




Part 1-1-4-4a: Conditional Weather

In this part of the activity, you will review Conditionals and learn about complex conditionals, which can be helpful in situations where you want to respond to the status of a combination of conditions.


- 18 Review information on  **Conditionals**, answering the review questions on the website to verify that you have learned what you need.
- 19 Within the `getWeatherAdvice` method of the `WeatherConditionals` class, remove your `return` statement and create a variable of primitive type `boolean` with the identifier `windy` and a value `false`.
- 20 Add a conditional statement to your program to determine if the string `description` contains “windy” and set `windy` appropriately.
- 21 Use `windy` and `temperature` in another conditional statement to determine if it is not windy and also warm enough (more than 30 degrees) to go outside. Test your program for the following results:

temperature	description	result
34	sunny	It's safe to go outside, 34 degrees and sunny.
32	windy	Too windy or cold! Enjoy watching the weather through the window.
33	snow	It's safe to go outside, 33 degrees and snow.
30	snow	Too windy or cold! Enjoy watching the weather through the window.
30	windy	Too windy or cold! Enjoy watching the weather through the window.

- 22 Review  **Complex Conditionals**, answering the review questions to verify that you have learned what you need.
- 23 Add a conditional statement so that your `getWeatherAdvice` method can determine a weather condition where the description contains “snow” and the temperature is over 100 degrees. Return a message expressing disbelief at this combination. Test your program.
- 24 Assume you can store more than one weather condition at a time, and that you don't like to go out if it is both freezing and cloudy. One or the other is fine, but not both. In computer science you can evaluate this kind of condition using **short circuit evaluation**. Read about  **short circuit evaluation** and answer the review question.

short circuit evaluation

A complex conditional expression where the subsequent condition(s) might not be executed.

- 25 Assume the boolean variables `freezing`, `cloudy`, `fair`, and `sunny`. Determine the values that would cause a short circuit evaluation in the following statements.
- a. `if (freezing && cloudy)`
 - b. `if (sunny || fair)`
 - c. `if (!sunny && !fair)`
- 26 Sometimes rewriting a conditional expression can make it easier for humans to read or understand.  **Learn DeMorgan's Law** and answer the review questions.
- 27 Given the boolean variables `sunny`, `clear`, `raining`, and `snowing`, rewrite the following conditional expressions using **DeMorgan's Law**.
- a. `if (!sunny || !clear)`
 - b. `if (!(!raining && !snowing))`
- 28 Similar to DeMorgan's Law, you can rewrite relational operators when they are used with the not operator `!`. For example, “not less-than” is the same as “greater-than-or-equal-to”. Rewrite the following conditional expressions without using `!`.
- a. `if (!(temperature > 75))`
 - b. `if (!(temperature <= 100))`
 - c. `if (!(temperature == 32))`


DeMorgan's Law

A method you can use to rewrite the negative version of complex conditionals, for example `!(cold && raining)`.

Part 1-1-4-4b: Planning for a Weather App

Now that you have the necessary knowledge of conditionals, program your app to make recommendations based on several weather indications.

- 29 A client wants an app that provides guidance as they prepare to go for a hike in the morning. You have access to the following information:
- a. `temperature` as an `int`
 - b. `windchill` as an `int`
 - c. `humidity` as an `int`
 - d. `description` as a `String`

The `temperature` and `windchill` units are Fahrenheit, and `humidity` represents a percentage. The `description` will be one of the  **Yahoo! Weather conditions** in the table referenced in Step 5. Plan out how you would advise this client based on these inputs.

- 30 As directed by your teacher, work with a partner to refine your plan for advising the hiker. Determine favorable (or unfavorable) hiking conditions, such as rain, heat, cold, and the other conditions listed.
- 31 When you are ready, implement a new method within `WeatherConditionals` using the signature shown below and filling in the body of the method (line 3) with conditionals that you designed in the previous two steps.

```
1: public static String getHikingAdvice(int temperature,  
2:           int windchill, int humidity, String description){  
3:  
4: }
```

- 32 Call your method from `StringTester`, passing in various values to make sure that you have tested all of your **boundary conditions**. Testing boundary conditions means that you should test all of the conditions in your `if` statements, confirm that the correct code is executed, and that *all* statements can be reached or executed.

boundary conditions

In Java, the `if` statements that limit or define the execution path of code.