Lab 1: Browser Tools

In this lab you'll explore the browser tools available with the major browsers: Internet Explorer, Mozilla Firefox and Google Chrome. You'll examine the various consoles, panes and code viewers as you look at some HTML pages with JavaScript. You'll *not* have to code any JavaScript (a bit too early for that) but. you may have to change a line or two.

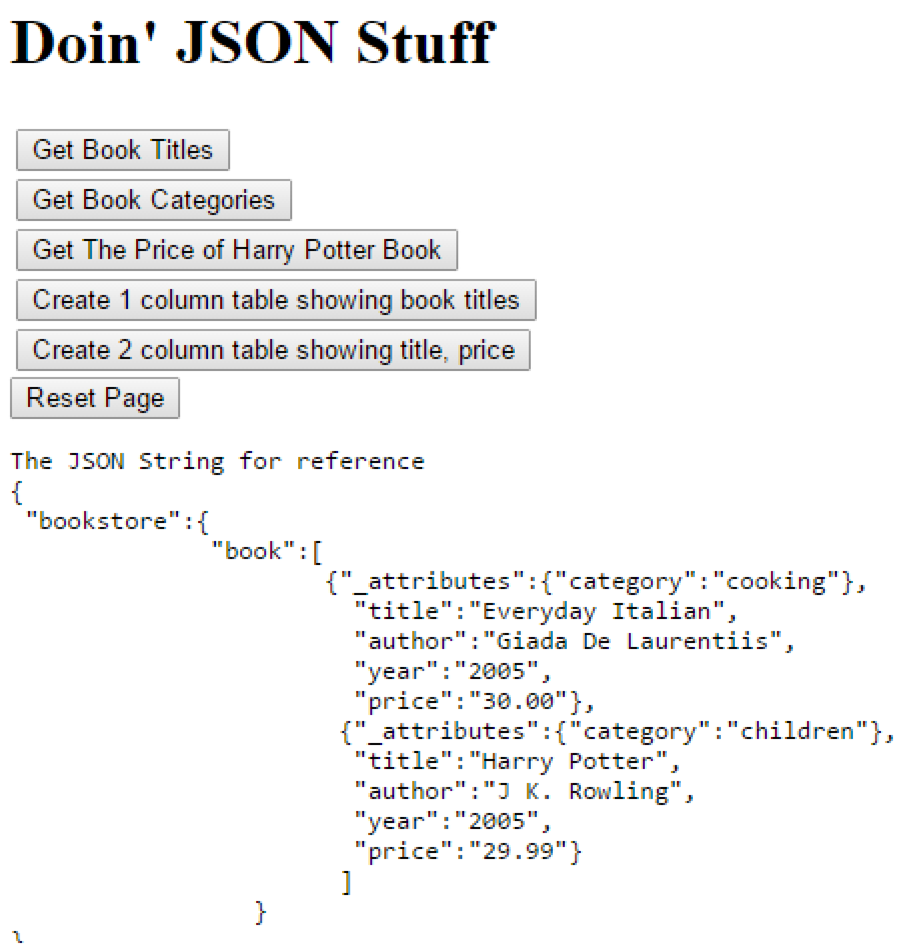
Try not to concentrate on the JavaScript used in these examples; we're only showing the tools, not the JavaScript!

You'll be using these tools in your labs (and anytime you write JavaScript)

The File You'll be Using

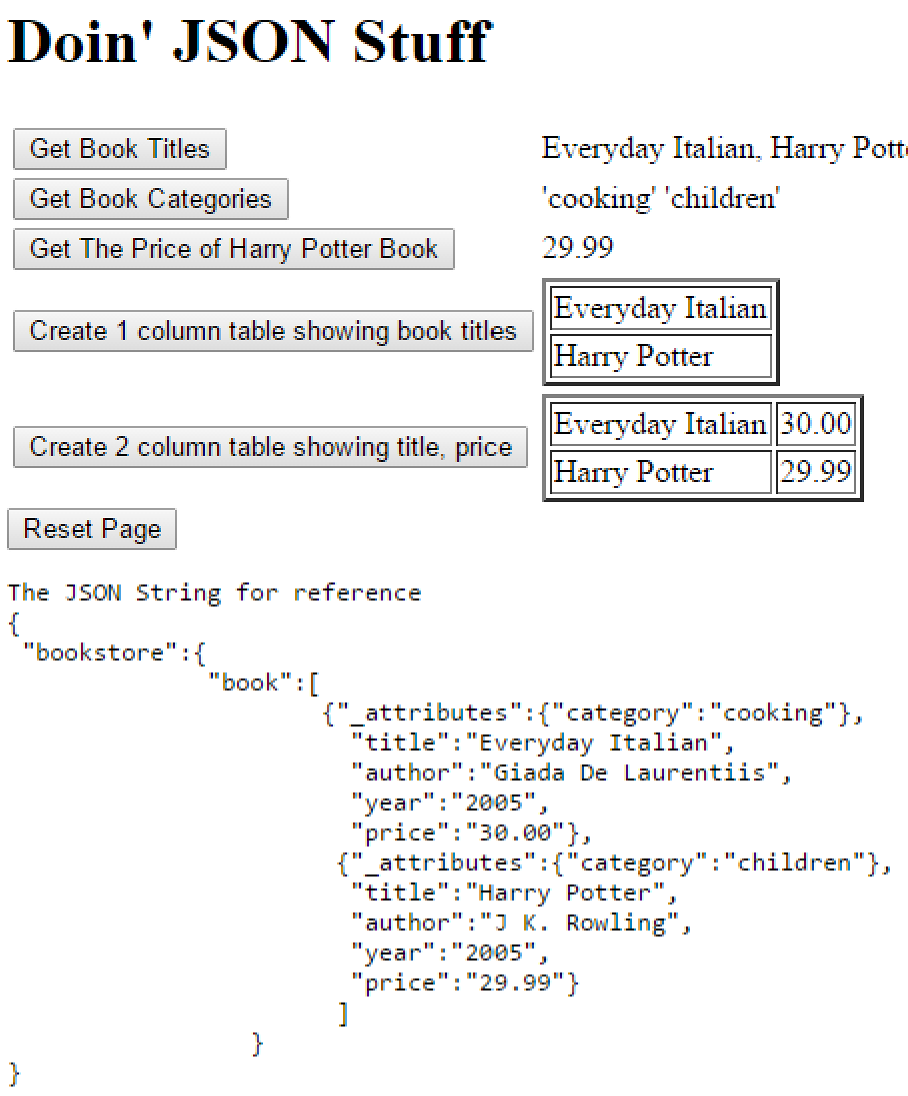
You'll be using the file **JSONExerciseWorkCopy.html** but first, let's look at what this file would look like once all errors are removed.   Inside your *Lab1* folder pick a browser – as long as it's IE, FF or Chrome – and load the following page: **JSONExerciseSolution.html**

It'll look like this.


Doin' JSON Stuff
Get Book Titles
Get Book Categories
Get The Price of Harry Potter Book
Create 1 column table showing book titles
Create 2 column table showing title, price
Reset Page

```
The JSON String for reference
{
 "bookstore":{
              "book":[
                      {"_attributes":{"category":"cooking"},
                        "title":"Everyday Italian",
                        "author":"Giada De Laurentiis",
                        "year":"2005",
                        "price":"30.00"},
                       {"_attributes":{"category":"children"},
                        "title":"Harry Potter",
                        "author":"J K. Rowling",
                        "year":"2005",
                        "price":"29.99"}
                       ]
                 }
}
```

Click all the buttons on the page (except for *reset*); it should resemble:


Get Book Titles
Get Book Categories
Get The Price of Harry Potter Book
Create 1 column table showing book titles
Create 2 column table showing title, price
Reset Page

Doin' JSON Stuff

Everyday Italian, Harry Potter

'cooking' 'children'

29.99

Everyday Italian
Harry Potter

Everyday Italian	30.00
Harry Potter	29.99

```
The JSON String for reference
{
 "bookstore":{
              "book":[
                      {"_attributes":{"category":"cooking"},
                        "title":"Everyday Italian",
                        "author":"Giada De Laurentiis",
                        "year":"2005",
                        "price":"30.00"},
                       {"_attributes":{"category":"children"},
                        "title":"Harry Potter",
                        "author":"J K. Rowling",
                        "year":"2005",
                        "price":"29.99"}
                       ]
                 }
}
```

