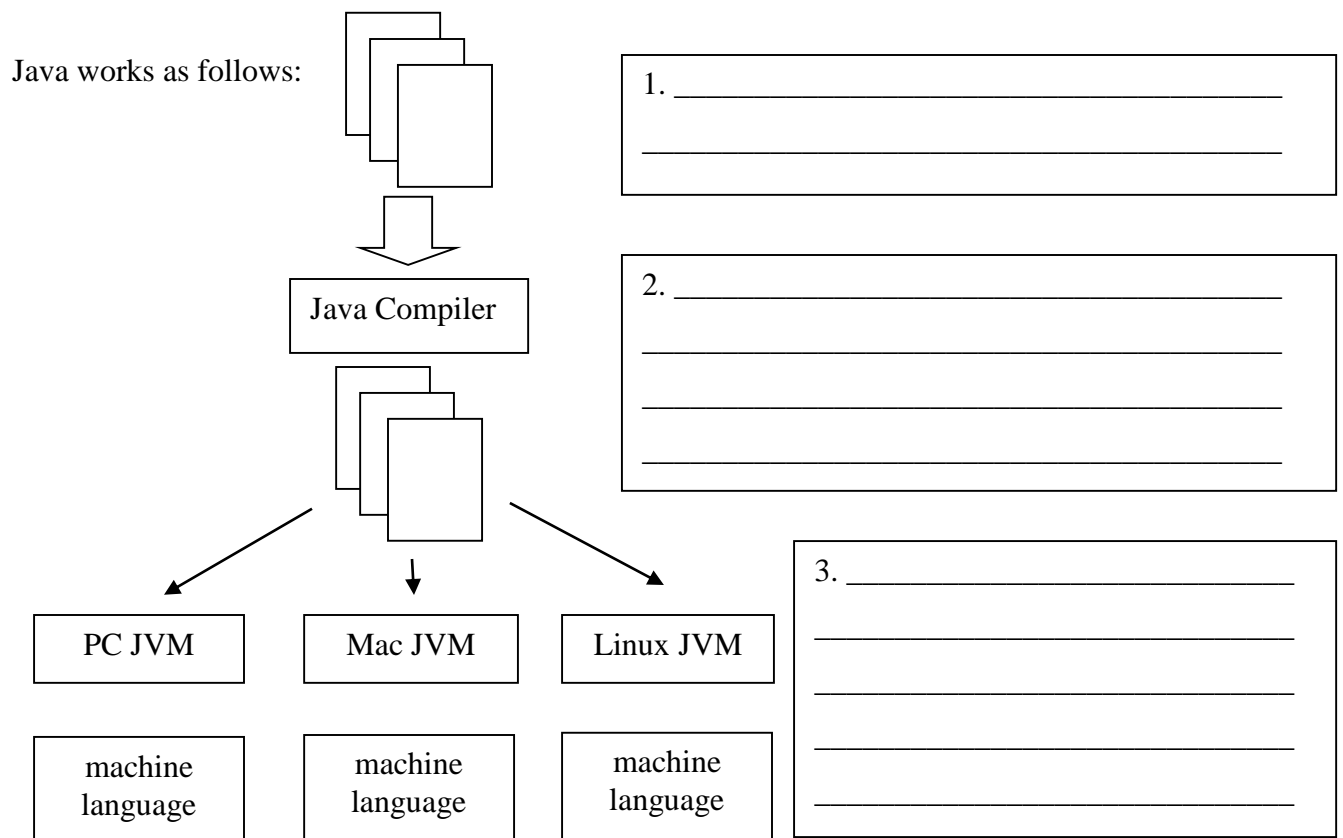
Name: \_\_\_\_\_

**Objectives:** By the completion of this packet, students should be able to

- describe the difference between .java and .class files and the JVM.
- create and use flow charts
- evaluate and use Boolean expressions
- create and use if and if-else statements
- use the && (AND) and || (OR) operators
- use some methods of the Math class.

