Syntax Recap

- if statement – do something or do nothing

```
if ( boolean condition ) {
  statements          // do these only if the condition is true
}
```

- if statement with else – do one of two alternatives

```
if ( boolean condition ) {
  statements          // do these only if the condition is true
} else {
  statements          // do these only if the condition is false
}
```

- else if – choose from more than two alternatives
  - any number of `else if`s can be included
  - include final `else` if "do nothing" is not an option

```
if ( boolean condition ) {
  statements          // do these only if the condition is true
} else if ( boolean condition ) {
  statements          // do these only if all previous conditions
                      // are false and this condition is true
} else {
  statements          // do these only if all of the conditions
                      // are false
}
```

- relational operators

```
<                     // less than
>                     // greater than
<=                    // less than or equal to
>=                    // greater than or equal to
==                    // equal to
!=                    // not equal to
```

- logical operators

```
||                    // or
&&                    // and
!                     // not
```

At the End of Class

- Make sure each sketch has a comment with the names of your group, and that you have saved your sketches.
- Use FileZilla to copy the entire folder for each of your sketches into your handin directory (**/classes/cs120/handin/*username***). You only need to hand in one copy for the group.

Conditionals Questions

- Does something happen differently at different times, or only happen sometimes? → if so, need a conditional
- Imagine that you only see a snapshot of the sketch, frozen at a moment in time – can you determine what should happen next? i.e. can you use the current values of system variables and/or animation variables to make the decision?
  - yes → on-the-spot decision
  - no, it depends on prior events / what has been going on → state machine
- For on-the-spot decisions –
  - How many alternatives are there for what happens?
    - "to do or not to do" (do something or do nothing) → if
    - two alternatives → if / else
    - more than two alternatives
      - "do nothing" is not an option → if / else if / … / else
      - "do nothing" is an option → if / else if / …
  - What are the alternatives? → body of each part
  - How do you decide which alternative to do? → boolean condition for each if / else if part

Exercises

For all sketches, use the conditionals questions to determine the structure for the `if` statement. Also be sure to include a comment with the names of your group at the beginning of the sketch.

1. Create a sketch named **sketch_240923a** which contains a 100x100 rectangle centered in the drawing window. The rectangle should be outlined in black, and should be filled red when the mouse is over it and white otherwise. (Hint: to be over the rectangle, the mouse position has to be simultaneously right of the rectangle's left side, left of the rectangle's right side, above the rectangle's bottom, and below the rectangle's top.)
2. Modify your sketch so that the rectangle also grows in width when the mouse is over the rectangle. (The rectangle should remain centered in the window.)
3. Modify your sketch so that the rectangle's width resets to 100 when it reaches the edge of the window.
4. Save a copy of your sketch as **sketch_240923b**, then modify the copy so the rectangle grows in both width and height when the mouse is over the rectangle, and the size resets to 100 when it reaches the edge of the window. Make sure that your sketch works even when the window isn't square – the rectangle's size should reset as soon as either edge reaches the edge of the window.

If you have time –

- Create a sketch named **sketch_240923c** which contains a 100x100 rectangle centered in the drawing window. The rectangle should grow in width when the mouse is moved to the right, shrink in width when the mouse is moved to the left, and not change when the mouse stays still (or moves only vertically). (Hint: recall the system variables `pmouseX` and `pmouseY` for the mouse's previous position.)
- Create a sketch named **sketch_240923d** which contains a 100x100 rectangle centered at a random position in the drawing window. Clicking the mouse inside the rectangle should cause it to move to a new random position.