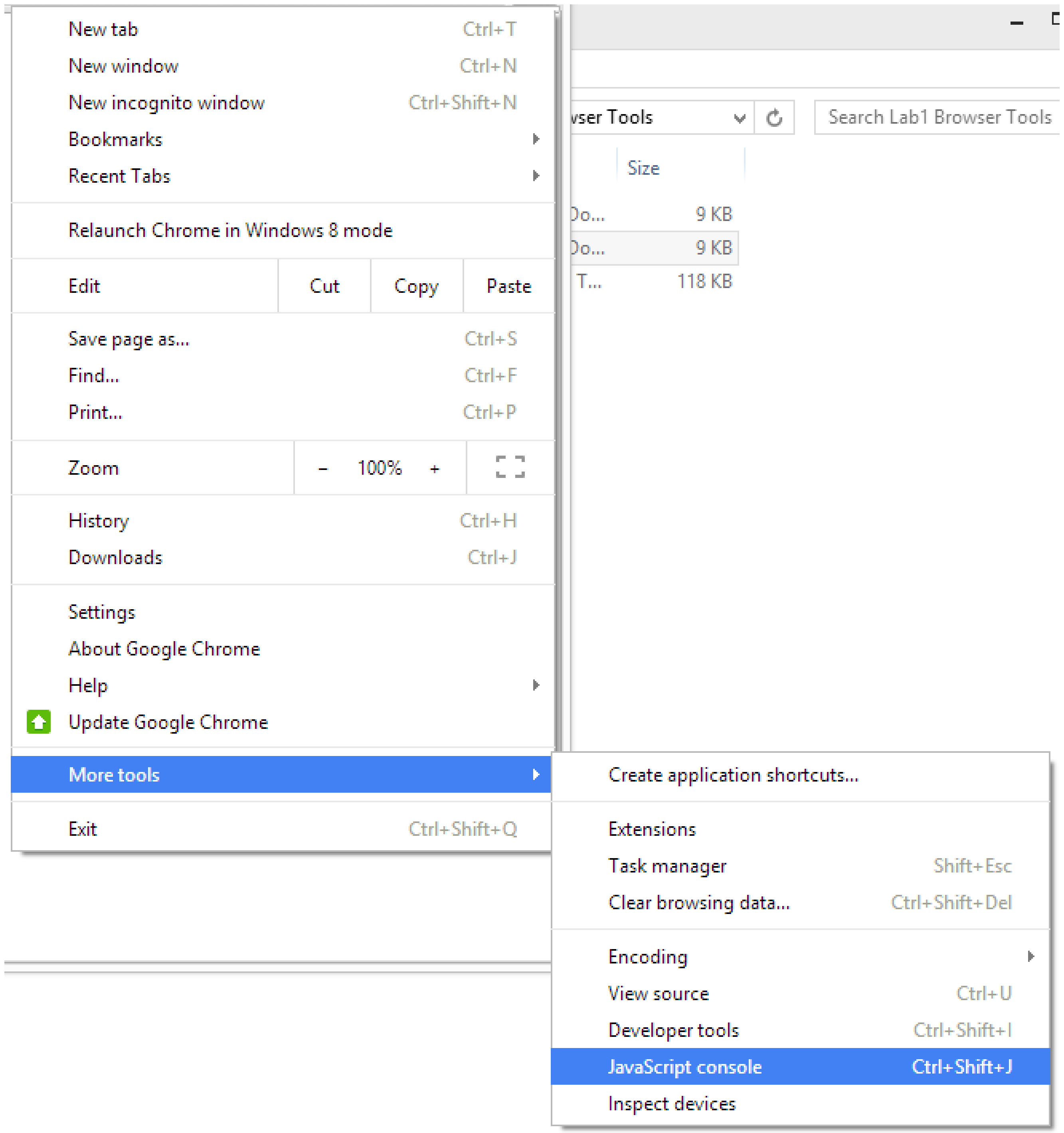
Use the *source code* in this file to correct errors in the work copy.

Now we'll look at a flawed version of this page in each of the major browsers and examine the tools available to help us fix it. Read the Chrome section to see *what the errors are* and skip to your browser to see *how to detect the errors*..

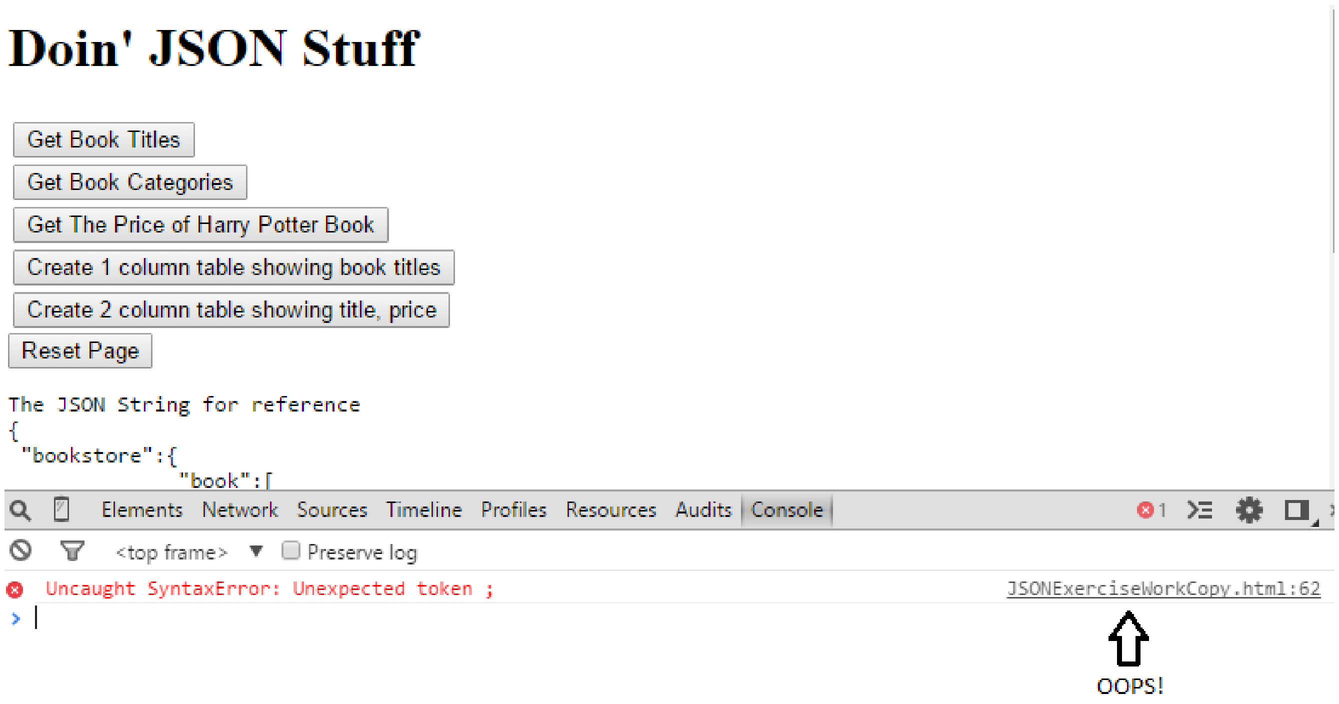
For you Chrome users...

Load **JSONExerciseWorkCopy.html** into Chrome. The page should display as before. At the ***upper right corner*** is a button that allows access to the Chrome menus. ☰ Click on this button and you'll see a slew of menus.

Select the JavaScript Console from the menus below (SHIFT-CTRL-J also brings up the console):



You'll see the *JavaScript console* which displays any parsing errors or any runtime errors encountered during execution. Here's what the console looks like after selected:



Doin' JSON Stuff

Get Book Titles
Get Book Categories
Get The Price of Harry Potter Book
Create 1 column table showing book titles
Create 2 column table showing title, price
Reset Page

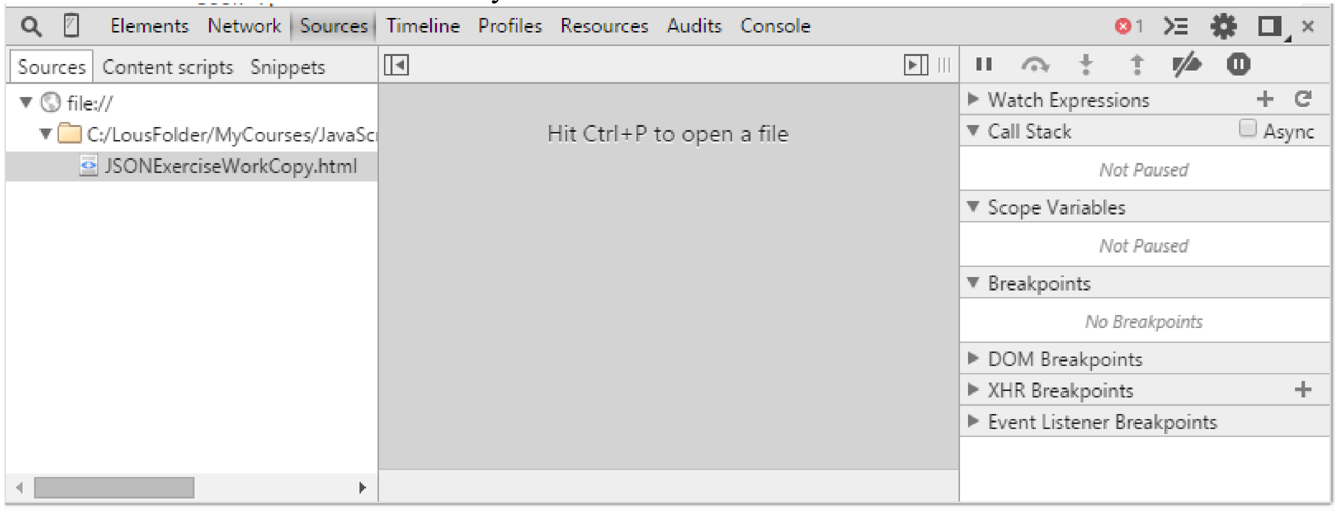
```
The JSON String for reference
{
 "bookstore":{
              "book":[
```

