

8. Decision-Making Statements

Java

Fall 2009

Instructor: Dr. Masoud Yaghini

Outline

- The **if-then** Statement
- The **if-then-else** Statement
- The **switch** Statement
- References



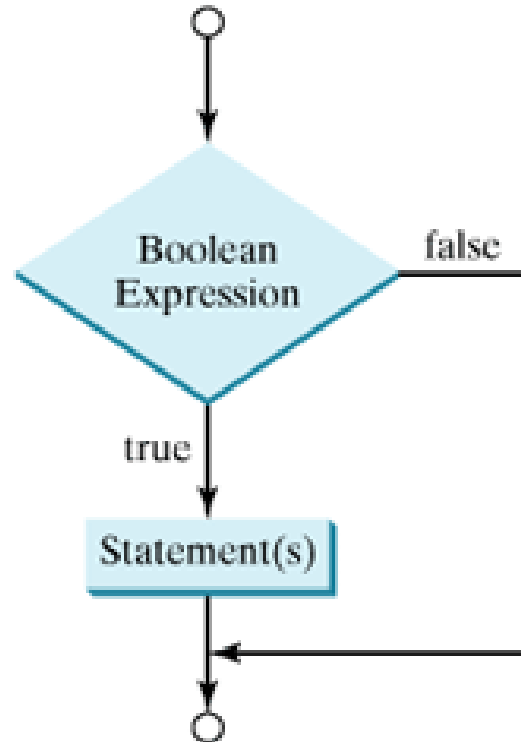
The if-then Statement

The if-then Statement

- The **if-then** statement is the most basic of all the control flow statements.
- It tells your program to execute a certain section of code only if a particular test evaluates to **true**.
- For example,
 - the **Bicycle** class could allow the brakes to decrease the bicycle's speed only if the bicycle is already in motion.

The if-then Statement

- An **if** statement executes statements if the **booleanExpression** evaluates to **true**.



The if-then Statement

- One possible implementation of the `applyBrakes` method could be as follows:

```
void applyBrakes()
{
    if (isMoving)           // the "if" clause: bicycle must moving
    {
        currentSpeed--;     // decrease current speed
    }
}
```

The if-then Statement

- The opening and closing braces are optional, provided that the "then" clause contains only one statement:

```
void applyBrakes()
{
    if (isMoving)
        currentSpeed--;           // decrease current speed
}
```

