

# Quick Browserify

## **Copyright Blurb**

All rights reserved. No part of this book may be reproduced in any form or by any electronic or mechanical means including information storage and retrieval systems – except in the case of brief quotations in articles or reviews – without the permission in writing from its publisher, **Sandeep Kumar Patel**.

All brand names and product names used in this book are trademarks, registered trademarks, or trade names of their respective holders. I am not associated with any product or vendor in this book.

## **Published By**

**Sandeep Kumar Patel.**

# Table of Content

<b>Getting Started with Browserify</b>	<b>4</b>
What is Browserify	4
Browserify components	4
Installing Browserify	5
Creating a basic bundle using browserify	5
Summary	8
<b>Export Node.js Method</b>	<i>Error! Bookmark not defined.</i>
Introduction	Error! Bookmark not defined.
CommonJS Export Methods	Error! Bookmark not defined.
Summary	Error! Bookmark not defined.
<b>Working with Watchify</b>	<i>Error! Bookmark not defined.</i>
Introduction	Error! Bookmark not defined.
What is Watchify	Error! Bookmark not defined.
Watchify Installation	Error! Bookmark not defined.
Creating a development workflow using watchify	Error! Bookmark not defined.
Summary	Error! Bookmark not defined.
<b>Automation with Beefy</b>	<i>Error! Bookmark not defined.</i>
Introduction	Error! Bookmark not defined.
What is Beefy	Error! Bookmark not defined.
Beefy automation	Error! Bookmark not defined.
Summary	Error! Bookmark not defined.
<b>Browserify with Babel is Babelify</b>	<i>Error! Bookmark not defined.</i>
Introduction	Error! Bookmark not defined.
What is Babelify	Error! Bookmark not defined.
Babelify with gulp browserify	Error! Bookmark not defined.
Summary	Error! Bookmark not defined.
<b>Bundling Common Module</b>	<i>Error! Bookmark not defined.</i>
Introduction	Error! Bookmark not defined.

Browserify common JavaScript File \_\_\_\_\_ Error! Bookmark not defined.

Summary \_\_\_\_\_ Error! Bookmark not defined.

***Browserify with Grunt*** \_\_\_\_\_ *Error! Bookmark not defined.*

Introduction \_\_\_\_\_ Error! Bookmark not defined.

Grunt Task for Browserify \_\_\_\_\_ Error! Bookmark not defined.

Summary \_\_\_\_\_ Error! Bookmark not defined.

**About The Author** \_\_\_\_\_ Error! Bookmark not defined.

One Last Thing... \_\_\_\_\_ 9

# Getting Started with Browserify

## What is Browserify

The **require()** method is so popular and used by **NodeJS** developer to **load modules**. **Browserify** provides the feature of using **require()** method for the **client side** JavaScript and bundling up of all dependencies together. In other words, Browserify provides a **way** to use the **NPM modules** published for NodeJS server side programming, in the **client/browser environment**.

Browserify brings a new area of application development where code base can be shared to both server-client and termed as **Isomorphic JavaScript**. In this chapter, we will learn to install Browserify and checkout the usage by a simple example where we will try to browserify a NPM module.

*The home page for Browserify module is as follows: -*  
<http://browserify.org>

## Browserify components

The following diagram shows the main components of browserify process to generate browser specific code: -



The details of the previous diagram are as follows: -

- **Entry file(s):** The entry files represent a single file or list of file name which needs to be browserify to generate bundle files.