Elements Network Sources Timeline Profiles Resources Audits Console
<top frame> Preserve log

```
Uncaught SyntaxError: Unexpected token ;          JSONExerciseWorkCopy.html:62
```

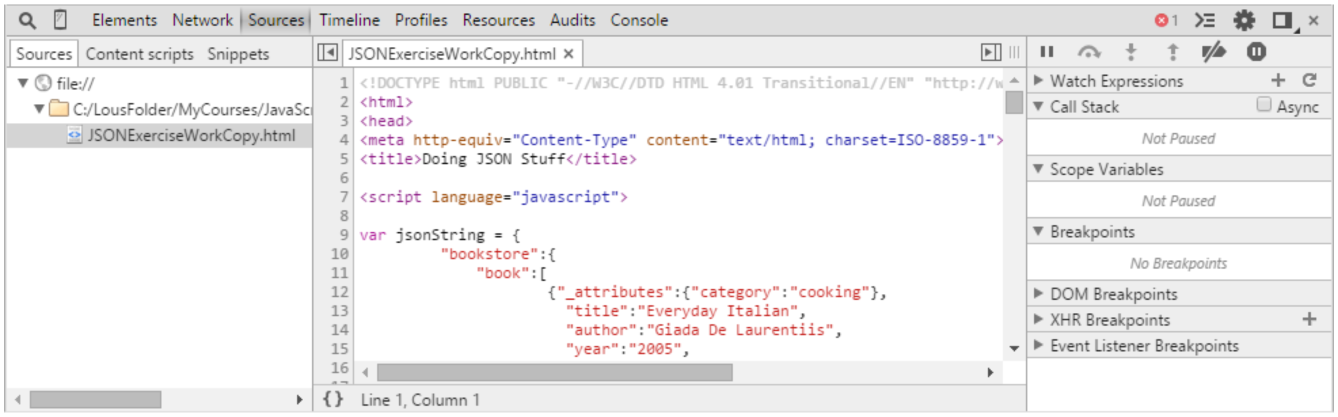
The diagnostic varies depending on the Chrome version used. Note that Chrome caught some sort of error on line 62. We'll check that out soon but first, a few views of the Chrome tools.

Click on the *Sources* tab. Here's what you'll see:



Elements Network Sources Timeline Profiles Resources Audits Console
Sources Content scripts Snippets
file://
C:/LousFolder/MyCourses/JavaSc
JSONExerciseWorkCopy.html
Hit Ctrl+P to open a file
Watch Expressions
Call Stack
Async
Not Paused
Scope Variables
Not Paused
Breakpoints
No Breakpoints
DOM Breakpoints
XHR Breakpoints
Event Listener Breakpoints

Note the *file name* being shown is hilighted to the left. Click on that; the middle pane will display the code in that file:

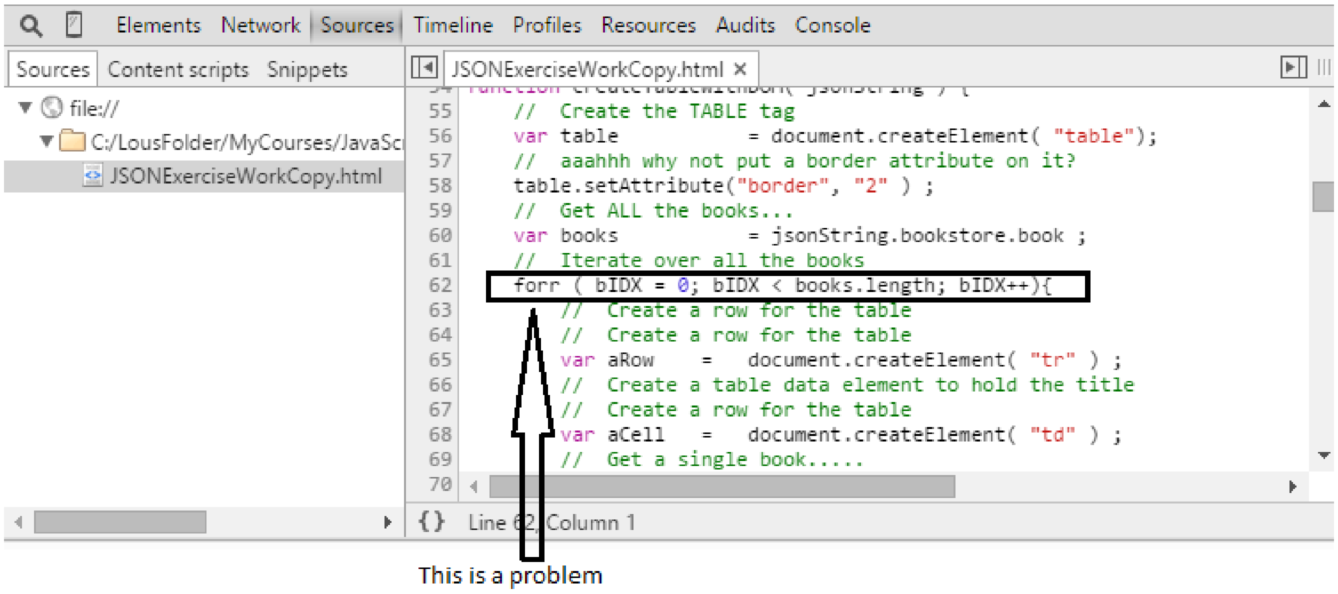

Elements Network Sources Timeline Profiles Resources Audits Console
Sources Content scripts Snippets
file://
C:/LousFolder/MyCourses/JavaSc
JSONExerciseWorkCopy.html
JSONExerciseWorkCopy.html
Watch Expressions
Call Stack
Async
Not Paused
Scope Variables
Not Paused
Breakpoints
No Breakpoints
DOM Breakpoints
XHR Breakpoints
Event Listener Breakpoints
Line 1, Column 1

```
 1  <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://w
 2  <html>
 3  <head>
 4  <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
 5  <title>Doing JSON Stuff</title>
 6
 7  <script language="javascript">
 8
 9  var jsonString = {
10          "bookstore":{
11              "book":[
12                      {"_attributes":{"category":"cooking"},
13                        "title":"Everyday Italian",
14                        "author":"Giada De Laurentiis",
15                        "year":"2005",
16
```

Note the *line numbers to the left* of the source code. Any errors shown in the console refer to these line numbers.

Later we'll explore the options *to the right*; there are equivalents in the other browser's tools.

Click back to *Console and click on the error shown to the right.* There's an error on line 62. You'll see something resembling:


Elements Network Sources Timeline Profiles Resources Audits Console
Sources Content scripts Snippets
file://
C:/LousFolder/MyCourses/JavaSc
JSONExerciseWorkCopy.html
JSONExerciseWorkCopy.html
Line 62, Column 1
This is a problem