### Compiling Code in Java

A compiler is a program that translates code to machine language that the central processing unit (the computer's "brain") can understand. In many languages, the source code must be customized for different operating systems and then compiled for certain types of processors.



---

---

---

## Boolean Expressions

A Boolean expression is an expression that is either \_\_\_\_\_ or \_\_\_\_\_.

For example:  $x \geq 8$

The above expression compares  $x$  to 8. If  $x$  is greater than or equal to 8, then the expression is true. Otherwise the expression is false.

Java uses 6 relational operators in its Boolean expressions.

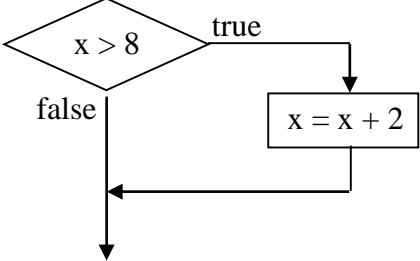
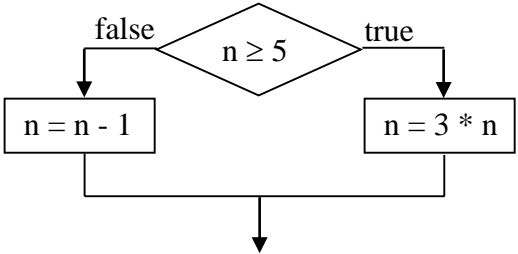
1. If $x$ is an int with a value of 12, what is the value of this expression?	$x > 10$
2. If $num$ is an int with a value of 22, what is the value of this expression?	$num == 14$
3. If $y$ is an int with a value of 7, what is the value of this expression?	$50 \leq y$

<p>Boolean expressions can be manipulated the same way inequalities in math can be manipulated to produce equivalent expressions.</p> <p>Rewrite the expression so that all <math>x</math> terms are combined on the left and the coefficient is 1.</p>	$x > 20 + 5 * x$
---	------------------

4. Are the two expressions equivalent?	$num \leq k$	$k \leq num$
5. Are the two expressions equivalent?	$h \neq 20$	$20 \neq h$
6. Are the two expressions equivalent?	$20 - y > 5$	$y < 15$

## The if and if-else Statements

These statements are sometimes called control statements because they control the flow of the program. Some texts refer to them as conditionals because they represent a condition that may be true or false.

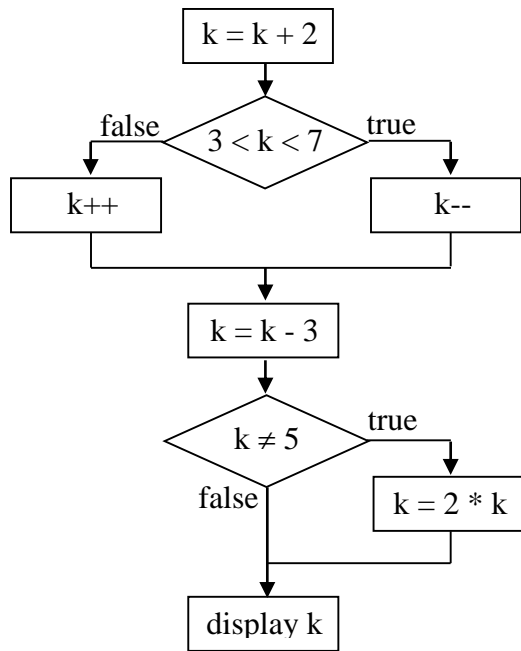
Flowchart	Java Code
	<hr/> <hr/> <hr/>
	<hr/> <hr/> <hr/> <hr/>

**Logical Operators.** More complicated Boolean expressions can be composed through the use of logical operators such as AND and OR.

The following table shows how certain expressions can be written.

