# \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**CSC174 server-side javascript**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Unit 08 LAB: Add tracking to Wake tech credit union web site

# Objectives

In this lab assignment, students will learn:

* Use Node.js built-in http module to create networked applications
* Use third-party connect module to create simple web applications and web servers
* Understand the concept of Node.js middleware frameworks
* Deepen their understanding of NPM system

# COURSE PREPARATION

You should have done your reading assignment listed under “Reading Assignment” and “Video Assignment” sections in BlackBoard. You should also have reviewed the lecture slides in BlackBoard. There is an optional section called “In Case You Don’t Know” in BlackBoard for those who have limited exposure to JavaScript language.

# WHat to submit

For this lab you need to submit the following files:

* **app.js**

# grading rubric:

Be sure to follow the Coding Standard Guidelines. You must properly indent and comment your code. This assignment is worth 100 points.

* Indent code and insert comments to document your program. [10 pts]
* Program must be implemented and run with no syntax errors. [40 pts]
* Program must be implemented and run with no logic errors. [40 pts]
* Required source files should be zipped and uploaded to BlackBoard assignment drop box before the deadline. [10 points]

# \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

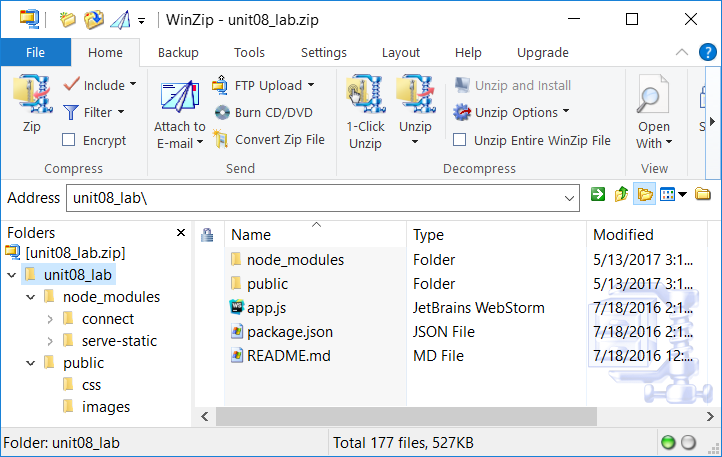
**Add Tracking to a static Wake Tech Credit Union Web Site**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

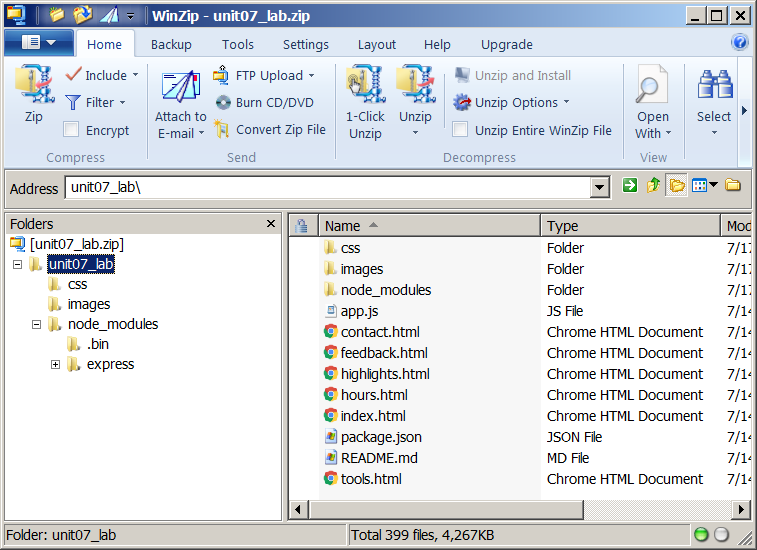
**Project Description:**

You are given a static Wake Tech Credit Union web site made by Node.js. It is very similar to the program given in the past few labs except that the instructor took away the feedback page. All the content of this site is static.

The instructor should provide you the source code in ***unit08\_lab.zip*** file in BlackBoard Unit 08 Lab sections. There are many files and folders in this project seen below:



You can compare this project file structure with the one from Unit 07 Lab project.



You may see some differences. Lab 08 is built on ***connect*** and ***serve-static*** modules while Lab 07 is built on ***express*** module.

Your job is to modify app.js file in this project so that your Node.js web server will log all client access requests to a text file called **access.log**. The format of the log file is:

datetime | user-agent | browser language | original url requested

Here are some sample log entries from access.log file:

Mon Jul 18 2016 15:46:24 GMT-0400 (Eastern Daylight Time)|Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/51.0.2704.103 Safari/537.36|en-US,en;q=0.8,es;q=0.6|/

Mon Jul 18 2016 15:47:12 GMT-0400 (Eastern Daylight Time)|Mozilla/5.0 (Windows NT 6.1; WOW64; Trident/7.0; rv:11.0) like Gecko|en-US,es;q=0.8,zh-CN;q=0.5,en;q=0.3|/images/career.jpg

Mon Jul 18 2016 15:46:47 GMT-0400 (Eastern Daylight Time)|Mozilla/5.0 (Windows NT 6.1; WOW64; rv:47.0) Gecko/20100101 Firefox/47.0|en-US,en;q=0.5|/css/interface.css

The above entries show three access attempts to our web site. The first one is from a Chrome browser which accepts only en-US and en languages. The requested URL is “/”. The second one is from an IE browser which accepts en-US, es, and zh-CN languages. The requested URL is “/images/career.jps”. The last one is from a FireFox browser which accepts en-US and en languages. The requested URL is “/css/interface.css”.

We can write a small program to analyze this log file and show the daily hits of each page of our web site. We can also tally the most popular browsers used to access our web site.

You need to submit only the modified app.js file.

**Hint:**

1. You need to require the ‘fs’ module and create a file write stream to write the log to access.log.

Example: var ws = fs.createWriteStream(‘access.log’, {‘flags’: ‘a’}); //append

1. You can re-use the existing custom middleware called “logit” or create a new one.
2. You can get user agent, browser language and requested URL by doing this:

var reqHeader = req.headers;

var userAgent = reqHeader[‘user-agent’];

var browserLanguage = reqHeader[‘accept-language’];

var requestedUrl = req.originalUrl;

**Note:**

It will be beneficial for you to study the official documentation of ***connect*** and ***serve-static*** module:

<https://www.npmjs.com/package/connect>

<https://www.npmjs.com/package/serve-static>