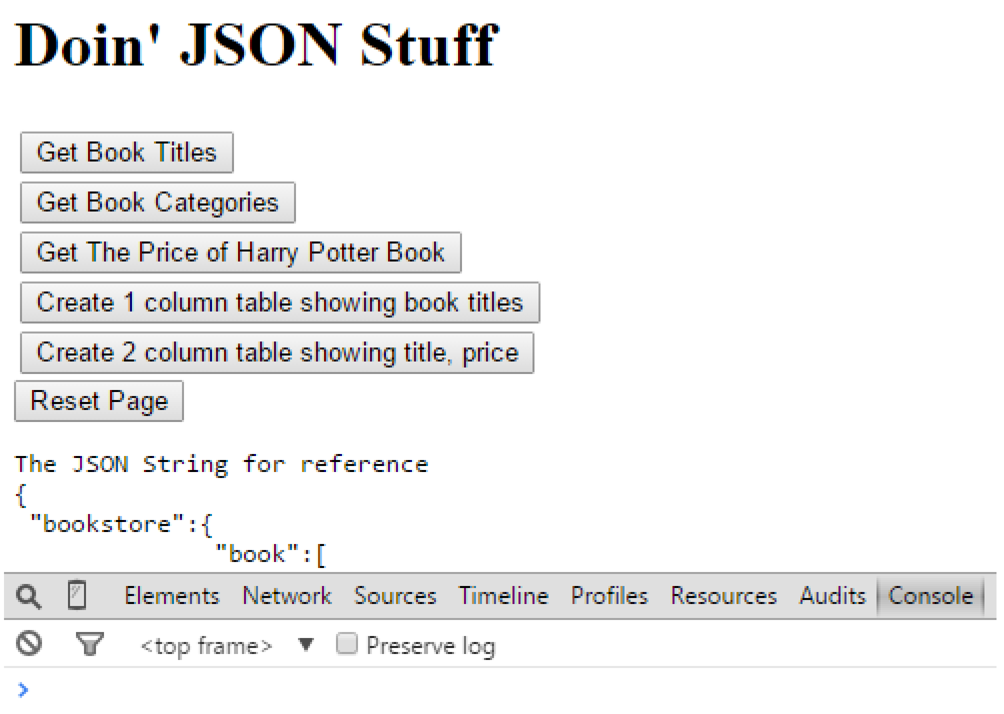
```
55     //  Create the TABLE tag
56     var table           = document.createElement( "table");
57     //  aaahhh why not put a border attribute on it?
58     table.setAttribute("border", "2" ) ;
59     //  Get ALL the books...
60     var books           = jsonString.bookstore.book ;
61     //  Iterate over all the books
62     forr ( bIDX = 0; bIDX < books.length; bIDX++){
63         //  Create a row for the table
64         //  Create a row for the table
65         var aRow    =   document.createElement( "tr" ) ;
66         //  Create a table data element to hold the title
67         //  Create a row for the table
68         var aCell   =   document.createElement( "td" ) ;
69         //  Get a single book.....
70
```

Yeah. *Forr????????* Even though you are not savvy in JavaScript I think you sense that *forr* is not a valid JavaScript command/keyword/whatever. Maybe change this to the word ***for*** and try again.

Open the file in a text editor (any editor you have is fine; Notepad++ works well) and change that *forr to* ***for***. Reload the page and select the *Console* tab. You'll likely see something like this:

Yeah! Looking good.

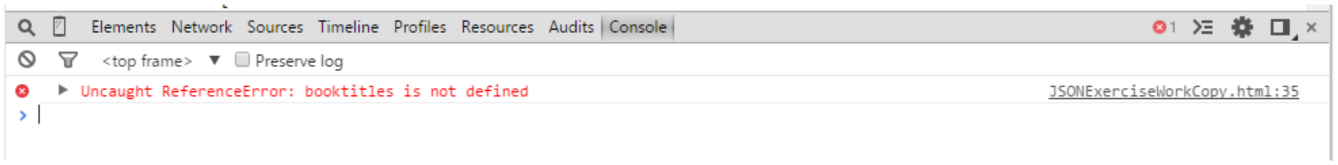
Click on 'Get Book Titles'



Doin' JSON Stuff
Get Book Titles
Get Book Categories
Get The Price of Harry Potter Book
Create 1 column table showing book titles
Create 2 column table showing title, price
Reset Page
Elements Network Sources Timeline Profiles Resources Audits Console
<top frame>
Preserve log

```
The JSON String for reference
{
 "bookstore":{
              "book":[
```

And you are greeted with:



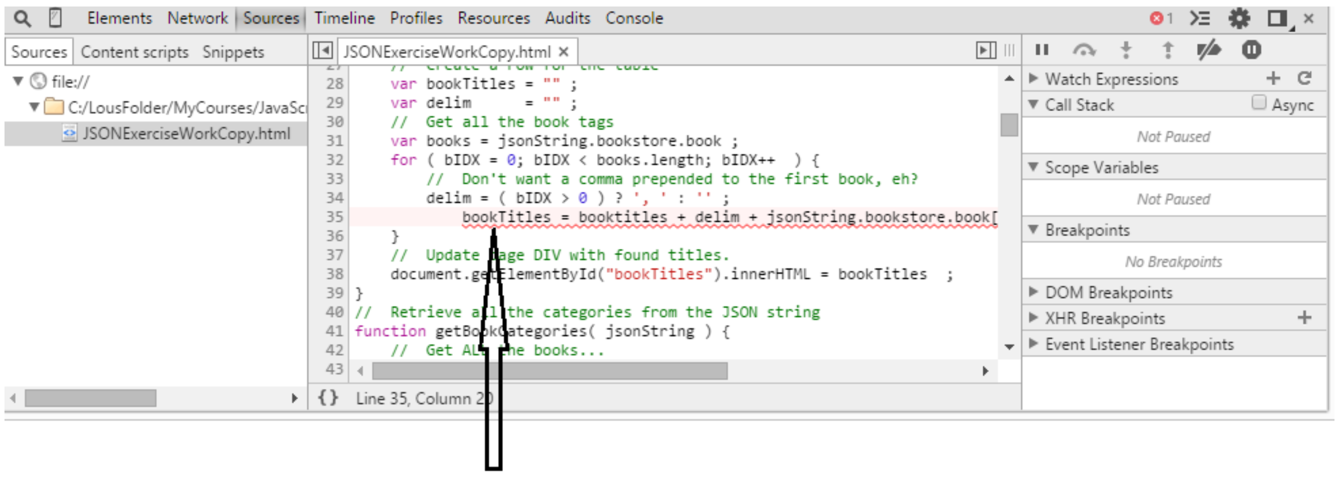
Elements Network Sources Timeline Profiles Resources Audits Console
<top frame>
Preserve log

```
Uncaught ReferenceError: booktitles is not defined          JSONExerciseWorkCopy.html:35
```

OOOOWWWWW!!!!!

Some errors arise when the code executes. Here, this wrong code had an error that we did not see until you clicked the button which caused the errorenous code to execute.

Again, click on the error showing the line number and you'll see:



Elements Network Sources Timeline Profiles Resources Audits Console
Sources Content scripts Snippets
file://
C:/LousFolder/MyCourses/JavaSc
JSONExerciseWorkCopy.html
Watch Expressions
Call Stack
Async
Not Paused
Scope Variables
Not Paused
Breakpoints
No Breakpoints
DOM Breakpoints
XHR Breakpoints
Event Listener Breakpoints

```
28     var bookTitles = "" ;
29     var delim      = "" ;
30     //  Get all the book tags
31     var books = jsonString.bookstore.book ;
32     for ( bIDX = 0; bIDX < books.length; bIDX++  ) {
33         //  Don't want a comma prepended to the first book, eh?
34         delim = ( bIDX > 0 ) ? ', ' : '' ;
35             bookTitles = booktitles + delim + jsonString.bookstore.book[
36     }
37     //  Update page DIV with found titles.
38     document.getElementById("bookTitles").innerHTML = bookTitles  ;
39 }
40 //  Retrieve all the categories from the JSON string
41 function getBookCategories( jsonString ) {
42     //  Get ALL the books...
43
```

```
Line 35, Column 20
```

Note the line AND column number!

LAB1: Browser Tools

A moment's inspection reveals that there are variables named ***bookTitles*** and ***booktitles;*** **l**ooks like one of them is wrong.

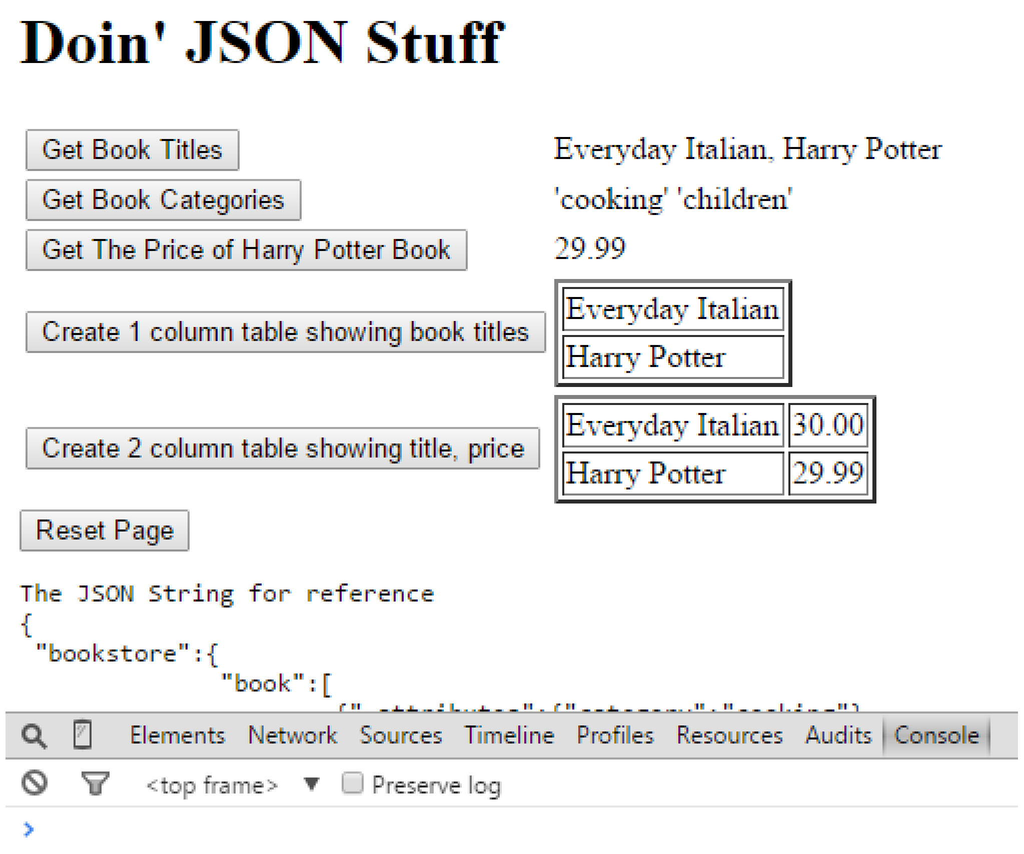
Line 28 and the first word in line 35 have ***bookTitles***; the error in question states that ***booktitles*** is undefined. Doesn't take a rocket scientist to deduce that **booktitles is misspelled.**

