* // indicates that the entire line is comment
* /\*  \*/ are used to indicate multi-line comments
* JavaScript has seven types of data types
  + undefined
  + null
  + boolean
  + string
  + symbol
  + number
  + object
* Variables - names that are used to refer to specific data that is stored; can change values over time
  + Use the keyword "var" to declare a variable: *var myFirstVariable;*
  + Variable names can be made up of numbers, letters, and $ or \_, but may not contain spaces or start with a number
  + You can assign values to variables, either by directly using values or other variables; examples:
    - *myFirstVariable = 5;*
    - *myFirstVariable = anotherVariable;*
  + It is common to initialize a variable to an initial value in the same line that it is declared; example:
    - var numOfCats = 2;
  + Before variables are given values, they have a value of "undefined"
    - Doing a mathematical operation on an undefined variable will result in NaN (which means "Not a Number")
* Function and variable names are case sensitive
* It is best practice to write variables names in camelCase
* Mathematical operations can be performed on variables
  + Addition operator: +
  + Subtraction operator: -
  + Multiplication operator: \*
  + Division operator: /
  + Remainder operator: %
* Increment a variable - add one to the variable
  + *var1++;* is equivalent to *var1 = var1 + 1;*
* Decrement a variable - subtract one from the variable
  + *var1--;* is equivalent to *var1 = var1 - 1;*
* Decimal numbers are sometimes called floating numbers or floats
* There are operators to allow you to do a mathematical operation and an assignment in one step; for example: +=, -=, \*=, /=
* A string is enclosed in single (') or double (") quotation marks
  + If a string contains a quotation mark, you have to "escape" it by putting a backslash (\) before the quotes - this indicates that the quote is not the end of the string
  + If single quotes are used for indicating a string, then double quotes can be used inside the string without needing to use backslash
  + If double quotes are used for indicating a string, then single quotes can be used inside the string without needing to use backslash
  + Backslash (\) can be used to escape other characters as well:
    - \' (single quote)
    - \" (double quote)
    - \\ (backslash)
    - \n (newline)
    - \r (carriage return)
    - \t (tab)
    - \b (backspace)
    - \f (form feed)
  + Strings can be concatenated (or joined) with other strings using the + operator
    - += can be used to concatenate a string to the end of an existing string
  + .length property can be used to find the length of a string variable or string literal
  + bracket notation can be used to get the character at a specific index within a string
    - First character within a string is at index 0
      * Example: string[0]
    - You can get the last character of a string by subtracting one from the string's length:
      * string[string.length-1]
      * You can also use this notation to find nth character to last
        + Example: 2nd character to last: string[string.length-2]
  + In JavaScript, you cannot change the individual characters of a string variable, but you can replace the entire value of a string variable with another string