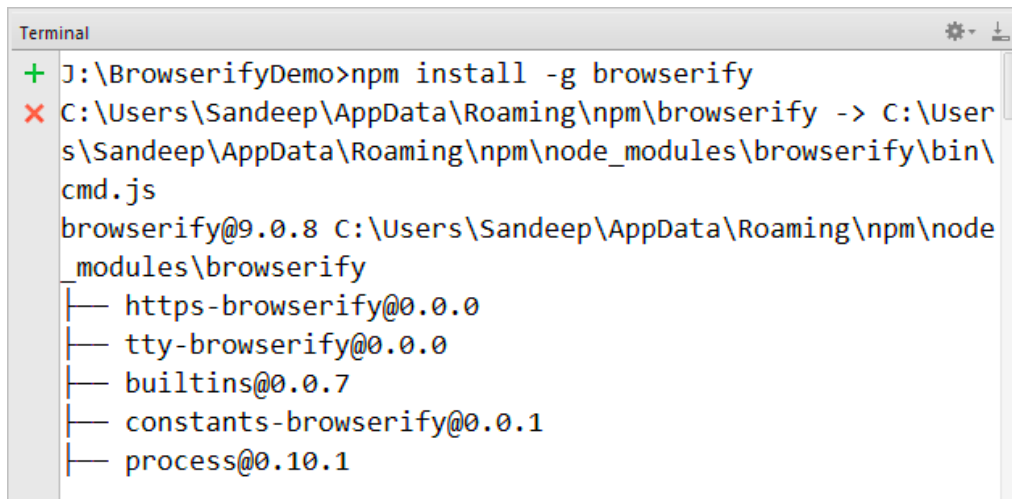
- **Bundle file(s):** The bundle files represent the browserify version of entry files that can be used directly by browsers.

## Installing Browserify

We need to have **NPM -Node package manager** installed in prior to installation of Browserify. If NPM is not installed in your machine go and download the NPM from NodeJS home page. To **install** Browserify use the following command in the terminal : -

```
npm install -g browserify
```

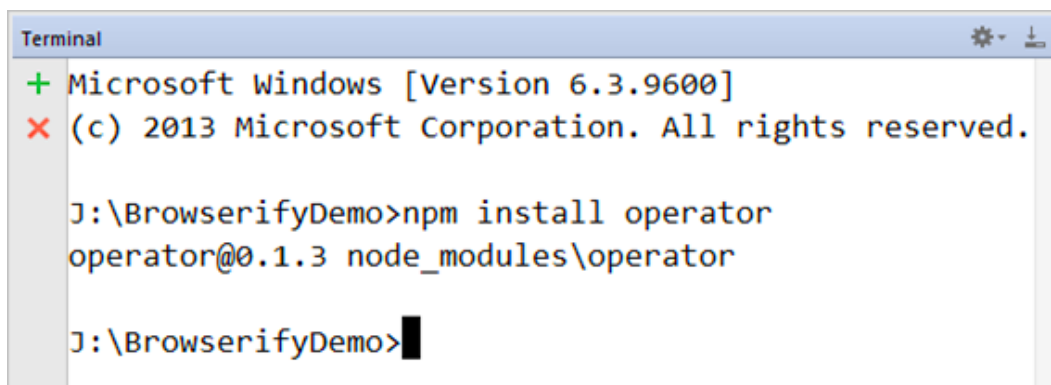
The following **screenshot** shows the **terminal** with Browserify **installation** in progress.

A screenshot of a Windows terminal window titled "Terminal". The command prompt shows the execution of "npm install -g browserify". The output indicates the installation path for browserify and lists its dependencies: https-browserify, tty-browserify, builtins, constants-browserify, and process.

```
Terminal
+ J:\BrowserifyDemo>npm install -g browserify
x C:\Users\Sandeep\AppData\Roaming\npm\browserify -> C:\Users\Sandeep\AppData\Roaming\npm\node_modules\browserify\bin\cmd.js
browserify@9.0.8 C:\Users\Sandeep\AppData\Roaming\npm\node_modules\browserify
├─ https-browserify@0.0.0
├─ tty-browserify@0.0.0
├─ builtins@0.0.7
├─ constants-browserify@0.0.1
└─ process@0.10.1
```

## Creating a basic bundle using browserify

To demonstrate a sample use of **Browserify** we have installed a node module called operator which provides basic mathematical operations like addition, subtraction etc. The following screenshot shows the installation operator npm module in a terminal.

