

Conditionals

Chapter 7 - Lecture Slides

(c) 2003 E.S.Boese

1

Conditional Statements

- **if** :: checks condition, if true, conditional statement executed
- **if-else** :: depending on conditions executes either the if or the else statements
- **switch** :: jumps to one of many labels depending on the value of switch expression

(c) 2003 E.S.Boese

2

if statement

```
if ( booleanExpression )
{
    statement;
    statement;
    ...
}
```

**** OR ****

```
if ( booleanExpression )
    statement;
```

Note that `{` and `}` are unnecessary when the if statement only has one single statement to do

Note also that there is no semicolon at the end of `if (...)`

(c) 2003 E.S.Boese

3

if statement

```
if ( booleanExpression )
{
    statement;
    statement;
    ...
}
```

**** OR ****

```
if ( booleanExpression )
    statement;
```

- Evaluate the boolean expression
- If it evaluates to true,
 - Do the statement(s) following
- Otherwise do nothing and continue with the program

(c) 2003 E.S.Boese

4

Boolean Expressions

- Boolean expressions evaluate to either
- Examples (refer to Chapter 9)
 - Is A greater than B?
 - Is A equal to B?
 - Is A less than or equal to B?
- In most cases, A and B can be any primitive data type.
- Primitives (int, double, char, etc.) work differently than objects (String, etc.)

Boolean Expressions con't

- Examples (refer to ch 9)
 - Is A greater than B?
 - Is A equal to B?
 - Is A less than or equal to B?

```
if ( A == B )
    C = A*2;
```

Boolean Expressions

- A boolean expression often uses one of Java's *equality operators* or *relational operators*, which all return boolean results:

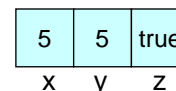
□ ==	equal to
□ !=	not equal to
□ <	less than
□ >	greater than
□ <=	less than or equal to
□ >=	greater than or equal to

■

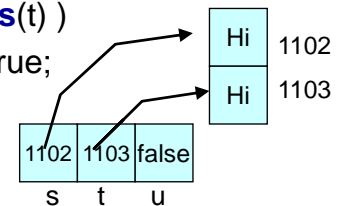
Boolean Expressions con't

- Primitives (int, double, char, etc.) work differently than objects (String, etc.)

```
int x = 5;
int y = 5;
boolean z = ( x == y );
if ( x >= y )
    y = y + x;
```

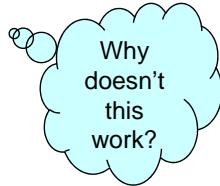


```
String s = "Hi";
String t = "Hi";
boolean u = ( s==t );
if ( s.equals(t) )
    u = true;
```



Grades if example

```
int grade = Integer.parseInt( tf_grade.getText() );  
if( grade > 70 )  
    labelGrade.setText( "You got a C" );  
if( grade > 80 )  
    labelGrade.setText( "That's a B" );  
if( grade > 90 )  
    labelGrade.setText( "You got a A" );  
if (grade < 70 )  
    labelGrade.setText( "Not so good..." );
```

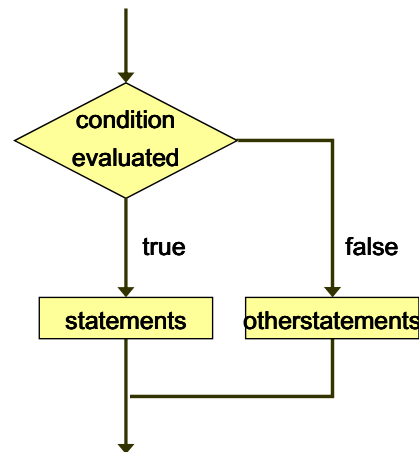


Grades if example – 1 way to fix it

```
int grade = Integer.parseInt( tf_grade.getText() );  
if( grade >= 70 )  
    labelGrade.setText( "You got a C" );  
if( )  
    labelGrade.setText( "That's a B" );  
if( )  
    labelGrade.setText( "You got a A" );  
if ( )  
    labelGrade.setText( "Not so good..." );
```

if else statement

```
if ( booleanExpression )  
{  
    statements;  
    multiplestatements;  
}  
else  
{  
    otherstatements;  
}
```

