Key point: *The logical operators* **!**, **&&**, **||**, *and* **^** *can be used to create a compound Boolean expression.*

Table 3.3 Boolean Operators

Operator	*Name*	*Description*
!	not	Logical negation
&&	and	Logical conjunction
||	or	Logical disjunction
^	exclusive or	Logical exclusion

Table 3.4 Truth Table for Operator !

p	**!p**	*Example (assume* **age = 24, weight = 140***)*
true	**false**	**!(age > 18)** is **false**, because **(age > 18)** is **true**
false	**true**	**!(weight == 150)** is **true**, because **(weight == 150)** is **false**

Table 3.5 Truth Table for Operator &&

p_1	**p_2**	**p_1 && p_2**	*Example (assume* **age = 24, weight = 140***)*
false	**false**	**false**	**(age > 28) && (weight <= 140)** is **false**, because **(age > 28)** is **false**
false	**true**	**false**	
true	**false**	**false**	**(age > 28) && (weight > 140)** is **false**, because **(weight > 140)** is **false**
true	**true**	**true**	**(age > 18) && (weight >= 140)** is **true**, because **(age > 18)** is and **(weight >= 140)** are both **true**

Table 3.6 Truth Table for Operator ||

p_1	**p_2**	**p_1 || p_2**	*Example (assume* **age = 24, weight = 140***)*
false	**false**	**false**	**(age > 34) || (weight >= 150)** is **false**, because **(age > 28)** are both **false**
false	**true**	**true**	**(age > 18) || (weight < 140)** is **true**, because **(age > 18)** is **true**
true	**false**	**true**	
true	**true**	**true**	

Table 3.5 Truth Table for Operator ^

p_1	**p_2**	**p_1 ^ p_2**	*Example (assume* **age = 24, weight = 140***)*
false	**false**	**false**	**(age > 34) ^ (weight > 140)** is **false**, because **(age > 34)** and **(weight >140)** are both **false**
false	**true**	**true**	**(age > 34) ^ (weight >= 140)** is **true**, because **(age > 34)** but **(weight >= 140)** is **true**
true	**false**	**true**	
true	**true**	**false**	

Listing 3.6 TestBooleanOperators.java

```
1  import java.util.Scanner:
2
3  public class TestBooleanOperators {
4      public static void main(String[] args) {
5          //Create a Scanner
6          Scanner input = new Scanner(System.in);
7
8          //Receive an input
9          System.out.print("Enter an integer: ");
10         int number = input.nextInt();
11
12         if (number % 2 == 0 && number % 3 == 0)
13             System.out.println(number + " is divisible by 2 and 3. ");
14
15         if (number % 2 == 0 || number % 3 == 0)
16             System.out.println(number + " is divisible by 2 or 3. ");
17
18         if (number % 2 == 0 ^ number % 3 == 0)
19             System.out.println(number + " is divisible by 2 or 3, but not both. ");
20
21     }
22 }
```

```
run:
Enter an integer: 4
4 is divisible by 2 or 3.
4 is divisible by 2 or 3, but not both.
BUILD SUCCESSFUL (total time: 7 seconds)
```

```
run:
Enter an integer: 18
18 is divisible by 2 and 3.
18 is divisible by 2 or 3.
BUILD SUCCESSFUL (total time: 7 seconds)
```