A screenshot of a Windows terminal window titled "Terminal". The command prompt shows the execution of "npm install operator". The output indicates the installation path for the operator module.

```
Terminal
+ Microsoft Windows [Version 6.3.9600]
x (c) 2013 Microsoft Corporation. All rights reserved.

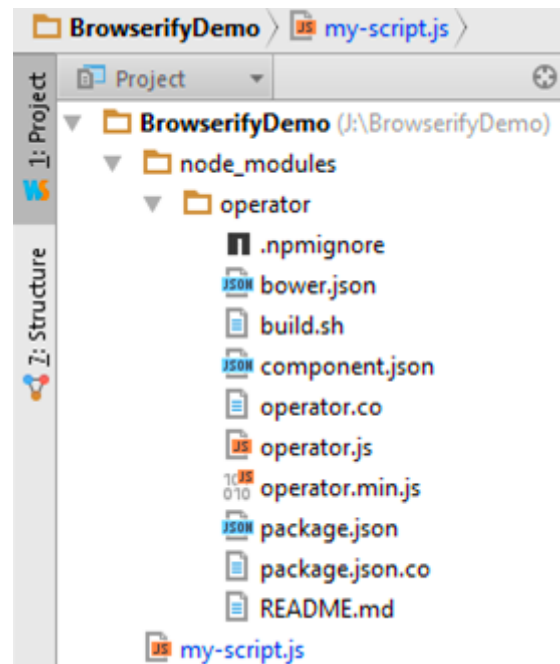
J:\BrowserifyDemo>npm install operator
operator@0.1.3 node_modules\operator

J:\BrowserifyDemo>
```

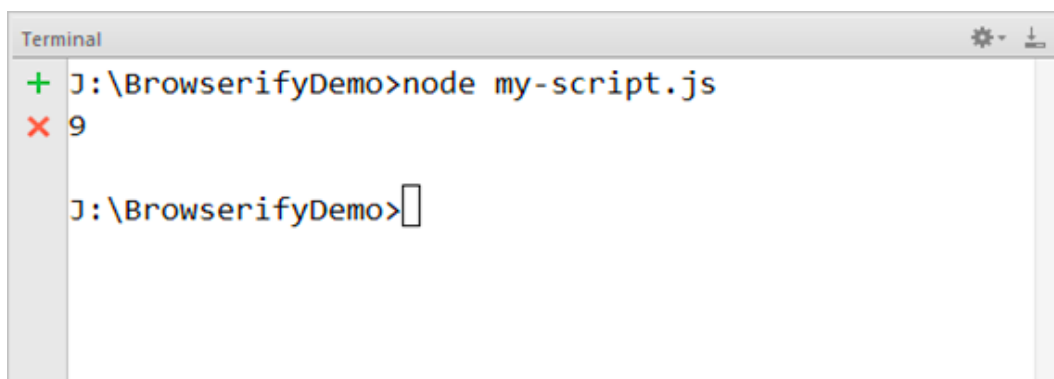
We have created **my-script.js** which uses the operator module to find out the **addition** of two number **5** and **4** and **logs** the result in the **console**. The following code shows the content of **my-script.js**.

```
var operator = require('operator'),  
    result = operator.add(5,4);  
console.log(result);
```

For the reference the following screenshot shows the updated file structure for this demo.



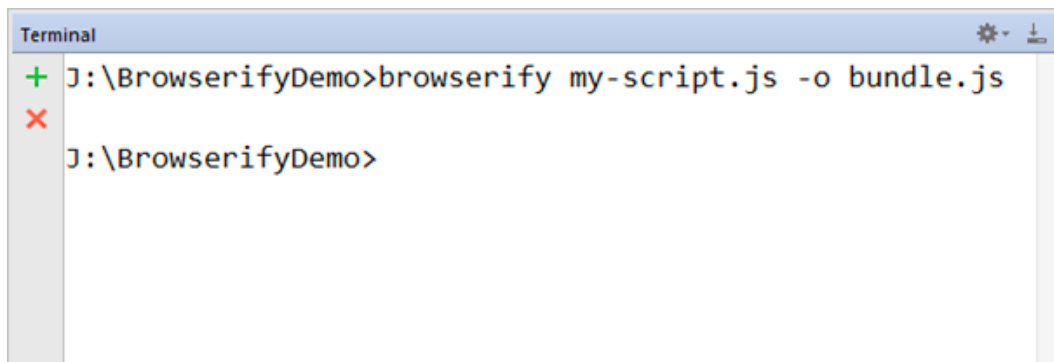
Now we can run the **my-script.js** file using **node** command. The following screenshot shows **execution** of **node** command on **my-script.js** file.