Back to your favorite text editor and change *booktitles* to ***bookTitles***; rerun the page, click all the buttons (except for *reset)* and *check out the Console*. You'll see:

Looking good!

Click the ***Reset Page*** button.

And you see:



Doin' JSON Stuff

Everyday Italian, Harry Potter

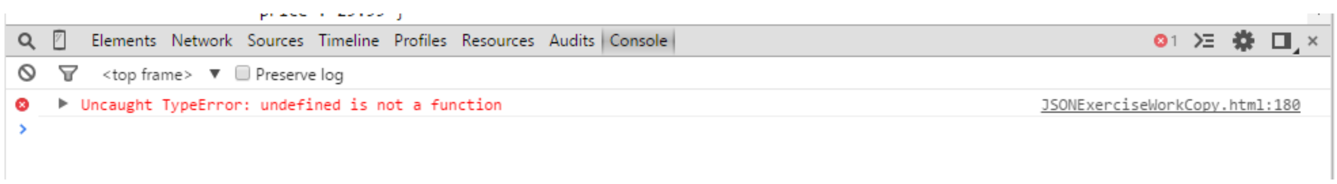
'cooking' 'children'

29.99

Everyday Italian
Harry Potter

Everyday Italian	30.00
Harry Potter	29.99

```
The JSON String for reference
{
 "bookstore":{
              "book":[
```



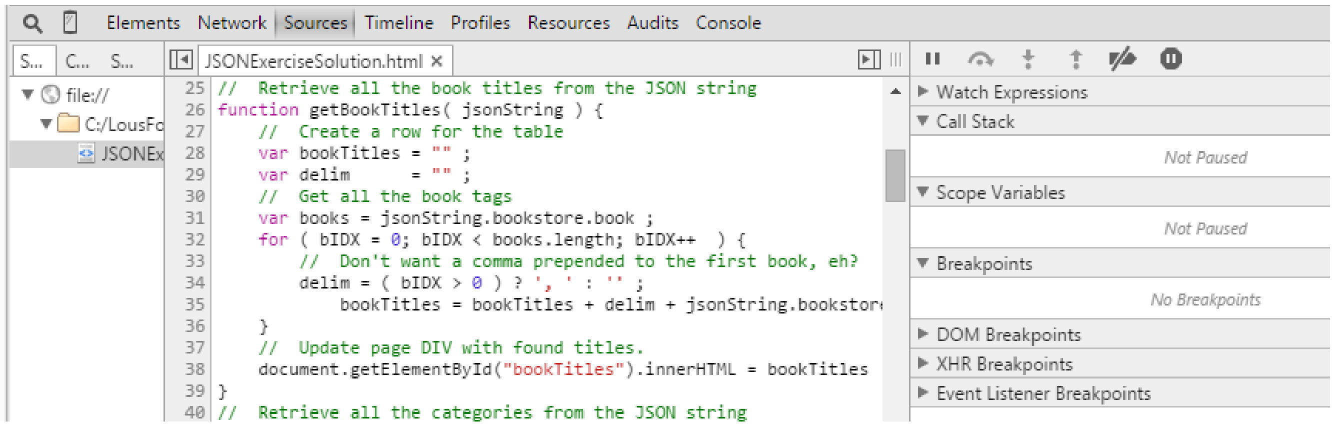
```
Uncaught TypeError: undefined is not a function          JSONExerciseWorkCopy.html:180
```

Figure this one out by looking at the code around it; fix, redisplay, check console.

Using the debugger

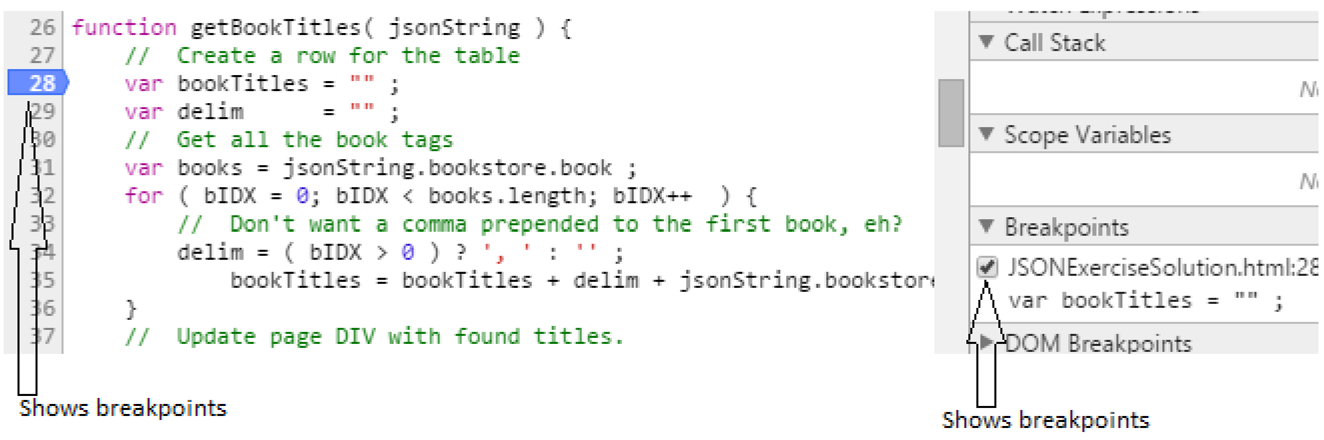
The debuggers for all teh majorr browsers behave the same way. You load your page, *set one or more breakpoints*, then execute your script/program. When processing gets to a statement with a breakpoint, *processing stops* and you can *examine the values of variables known to your program.*

We'll see a few screen shots. Load *JSONExerciseSolution.html* into Chrome, call up the *developer tools* as before (CTRL-SHIFT-J works), click the *Sources* pane and click the html file to the left and scroll the source file in the center pane until *line 28 is visible*. Here's what you should see:



```
25 //  Retrieve all the book titles from the JSON string
26 function getBookTitles( jsonString ) {
27     //  Create a row for the table
28     var bookTitles = "" ;
29     var delim      = "" ;
30     //  Get all the book tags
31     var books = jsonString.bookstore.book ;
32     for ( bIDX = 0; bIDX < books.length; bIDX++  ) {
33         //  Don't want a comma prepended to the first book, eh?
34         delim = ( bIDX > 0 ) ? ', ' : '' ;
35             bookTitles = bookTitles + delim + jsonString.bookstor
36     }
37     //  Update page DIV with found titles.
38     document.getElementById("bookTitles").innerHTML = bookTitles
39 }
40 //  Retrieve all the categories from the JSON string
```