- without braces **is not recommended.**

The if-then-else Statement

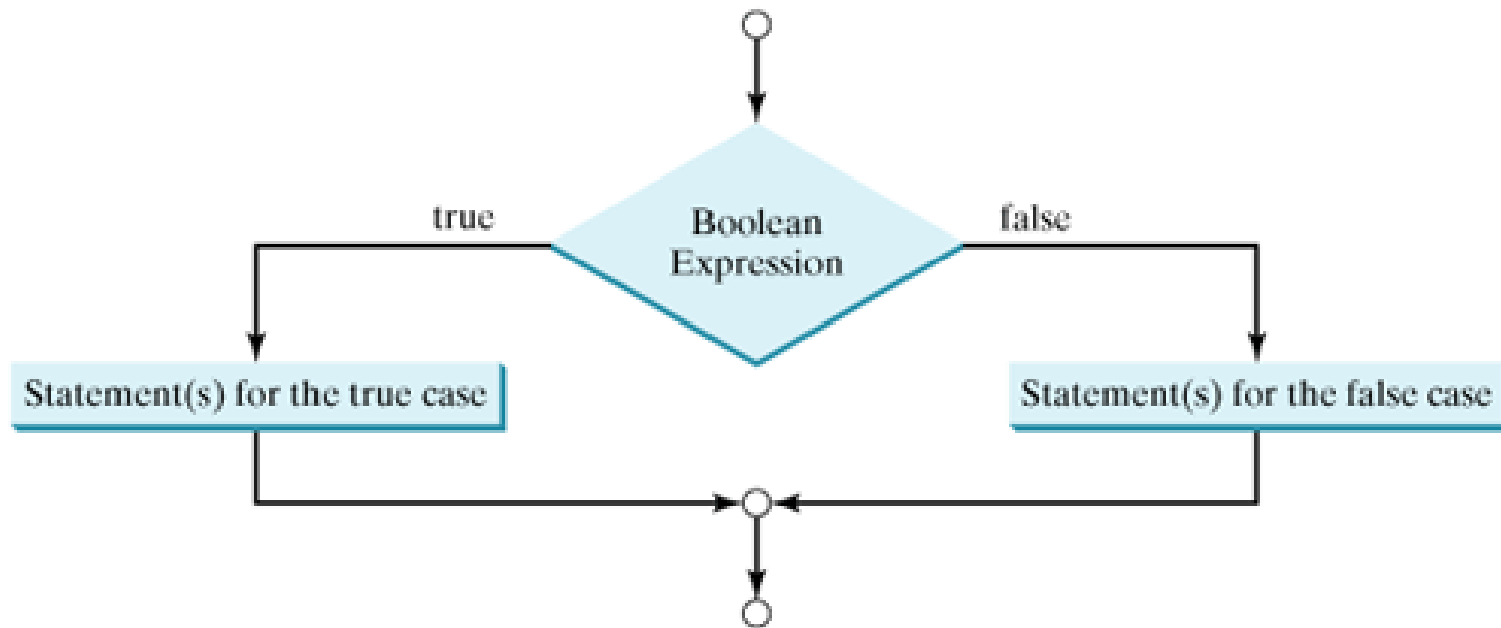


The if-then-else Statement

- The **if-then-else** statement provides a secondary path of execution when an "if" clause evaluates to **false**.
- You could use an **if-then-else** statement in the **applyBrakes** method to take some action if the brakes are applied when the bicycle is not in motion.

The if-then-else Statement

- An if ... else statement:



The if-then-else Statement

- In this case, the action is to simply print an error message stating that the bicycle has already stopped.

```
void applyBrakes()
{
    if (isMoving)
    {
        currentSpeed--;
    } else
    {
        System.out.println("The bicycle has already stopped!");
    }
}
```

The if-then-else Statement

- The following program assigns a grade based on the value of a test score:
 - [IfElseDemo.java](#)

The switch Statement

A decorative graphic on the left side of the slide. It consists of a large green shape with a white, rounded rectangular cutout on its right side. The text 'The switch Statement' is centered within this white cutout. Below the cutout, a dark blue horizontal bar extends to the right, starting from the edge of the green shape.

The switch Statement

- The `switch` statement allows for any number of possible execution paths.
- A `switch` works with the `byte`, `short`, `char`, and `int` primitive data types.
- It also works with some other types which discussed in later
- Example:
 - [SwitchDemo.java](#)

The switch Statement

- You could also implement the same thing with if-then-else statements:

```
int month = 8;  
if (month == 1)  
{  
    System.out.println("January");  
} else if (month == 2)  
{  
    System.out.println("February");  
}  
... // and so on
```

The switch Statement

- The body of a **switch** statement is known as a **switch block**.
- Any statement immediately contained by the switch block may be labeled with one or more **case** or **default** labels.
- The **switch** statement evaluates its expression and executes the appropriate case.

The switch Statement

- **if-then-else** vs. **switch** statement
 - Deciding whether to use **if-then-else** statements or a **switch** statement is sometimes a judgment call.
 - You can decide which one to use based on readability and other factors.
 - An **if-then-else** statement can be used to make decisions based on ranges of values or conditions,
 - Whereas a **switch** statement can make decisions based only on a single integer or enumerated value.

break statement

- Each **break** statement terminates the enclosing switch statement.
- The **break** statements are necessary because without them, control will flow sequentially through subsequent case statements.
- Example:
 - [SwitchDemo2.java](#)
- The output is:
 - case k = 10
 - case k = 15

break statement

- Example:
 - [SwitchDemo3.java](#)
- This is the output from the program:
Number of Days = 29
- Technically, the final **break** is not required because flow would fall out of the switch statement anyway.
- However, we recommend using a **break** so that modifying the code is easier.
- The **default** section handles all values that aren't explicitly handled by one of the **case** sections.



References



References

- S. Zakhour, S. Hommel, J. Royal, I. Rabinovitch, T. Risser, M. Hoeber, **The Java Tutorial: A Short Course on the Basics**, 4th Edition, Prentice Hall, 2006. (Chapter 3)

The End