English	Mathematically	Java
If the number is greater than 0 and less than or equal to 5.		
If the number is less than 12 or greater than 65		

Example. Complete the code to the right based on the flowchart below.



```

import java.util.Scanner;

public class Example {
    public static void main(String[] args) {
        Scanner x = new Scanner( System.in );
        System.out.print("Enter an integer ");
        int k = x.nextInt();
        _____
        _____
        _____
        _____
        _____
        _____
        _____
        _____
        _____
        _____
    }
}
  
```

### The Math Class.

The Math Class contains a collection of standard math functions. Here are some examples on how to use them.

To take the square root of a number:

```

double a = Math.sqrt( 16 );           // a equals _____
a = Math.sqrt( a );                  // a equals _____
  
```

To raise a number to nth power:

```

double b = Math.pow( 2, 3 );          // equivalent to _____, b equals _____
double c = Math.pow( b, 2.0 );        // c equals _____
  
```

Note: to square a number you can always multiply it by itself.

```

int e = 7;
int f = e * e;                       // or int f = (int) Math.pow( e, 2 );
  
```

Find the absolute value:

```
int x = Math.abs( -5 );           // if the argument is an int, the method returns an int
double y = Math.abs( -7.3 );     // if the argument is a double, the method returns a double
```

To generate a random decimal between 0 (inclusive) and 1 (exclusive):

```
double g = Math.random();        // _____
double h = 5 * Math.random();    // _____
```

To generate a random integer from min (inclusive) and max (inclusive), use the following formula:

```
int num = (int) ( range * Math.random() ) + min; // where range = max - min + 1
```

For example, to generate a random integer between 1 and 6:

```
int num = (int) ( 6 * Math.random() ) + 1; // num is equal to 1, 2, 3, 4, 5, or 6
```

1. <i>a</i> is a random decimal within these limits _____ ≤ <i>a</i> < _____	double a = 10 * Math.random() - 3;
2. <i>b</i> is a random integer within these limits _____ ≤ <i>b</i> ≤ _____	int b = (int)( 5 * Math.random() ) + 3;
3. <i>c</i> is a random integer within these limits _____ ≤ <i>c</i> ≤ _____	int c = (int)( 12 * Math.random() );
4. <i>d</i> is a random integer within these limits _____ ≤ <i>d</i> ≤ _____	int d = (int)( 3 * Math.random() ) - 7;

**if-else if Statements.** The basic if - else structure can be expanded to handle multiple conditionals through the use of else-if statements. For example, if you want to determine if a grade is an A, B, C, D, or E, a simple if - else statement won't work.

Flowchart	Java Code
<pre> graph TD     Start(( )) --&gt; D1{0 ≤ age &lt; 13}     D1 -- true --&gt; P1[print "child"]     D1 -- false --&gt; D2{age &lt; 20}     D2 -- true --&gt; P2[print "teen"]     D2 -- false --&gt; D3{age &lt; 120}     D3 -- true --&gt; P3[print "adult"]     D3 -- false --&gt; P4[print "???"]     P1 --&gt; J1(( ))     P2 --&gt; J1     P3 --&gt; J1     P4 --&gt; J1     J1 --&gt; Done[done]         </pre>	<hr/> <hr/> <hr/> <hr/> <hr/> <hr/> <hr/> <hr/> <hr/> <hr/>
<pre> graph TD     Start(( )) --&gt; P1[get cost get id]     P1 --&gt; D1{cost &gt; 100}     D1 -- true --&gt; P2[reduce cost by 10%]     D1 -- false --&gt; D2{id = 13}     D2 -- true --&gt; P3[reduce cost by 5%]     D2 -- false --&gt; P4[print cost]     P2 --&gt; J1(( ))     P3 --&gt; J1     P4 --&gt; J1     J1 --&gt; End(( ))         </pre>	<pre> import java.util.Scanner;  public class Example {     public static void main(String[] args) {         Scanner sc = new Scanner( System.in );         System.out.print("Enter cost ");         double cost = sc.nextDouble();         System.out.print("Enter id ");         int id = sc.nextInt();          }     }         </pre> <hr/> <hr/> <hr/> <hr/> <hr/> <hr/> <hr/> <hr/> <hr/> <hr/>

IMPORTANT. \_\_\_\_\_

---