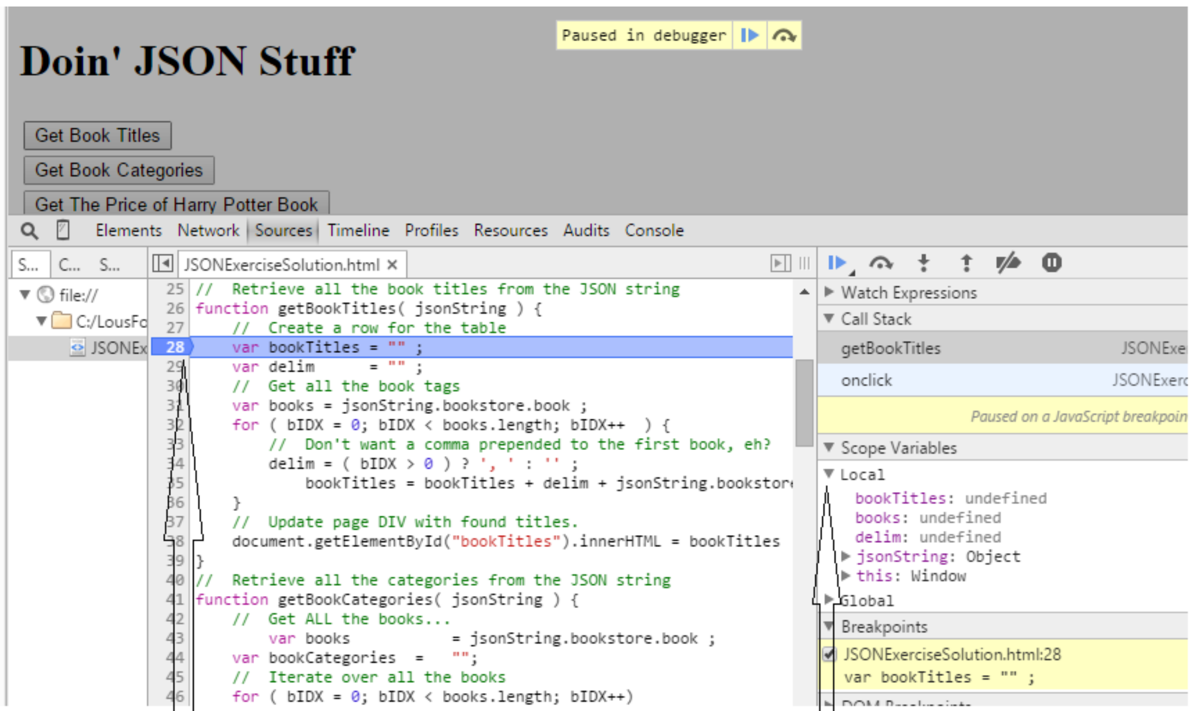
Let's tell Chrome to *pause execution by setting a breakpoint* at line 28. Click on the *line number in the source pane.* You'll see:



```
26 function getBookTitles( jsonString ) {
27     //  Create a row for the table
28     var bookTitles = "" ;
29     var delim      = "" ;
30     //  Get all the book tags
31     var books = jsonString.bookstore.book ;
32     for ( bIDX = 0; bIDX < books.length; bIDX++  ) {
33         //  Don't want a comma prepended to the first book, eh?
34         delim = ( bIDX > 0 ) ? ', ' : '' ;
35             bookTitles = bookTitles + delim + jsonString.bookstore
36     }
37     //  Update page DIV with found titles.
```

Chrome will pause when execution gets to line 28.

Let's run the script now. Click on Get Book Titles to run the script. Chrome stops at line 28. Here's what you'll see:

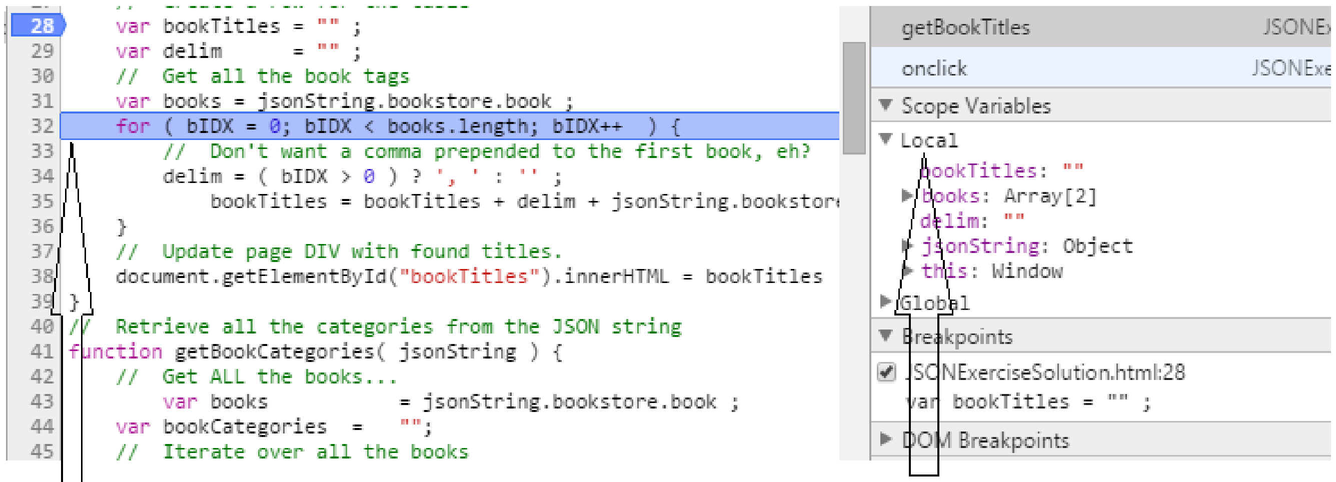

Doin' JSON Stuff
Get Book Titles
Get Book Categories
Get The Price of Harry Potter Book
Paused in debugger
Elements Network Sources Timeline Profiles Resources Audits Console
JSONExerciseSolution.html ×
file://
C:/LousFo
JSONEx
Breakpoint line hilighted
Values of variables at this point of program execution

```
25 //  Retrieve all the book titles from the JSON string
26 function getBookTitles( jsonString ) {
27     //  Create a row for the table
28     var bookTitles = "" ;
29     var delim      = "" ;
30     //  Get all the book tags
31     var books = jsonString.bookstore.book ;
32     for ( bIDX = 0; bIDX < books.length; bIDX++  ) {
33         //  Don't want a comma prepended to the first book, eh?
34         delim = ( bIDX > 0 ) ? ', ' : '' ;
35             bookTitles = bookTitles + delim + jsonString.bookstore
36     }
37     //  Update page DIV with found titles.
38     document.getElementById("bookTitles").innerHTML = bookTitles
39 }
40 //  Retrieve all the categories from the JSON string
41 function getBookCategories( jsonString ) {
42     //  Get ALL the books...
43         var books          = jsonString.bookstore.book ;
44     var bookCategories =   "";
45     //  Iterate over all the books
46     for ( bIDX = 0; bIDX < books.length; bIDX++)
```

```
▶ Watch Expressions
▼ Call Stack
  getBookTitles                      JSONExe
  onclick                            JSONExerc
              Paused on a JavaScript breakpoin
▼ Scope Variables
▼ Local
    bookTitles: undefined
    books: undefined
    delim: undefined
  ▶ jsonString: Object
  ▶ this: Window
▶ Global
▼ Breakpoints
☑ JSONExerciseSolution.html:28
  var bookTitles = "" ;
```

The *main display* tells us execution is *paused in debugger*, the debugger *hilights the line we're paused at*, and the debugger *shows the values of variables*.

To *step into* your program – execute **one line at a time** – hit the ***F11 key***. Hit F11 **three times until line 32 is hilighted**. Here's what the console looks like now:

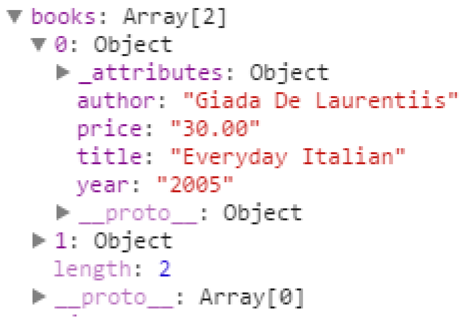


```
28     var bookTitles = "" ;
29     var delim      = "" ;
30     //  Get all the book tags
31     var books = jsonString.bookstore.book ;
32     for ( bIDX = 0; bIDX < books.length; bIDX++  ) {
33         //  Don't want a comma prepended to the first book, eh?
34         delim = ( bIDX > 0 ) ? ', ' : '' ;
35             bookTitles = bookTitles + delim + jsonString.bookstore
36     }
37     //  Update page DIV with found titles.
38     document.getElementById("bookTitles").innerHTML = bookTitles
39 }
40 //  Retrieve all the categories from the JSON string
41 function getBookCategories( jsonString ) {
42     //  Get ALL the books...
43         var books          = jsonString.bookstore.book ;
44     var bookCategories =   "";
45     //  Iterate over all the books
```

```
  getBookTitles                      JSONE
  onclick                            JSONExe
▼ Scope Variables
▼ Local
    bookTitles: ""
  ▶ books: Array[2]
    delim: ""
  ▶ jsonString: Object
  ▶ this: Window
▶ Global
▼ Breakpoints
☑ JSONExerciseSolution.html:28
  var bookTitles = "" ;
▶ DOM Breakpoints
```

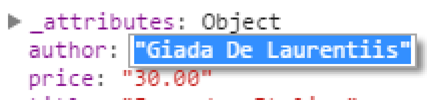
Note the *Scope Variables* pane is *updated* with new values; the execution of lines 28 through 31 assigned values to previously *undefined* variables.