The previous screenshot shows the **my-script.js** runs perfectly in NodeJS environment. Now it is time for the magical Browserify. To generate a code bundle file, use the following command: -

```
browserify my-script.js -o bundle.js
```

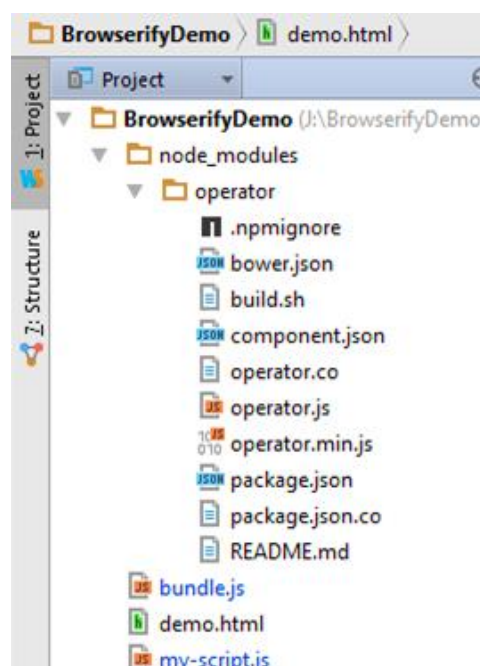
The following screenshot shows the terminal with Browserify command generating **bundle.js** from **my-script.js** which can be used by the client/browser environment.



```
Terminal
+ J:\BrowserifyDemo>browserify my-script.js -o bundle.js
X
J:\BrowserifyDemo>
```

*The -o option is used with browserify command to refer the file name of generated bundled file.*

Now we can create a **demo.html** file and include the generated **bundle.js** file in a **script** element. The **updated** file structure is as follows.

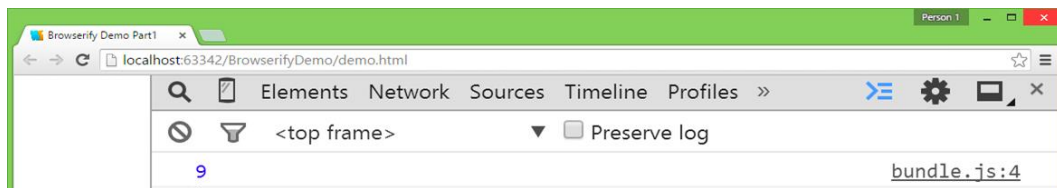


The following code shows the content of **demo.html** file which includes the generated **bundle.js** file.

```
<!DOCTYPE html>
<html>
<head lang="en">
  <meta charset="UTF-8">
  <title>Browserify Demo</title>
```

```
</head>
<body>
  <h1></h1>
  <script src="bundle.js"></script>
</body>
</html>
```

The following **screenshot** shows the **output** of the **demo.html** in the **browser** environment. We can clearly see the log message in **chrome developer console** which results **9** from **addition** of **5** and **4**.



## Summary

In this chapter we have learn the basics of Browserify to bundle the code that can work in the browser environment. In the coming chapter we will start to explore the new features provided by Browserify.

*The chapter1 code can be downloaded from the following URL: -*  
<https://github.com/saan1984/BrowserifyDemo>



## One Last Thing...

When you turn the page, Kindle will give you the opportunity to rate this book and share your thoughts on Facebook and Twitter. If you believe the book is worth sharing, please would you take a few seconds to let your friends know about it? If it turns out to make a difference in their professional lives, they'll be forever grateful to you, as will I.

All the best,

*Sandeep Kumar Patel.*