Conditional Statements

- Consider:
  
  if the mouse location is contained in the rectangle, display message “success”

- Some programming constructs can choose between blocks of code to execute
A Vote Counting Example

Complete source code

Code to Update Votes:

```java
// Update votes and display vote counts
public void onMouseClick( Location point ) {
    if ( point.getX() < MID_X ) {
        countA++;
        infoA.setText( "So far there are " + countA + " vote(s) for A."");
    } else {
        countB++;
        infoB.setText( "So far there are " + countB + " vote(s) for B.");
    }
}
```
Syntax of the `if` Statement

- The condition is evaluated. If true, the bracketed code following `if` is executed; otherwise, the bracketed code after `else` is executed.

  ```java
  if (condition) {
      //do something
  } else {
      //do something else
  }
  ```

- The else part can be omitted

  ```java
  if (condition) {
      //do something here
  }
  ```

---

![Flowchart of `if-else` and `if` statements]
if Statement and 2D Objects

To check if the mouse is within a 2D object:
```java
public void onMouseClick ( Location point ) {
    if ( anObject.contains ( point ) ) {
        //do something here
    }
}
```

Comparison Operators

<table>
<thead>
<tr>
<th>Operator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;</td>
<td>Greater than</td>
</tr>
<tr>
<td>&lt;</td>
<td>Less than</td>
</tr>
<tr>
<td>==</td>
<td>Equal to</td>
</tr>
<tr>
<td>&lt;=</td>
<td>Less than or equal to</td>
</tr>
<tr>
<td>&gt;=</td>
<td>Greater than or equal to</td>
</tr>
<tr>
<td>!=</td>
<td>Not equal to</td>
</tr>
</tbody>
</table>
• Examples
  If a=25, b=30
  a<b evaluates to true
  a<=b evaluates to true
  a==b evaluates to false
  a!=b evaluates to true
  a>b evaluates to false
  a>=b evaluates to false

Since these expressions are evaluated to either true or false, they are called boolean expressions.

The boolean Data Type

• A data type that has only two values: true or false
• Can be declared with the word boolean

Ex:  boolean a=true;  //holds the value true
     boolean b=c>d;  //holds the result of c > d
     boolean k;  //creates k, which can take on true or false
Dragging a Box

// boolean variable to determine whether the box is grabbed
private boolean boxGrabbed;

// Save starting point and whether point was in box
public void onMousePress( Location point ) {
    lastPoint = point;
    boxGrabbed = box.contains( point );
}

// if mouse is in box, then drag the box
public void onMouseDrag( Location point ) {
    if ( boxGrabbed ) {
        box.move( point.getX() - lastPoint.getX(),
                  point.getY() - lastPoint.getY() );
        lastPoint = point;
    }
}

More Uses of The if else Statement

• Picks one choice among many

EX: Converting a score into a letter grade
if ( score >= 90 ) {
    gradeDisplay.setText( "The grade is A" );
} else if ( score >= 80 ) {
    gradeDisplay.setText( "The grade is B" );
} else if ( score >= 70 ) {
    gradeDisplay.setText( "The grade is C" );
} else {
    gradeDisplay.setText( "No credit is given" );
}
Combining Multiple Conditionals

• && (and) combines adjoining conditions in a way that the final result will be true only if all are true
  Ex: a && b && c
  is true if a, b, c are true

• || (or) combines adjoining conditions in a way that if any of them is true, the final result will be true
  Ex: a || b || c
  is true if any of a, b, c, is true

The Craps Example

• A block of code that uses || (or) to determine whether the player wins a game of Craps

```java
if ( roll == 7 || roll == 11 ) { // 7 or 11 wins on first throw
    status.setText( "You win!" );
} else if ( roll == 2 || roll == 3 || roll == 12 ) { // 2, 3, or 12 loses
    status.setText( "You lose!" );
} else { // Set roll to be the point to be made
    status.setText( "Try for your point!" );
    point = roll;
    ...
}
```

Complete source code
Nesting

• Suppose we want to decide among several choices based on several conditions, such as shown by the table:

<table>
<thead>
<tr>
<th>Sunny</th>
<th>Not sunny</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rich</td>
<td>Outdoor Concert</td>
</tr>
<tr>
<td>Not Rich</td>
<td>Ultimate Frisbee</td>
</tr>
</tbody>
</table>

• To do this, we use conditionals inside a conditional. This is called nesting.

Nesting Example

```java
if ( sunny ) {
    if ( rich ) {
        activityDisplay.setText( "outdoor concert" );
    } else { // not rich
        activityDisplay.setText( "play ultimate" );
    }
} else { // not sunny
    if ( rich ) {
        activityDisplay.setText( "indoor concert" );
    } else { // not rich
        activityDisplay.setText( "watch TV" );
    }
}
```
Creating a Game of Craps

By adding an outer conditional, we can effectively determine whether the player wins:

```java
if ( newGame ) {
    //Starting a new game
    if ( roll == 7 || roll == 11 ) {
        // 7 or 11 wins on first throw
        status.setText( "You win!" );
    } else if ( roll == 2 || roll == 3 || roll == 12 ) {
        // 2, 3, or 12 loses
        status.setText( "You lose!" );
    } else {
        // Set roll to be the point to be made
        status.setText( "Try for your point!" );
        point = roll;
        newGame = false; // no longer a new game
    }
}
```

Complete source code

Student To Do’s

- HW6 (Three problems)
  - Bring your solutions to lab.
  - (If no class on Monday, check your email for submission instructions)

- Read *Java: An Eventful Approach*
  - Ch. 4 (Today)
  - Ch. 5-6 (Next week)

- Don’t fall behind! We’re moving quickly through Java topics because we already know the concepts from Alice!