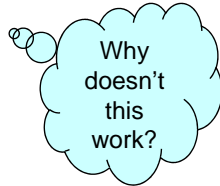


if else statement

- Evaluate the boolean expression
- If it evaluates to true,
 - Do the statement(s) following
- Otherwise do the other statements after the else

Grades if else if else example

```
if( grade > 70 )
    labelGrade.setText( "You got a C" );
else if( grade > 80 )
    labelGrade.setText( "That's a B" );
else if( grade > 90 )
    labelGrade.setText( "You got a A" );
else
    labelGrade.setText( "Not so good..." );
```



Grades if else if else example

```
int grade = Integer.parseInt( tf_grade.getText() );
if(
    )
    System.out.println( "You got a A" );
else if(
    )
    System.out.println( "That's a B" );
else if(
    )
    System.out.println( "You got a C" );
else
    System.out.println( "Not so good..." );
```

if and else

```
t = 5;
if ( x > 0 )
    if ( y > 0 )
        t = 1;
else
    t = 2;
```

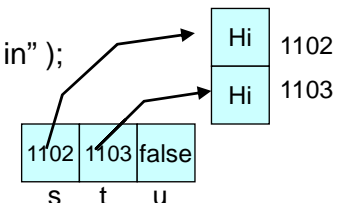
What is the value of t?

NOTE: an else clause is always matched to the nearest available unmatched if above it in the same scope

if-else Example Compare Strings

- Comparing strings – use `.equals()` method do NOT use `==`

```
String txt = textField.getText( );
if ( txt.equals( "Cookie Monster" ) )
    textarea.append( "Welcome back " + txt );
else
    textarea.append( "You're not logged in" );
```



instanceof operator

- **instanceof** operator returns true or false
- The left operand is the one you want to check
- The right operand is an object data type
- We will need to use this when we get to events

```
if ( obj instanceof JButton )
    ...
if ( obj instanceof JTextArea )
    ...
if ( obj instanceof JList )
    ...
```

if-else Example instanceof

- instanceof operator

```
if ( object instanceof JComboBox )
{
    textarea.append( "It's a combobox" );
}
else if( object instanceof JRadioButton )
{
    textarea.append( "It's a radio button" );
}
```

switch (Extra material)

```
switch( expression )
{
    case constant1:
        statements;
        statements;
        break;
    case constant2:
        statements;
        break;
    ...
    default:
        statements;
}
```

switch

- Select one of several actions, depending on the value of some expression
- If variable == constant, do this, otherwise if variable == constant2, do that, otherwise,
- Cases must be constants, of the same type as the expression
- Use default if no other cases match
- Break jumps out of the switch, otherwise keep executing until a break

switch

- Expression MUST evaluate to either a
- Constants must be literals or final variables
- 'default' is

switch example

```
switch( aValue )
{
    case 5:
        label.setText( "High" );
        statements;
        break;
    case 4:
    case 3:
    case 2:
        label.setText( "Medium" );
        break;
    default:
        label.setText( "Low" );
}
```

Grades switch example

```
int grade = Integer.parseInt( tf_grade.getText() );
switch( grade/10 )
{
    case 100:
    case 90:
        tf_grade.setText( "You ROCK!" );
        break;
    case 80:
        tf_grade.setText( "Doin' good" );
        break;
    case 70:
        tf_grade.setText( "Rather average" );
        break;
    default:
        tf_grade.setText( "Study More!" );
}
```

Good Fun Exercises

- Can any if-else structure be represented also as a switch structure?
- Can any switch structure also be represented as an if-else structure?

Summary

- if
- if else
- instanceof
- switch