Review: Identifiers

Which of the following is a valid identifier?

A) *trek  (begins with a special character)
B) document  (reserved word)
C) _am_i_valid  
D) how-do-you-do  (contains hyphens)
E) 22catch  (begins with a digit)

Answer: (C)
### Review: Data types

<table>
<thead>
<tr>
<th>Data Type</th>
<th>String</th>
<th>Boolean</th>
<th>Integer</th>
<th>Real</th>
</tr>
</thead>
<tbody>
<tr>
<td>1267</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12.67</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>“1267”</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>true</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>“1267 is `integer`”</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>false</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*NOTE the escape characters in the string data.*

### Review: Variables

- Which of the following is/are false?

A) A variable always has a name.

**B) Value of the variable cannot change.**

C) A variable has a type.

D) Variable has a location in memory.

Answer: **(B)**
Review: Variables

- Declaration: `var name;`
- Initialization: `var name = initialValue;`
- Assignment: `name = expression;`

Classify:
A) `var pie = 3.14;`  Initialization
B) `var doublePie;`  Declaration
C) `var name = “Djokovic”;`  Initialization
D) `doublePie = pie * 2;`  Assignment

Review: Expressions

<table>
<thead>
<tr>
<th>Expression</th>
<th>Type</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>X = 20 + 10;</code></td>
<td>Numerical</td>
<td><code>X = 30</code></td>
</tr>
<tr>
<td><code>X = “Comp”</code></td>
<td>String</td>
<td><code>X = Comp</code></td>
</tr>
<tr>
<td><code>X = 2 + 4 * 4 + 2</code></td>
<td>Numerical</td>
<td><code>X = 20</code></td>
</tr>
<tr>
<td><code>X = (2 + 4) * (4 + 2)</code></td>
<td>Numerical</td>
<td><code>X = 36</code></td>
</tr>
<tr>
<td><code>X = “Comp” + “110”</code></td>
<td>String</td>
<td><code>X = Comp110</code></td>
</tr>
<tr>
<td><code>X = “Comp” + 110</code></td>
<td>Mixed</td>
<td><code>X = Comp110</code></td>
</tr>
<tr>
<td><code>X = (5&gt;3) &amp;&amp; (30&lt;20)</code></td>
<td>Boolean</td>
<td><code>X = false</code></td>
</tr>
</tbody>
</table>

NOTE:
1. Evaluation Order of expressions
2. Mixed expressions
Exercise 1

• Write a simple html page that prints the sum and product of two numbers.
  – In Javascript code, declare and initialize two variables.
  – Print the variable values using document.write()
  – Compute the sum of the variables and assign it to a third variable.
  – Compute the product of the variables and assign it to a fourth variable.
  – Print the results (3rd and 4th variable values) using document.write()

• Answer html page [here](#)

Before we move ...

Finding errors in Javascript is not easy!

• Program Incrementally
  – If page does not work, focus on recently added section.

• Use block commenting.
  – Comment everything. Start uncommenting logical blocks one by one.
  – Stop when a particular block does not work.
  – Fix the error and continue.

• Use Firebug! (Install from [here](#))
  – It works only on Firefox browser.
  – Very useful to debug and detect errors
Firebug

Exercise 1

First variable a =

Note it hasn’t printed a’s value yet!

You can break the execution at a particular line to check the state.

You can check the values of the variables at any stage.

To continue to next line press this.

Review: Inputs

- Prompt()

  var var_name = prompt( “Please enter your name”, “Harry Potter” );
Review: Inputs

- confirm()

```javascript
var result = confirm("Do you want to continue?");
```

- result = true
- result = false

Exercise 2

- Extend exercise 1 to input the numbers using prompt() function.

Answer file: [here](#)
Review: Outputs

- `document.write()`

  ```javascript
  var x = 10;
document.write(x);
  ```

- `alert()`

  ```javascript
  var x = 10;
alert("Value of x is: 
  \n  + x);
  ```

Exercise 3

- Extend exercise 2 to show the output as shown below:

  ![JavaScript Alert Window](image)

  Answer file: [here](#)
Review: Arrays

- Array: An ordered set of variables
- Declaration:
  
  ```javascript
  var numbers = new Array();
  numbers[0] = 10;
  ```

Exercise 4

- Extend exercise 3 to take three numbers from the user and compute their sum and products.
  - DO NOT use three variables to hold values this time. Use an array to hold 3 numbers.

Answer file: [here](#)