THE JAVA IF-ELSE STATEMENT

The *if-else* statement introduces a “fork in the road” for the program’s execution path. The fork has two tines representing two completely separate execution paths. Flow of control can proceed down one of these two paths but never both.

### Java If-Else Statement Syntax

```
if ( truth value )
    true path
else
    false path
    statement below
```

```
if ( truth value )
{
    true path
}
else
{
    false path
}
    statement below
```

If you have only one statement in the *true path* then the braces `{ }` are optional; likewise for the *false path*.

### Java If-Else Statement Semantics

As illustrated in the picture, the computer begins execution of the *if-else* statement by (1) evaluating the *truth value*, which can be either *true* or *false*. If *true*, the computer (2)
executes the true path after which it continues with the statement below. If the truth value is false, the computer executes the false path, after which it continues with the statement below.

**Example**
This code illustrates the semantics of the if-else statement.

```java
System.out.print( "Enter temperature and scale (e.g. 75 F): " );
temp = scanner.nextDouble( );
scale = scanner.next( );
if ( scale.equals( "F" ) )  // Fahrenheit
{
   // convert Fahrenheit to Celsius
   f = temp;
   c = (5.0 / 9.0) * (f - 32);
}
else // Celsius
{
   // convert Celsius to Fahrenheit
   c = temp;
   f = 1.8 * c + 32;
}
System.out.println( f + "u00B0F = " + c + "u00B0C" );
```

For user input of 212 F, the computer executes the true path (lines 7 and 8).

For user input of 0 C, the computer executes the false path (lines 13 and 14).

<table>
<thead>
<tr>
<th>Enter temperature (e.g. 75 F):</th>
<th>Enter temperature (e.g. 75 F):</th>
</tr>
</thead>
<tbody>
<tr>
<td>212 F</td>
<td>0 C</td>
</tr>
<tr>
<td>12.0°F = 100.0°C</td>
<td>32.0°F = 0.0°C</td>
</tr>
</tbody>
</table>
Line Breaks, Indention and the { } Delimiters
The Java compiler ignores line breaks and indentation when it interprets the meaning of an \texttt{if-else} statement. It uses the \{ \} delimiters to determine what’s in the true and false paths; you may omit them if the paths contain only one.

\textbf{Examples}
Each of the following code fragments has the exact same behavior.

\begin{verbatim}
if ( x < y )
    smaller = x;
else
    smaller = y;

if ( x < y)
    smaller = x;
else
    smaller = y;

if ( x < y)
    smaller = x;
    else
    smaller = y;

if ( x < y ) { 
    smaller = x;
} 
else {
    smaller = y;
}

if ( x < y )
{ 
    smaller = x;
} 
else
{ 
    smaller = y;
}
\end{verbatim}
Beginner Errors on If-Else Syntax
Like the other control statements, the **if-else** statement doesn’t respond well to extraneous semicolons and missing or misplaced `{ }` delimiters. In the true path, the compiler responds well with complaints.

**Example**
In the code fragment below, the semicolon in line 1 marks the end of an **if** statement. The compiler thinks the **else** has no corresponding **if** and issues the diagnostic:

```
MyApp.java:1: 'else' without 'if'
   else
   ^
1   if ( x < y ) ;
2       smaller = x;
3   else
4       smaller = y;
```

**Example**
In the code fragment below, the compiler takes lines 1 and 2 as a complete **if** statement. The **else** at line 4 has no corresponding **if** and the compiler issues the diagnostic:

```
MyApp.java:4: 'else' without 'if'
   else
   ^
1   if ( rewardsMember )
2       qualifiedPurchases += purchase;
3       shippingCharge = 0.00;
4   else
5       shippingCharge = 6.00;
```
These same errors in the false path, however, are not caught by the compiler. Instead they cause logic errors much like similar errors with the `if` statement.

**Example**
This code fragment always sets `smaller` to `y`.

```java
if ( x < y )
    smaller = x;
else
    smaller = y;
```
### Exercises

1. What is the output of this code segment if the user enters 50? If the user enters 51?

```java
Scanner input = new Scanner( System.in );
double x = input.nextDouble( );
if ( x <= 50 )
    x /= 2.0;
else
    x *= 2.0;
System.out.println( "x = " + x );
```

For each of the **if-else** statements below, circle what’s wrong and explain. None of them is correct.

2. If ( x < y ) {
    smaller = x;
    larger = y;
} Else {
    smaller = y;
    larger = x;
}

3. if x < y {
    smaller = x;
    larger = y;
} else {
    smaller = y;
    larger = x;
}

4. if ( x < y )
    smaller = x;
    larger = y;
else
    smaller = y;
    larger = x;
For each of the **if-else** statements below, circle what’s wrong and explain. None of them is correct.

5. ```java
   if ( x < y ) {
     smaller = x;
     larger = y;
   } else {
     smaller = y;
     larger = x;
   }
``` 

For each of the following write a Java **if-else** statement that performs the calculation.

6. A mail-order catalog charges a $5.00 shipping charge on orders up to $100 and $10.00 on orders over $100.

7. A hotel charges $59 per night for a room but gives senior citizens (i.e. age of 60 or more) a 10% discount.

8. Given variables **hours** and **pay**, calculate an employee’s pay. He or she earns $9 per hour for up to 40 hours of work and $13.50 per hour for hours over 40. For example, an employee working 50 hours receives \(40 \times 9 + 10 \times 13.50 = 495\).

9. Given variables **familySize** and **fee**, calculate a family’s season membership fee to the municipal swimming pool. A season membership costs $90 for a family of 4. Additional family members can be added for a cost of $30 per member. For example, a family of 6 pays \(90 + 2 \times 30 = 150\).