Now we haven't explored what *objects* or *arrays* are yet and the use of the keyword *this* is beyond the scope of the intro class but we see that the debugger knows that *books* is an array and *jsonString* is an object. We can examine the values of these variables. Click on the ▶ to the left of *books* and click on the ▶ to the left of **0**. The debugger *expands the selection* and shows:

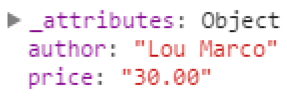


```
▼ books: Array[2]
  ▼ 0: Object
    ▶ _attributes: Object
      author: "Giada De Laurentiis"
      price: "30.00"
      title: "Everyday Italian"
      year: "2005"
    ▶ __proto__: Object
  ▶ 1: Object
    length: 2
  ▶ __proto__: Array[0]
```

Any of these values *may be changed* by *double-clicking* in a field and typing a new value. **The new value is in effect only for the current execution; your source code is NOT CHANGED!**

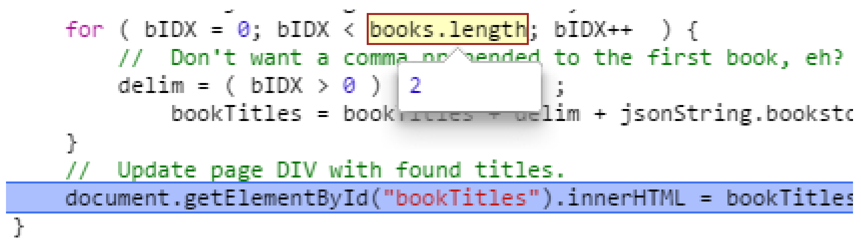


```
▶ _attributes: Object
  author: "Giada De Laurentiis"
  price: "30.00"
```



```
▶ _attributes: Object
  author: "Lou Marco"
  price: "30.00"
```

Continue stepping through the program by hitting F11 until the program stops at line 38. Note the *bookTitles* string has changed value. You may *hover* over variables/expressions and note their values:



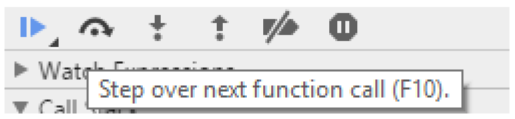
```
    for ( bIDX = 0; bIDX < books.length; bIDX++  ) {
        //  Don't want a comma prepended to the first book, eh?
        delim = ( bIDX > 0 )  2                       ;
            bookTitles = book                elim + jsonString.bookstc
    }
    //  Update page DIV with found titles.
    document.getElementById("bookTitles").innerHTML = bookTitles
}
```

Here, *books.length* is an *expression* that does not appear in the *variables* pane.

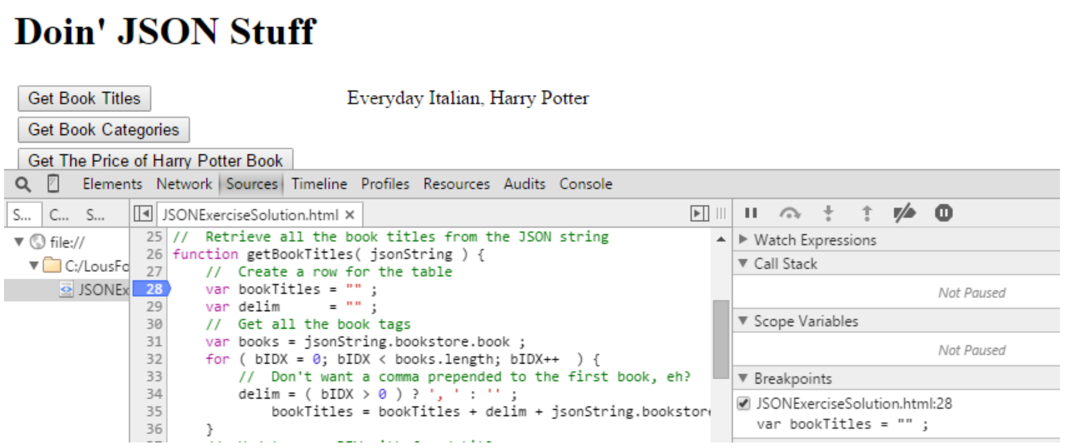
Use the buttons atop the right portion of the tools pane to continue execution or use keyboard shortcuts:

F8	Execute until next breakpoint (Run)
F10	Step **over** the function (do not show execution of the called function)
F11	Step **into** the function (use this to step line-by-line)

The buttons have *tooltip* help; just hover over them:



Hit F8 to **run** until the next breakpoint. Since there are no more breakpoints, the script will complete, you'll get output and the console pane will tell you the script *is not paused*.



Doin' JSON Stuff

Everyday Italian, Harry Potter

```
25 //  Retrieve all the book titles from the JSON string
26 function getBookTitles( jsonString ) {
27     //  Create a row for the table
28     var bookTitles = "" ;
29     var delim      = "" ;
30     //  Get all the book tags
31     var books = jsonString.bookstore.book ;
32     for ( bIDX = 0; bIDX < books.length; bIDX++  ) {
33         //  Don't want a comma prepended to the first book, eh?
34         delim = ( bIDX > 0 ) ? ', ' : '' ;
35             bookTitles = bookTitles + delim + jsonString.bookstor
36     }
```

Not Paused

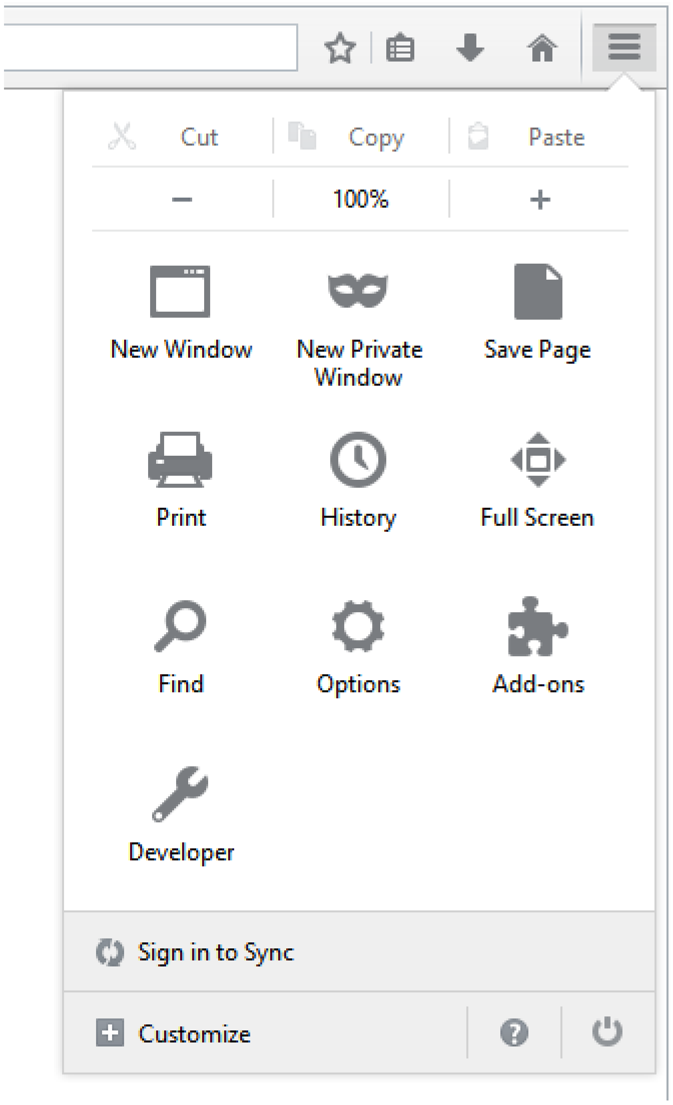
Not Paused

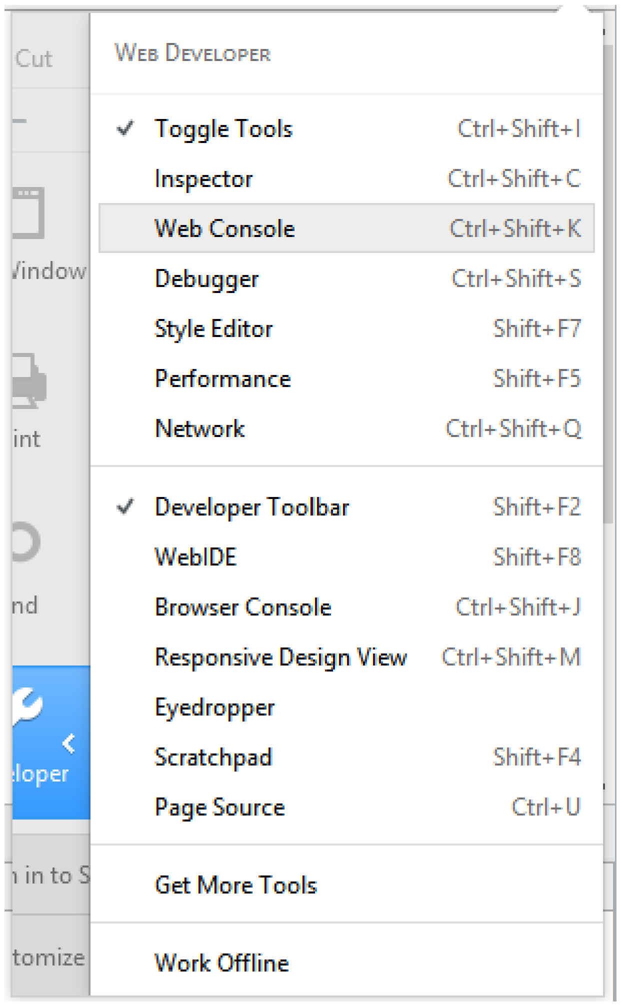
☑ JSONExerciseSolution.html:28
var bookTitles = "" ;

Clear breakpoints by clicking on the line number (28) or clicking on the *checkbox* in the *Breakpoints* pane.

