



**Topic:** If-Else Statements

### Activity Guidelines

**Group Size:** 3

**Method of Assigning Students:** Count the number of students in the class, divide by 3, count off from 1 to the quotient, and group identical numbers.

**Materials:**

- ✓ Handout (one copy per group) with questions to be answered at the end of the session

**Roles:**

**Coordinator/Leader:** Clarifies goals and objectives, allocates roles for each team member and divides the tasks within the group.

**Monitor/Evaluator:** Person designed to evaluate the different ideas to approach the problem and make an accurate judgment of the most beneficial option.

**Implementer:** Person in charge to transform discussions and ideas into a technical solution for the given problem.

**Individual Accountability:** Every student will be working on teams for the exercises. Every team member is given a specific role which allows everyone on the team to get involved and participate in the problem-solving activity. Every team member should participate in solving the exercises according to their determined role in the group.

### Activity Summary

Students are provided with a handout that describes the different comparison and logical operators used to test for true or false. Additionally, this handout provides student with the information to know how to handle situations in which strings must be compared. Students within their teams will be given time to discuss comparison and logical operators, as a class they will be required to execute a program that involves if-else statements.



## INTRODUCTION TO COMPUTER SCIENCE PEER SESSION

### If-Else Statements

#### Operators

Relational	< > <= >= (note: does not work for Strings)
Equality	== != (note: does not work for Strings)
Logical AND	&&
Logical OR	

#### Compare strings

String word = "something";

word.equals("something"); // ← true

word.equals("Something"); // ← false

word.equalsIgnoreCase("Something"); // ← true

word.compareTo("Something logically bigger"); ← -17

**equals** returns:

This method returns true if the String are equal; false otherwise.

**equalsIgnoreCase** returns:

This method returns true if the argument is not null and the Strings are equal, ignoring case; false otherwise.

**compareTo** returns:

The value 0 if the argument is a string lexicographically equal to this string; a value less than 0 if the argument is a string lexicographically greater than this string; and a value greater than 0 if the argument is a string lexicographically less than this string.

#### True & False

x	y	&&	
true	true	true	true
true	false	false	true
false	true	false	true
false	false	false	false



As a class execute the following program:

```
if (you like video games)
    stand up
if (you like anime)
    raise both of your hands hand
```

```
if (you are a Pokemon fan)
    clap your hands once
else if (you are a Bioshock)
    clap your hands two times
else
    clap your hands three times
```

```
if (you are standing up && you are a Pokemon fan)
    jump once
else if (you are standing up || you are a Pokemon fan)
    point out the logical mistake
else if (you have both of your hands up || like anime)
    lower both of your hands
```

```
if(you are taller than 1.8 yards) { //1.64 m
    say hi to the person at your right
    if(there is no one at your right)
        say hi to the wall
} else {
    say hi to the person at your left
    if(there is no one at your left)
        say hi to the window
}
```

```
if(you are a Computer Science major)
    shake hands with the T.A.
else
    shake hands with the Peer Leader
```