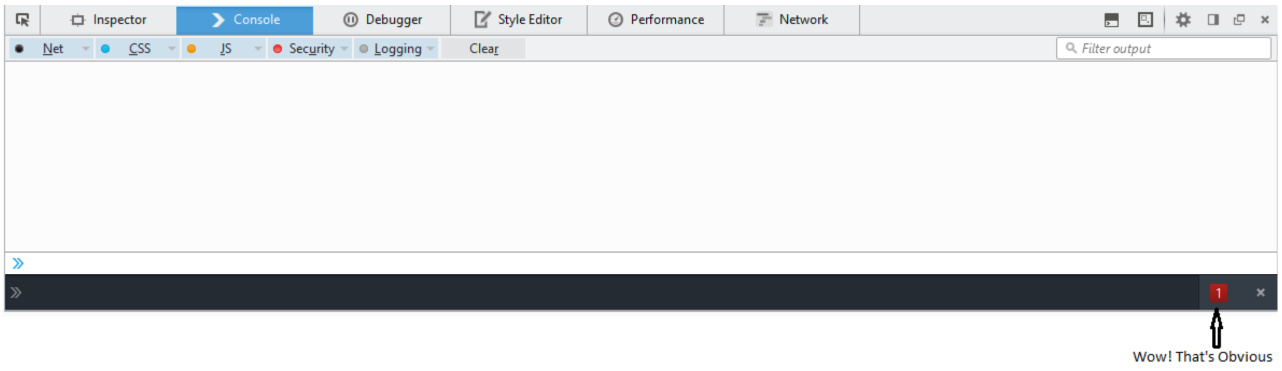
For you FireFox Users

Call up the *developer tools* by looking for the same icon/button to the far right and selecting the ***Web Console***:

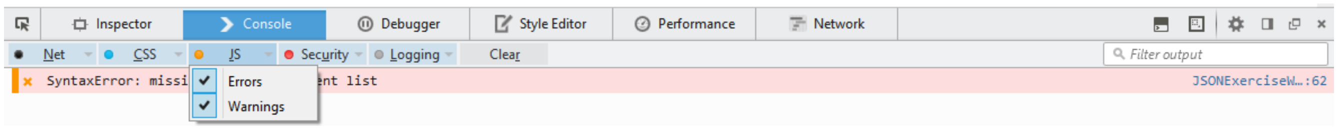




The *web console* looks like this:



Yeah the console window is clean *except for this leetle itty bitty red circle.* Click on the tab and select *warnings and errors*.



```
SyntaxError: missi...nt list                JSONExerciseW…:62
```

Looks like we finally see something!

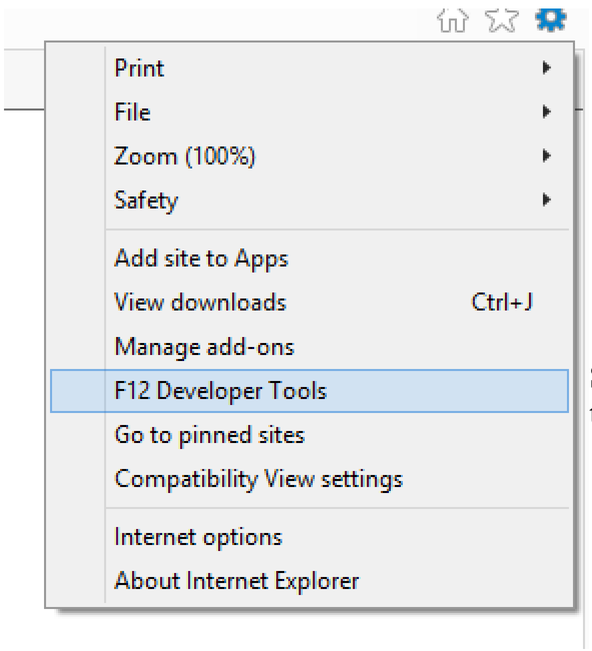
With FireFox, ***check if the JS tab has errors/warnings enabled***. For some odd reason, it's *disabled by default*.

Go through the changes cited in the *Chrome* section until the page works correctly. Clicking on the error brings up the failing source as it did in Chrome.

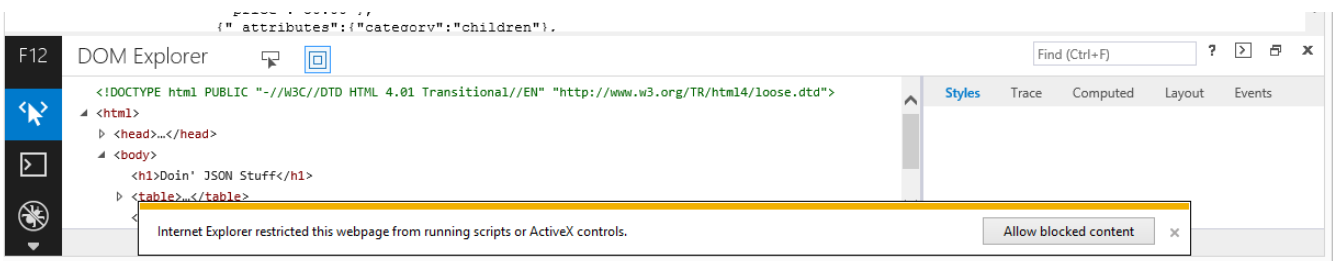
The **debugger works the same in Firefox as in Chrome**. The debugger pane shows source, click on the line number to set a breakpoint, the right of t hee debugger pane shows variables and their values, **the function keys F8, F10 and F11 perform the same actions**.

For you Internet Explorer Users

IE has the *gear* as the menu button in its far right corner. Click it and you'll see:



Select **F12 Developer Tools**. You'll see something like this:

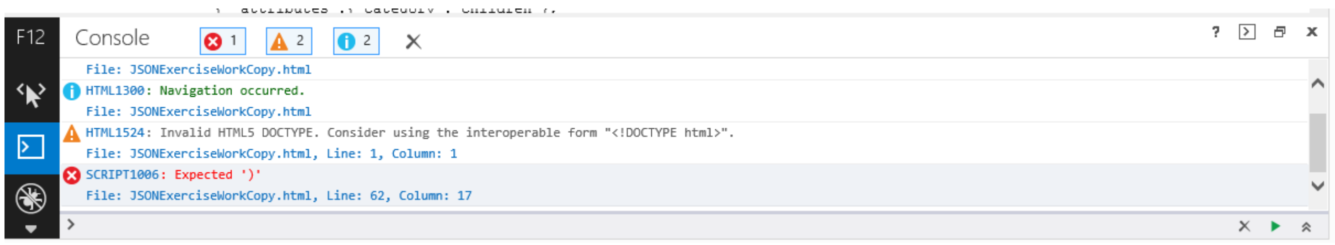


```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html>
  <head>…</head>
  <body>
    <h1>Doin' JSON Stuff</h1>
    <table>…</table>
```

Internet Explorer restricted this webpage from running scripts or ActiveX controls.

IE prompts from time to time about *Active X controls*. Here. Click on *Allow blocked content*. You'll see the screen above without the annoying prompt.

Click on the button to the right. Here's what you'll see:



```
File: JSONExerciseWorkCopy.html
HTML1300: Navigation occurred.
File: JSONExerciseWorkCopy.html
HTML1524: Invalid HTML5 DOCTYPE. Consider using the interoperable form "<!DOCTYPE html>".
File: JSONExerciseWorkCopy.html, Line: 1, Column: 1
SCRIPT1006: Expected ')'
File: JSONExerciseWorkCopy.html, Line: 62, Column: 17
```

IE tells you just about *everything that is wrong* with the page.

Make source code changes, refresh the page, redisplay the developer tools, track down errors until the page displays properly.

The **debugger works the same in IE as in Chrome**. The debugger pane shows source, click on the line number to set a breakpoint, the right of t hee debugger pane shows variables and their values, **the function keys F8, F10 and F11 perform the same actions**.