

## Git + NodeJS + MongoDB + ExpressJS

- Install Git.
- Install Node JS

### Node.js = Runtime Environment + JavaScript Library

```
1. C:\WINDOWS\system32>node --version
2. v5.10.0
3.
4. C:\WINDOWS\system32>node
5. > var x=10;
6. var x=10;
7. undefined
8. > var y=20;
9. undefined
10. > x
11. 10
12. > y
13. 20
14. > x+y
15. 30
16. >
```

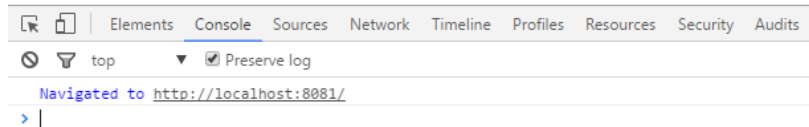
#### staticPage.js

```
1. var http = require("http");
2. http.createServer(function (request, response) {
3.
4.     // Send the HTTP header
5.     // HTTP Status: 200 : OK
6.     // Content Type: text/plain
7.     response.writeHead(200, {'Content-Type': 'text/plain'});
8.
9.     // Send the response body as "Hello World"
10.    response.end('welcome to NodeJS World!\n');
11. }).listen(8081);
12.
13. // Console will print the message
14. console.log('Node Server running at http://127.0.0.1:8081/');
15. console.log('welcome to NodeJS World!');
```

Running staticPage.js using node server, before that, we need to install modules using npn, like:

```
1. C:\Program Files\nodejs\node_modules\npm\node_modules>npm install connect
2. npm@3.8.3 C:\Program Files\nodejs\node_modules\npm
3.   -- connect@3.4.1
4.
5.
6. C:\Program Files\nodejs\node_modules\npm\node_modules>npm install server
7. npm@3.8.3 C:\Program Files\nodejs\node_modules\npm
8. +-- connect@3.4.1
9.   -- server@0.0.3 extraneous
10.
11.
12. C:\Program Files\nodejs\node_modules\npm\node_modules>npm install http
13. npm@3.8.3 C:\Program Files\nodejs\node_modules\npm
14. +-- connect@3.4.1
15. +-- http@0.0.0 extraneous
16.   -- server@0.0.3 extraneous
17.
18.
19. C:\Program Files\nodejs\node_modules\npm\node_modules>node server
20.
21. C:\Program Files\nodejs\node_modules\npm>E:
22.
```

```
23. E:\work\GitHub\Projects\nodeJSApps>node staticPage.js
24. Server running at http://127.0.0.1:8081/
25. welcome to NodeJS World!
```



- Install MongoDB

```
1. C:\WINDOWS\system32>node --version
2. v5.10.0
3.
4. C:\WINDOWS\system32>mongod --directoryperdb --dbpath E:\work\softwares\MongoDB\Server\3.2\data\db --
  logpath E:\work\softwares\MongoDB\Server\3.2\log\mongodb.log --logappend --rest --service --httpinterface --
  replSet "rs0" --install
5.
6. C:\WINDOWS\system32>net start mongoDB
7.
8. The MongoDB service was started successfully.
9.
10.
11. C:\WINDOWS\system32>mongo
12. MongoDB shell version: 3.2.4
13. connecting to: test
14. > rs.initiate()
15. {
16.   "info2" : "no configuration specified. Using a default configuration for the set",
17.   "me" : "thanooj:27017",
18.   "ok" : 1
19. }
20. rs0:SECONDARY> rs.conf()
21. {
22.   "_id" : "rs0",
23.   "version" : 1,
24.   "protocolVersion" : NumberLong(1),
25.   "members" : [
26.     {
27.       "_id" : 0,
28.       "host" : "thanooj:27017",
29.       "arbiterOnly" : false,
30.       "buildIndexes" : true,
31.       "hidden" : false,
32.       "priority" : 1,
33.       "tags" : {
34.
35.       },
36.       "slaveDelay" : NumberLong(0),
37.       "votes" : 1
38.     }
39.   ],
40.   "settings" : {
41.     "chainingAllowed" : true,
42.     "heartbeatIntervalMillis" : 2000,
43.     "heartbeatTimeoutSecs" : 10,
44.     "electionTimeoutMillis" : 10000,
45.     "getLastErrorModes" : {
```

```

46.
47.         },
48.         "getLastErrorDefaults" : {
49.             "w" : 1,
50.             "wtimeout" : 0
51.         },
52.         "replicaSetId" : ObjectId("5700cbbf631f200f57c27ddc")
53.     }
54. }
55. rs0:PRIMARY> rs.status()
56. {
57.     "set" : "rs0",
58.     "date" : ISODate("2016-04-03T07:53:01.361Z"),
59.     "myState" : 1,
60.     "term" : NumberLong(1),
61.     "heartbeatIntervalMillis" : NumberLong(2000),
62.     "members" : [
63.         {
64.             "_id" : 0,
65.             "name" : "thanooj:27017",
66.             "health" : 1,
67.             "state" : 1,
68.             "stateStr" : "PRIMARY",
69.             "uptime" : 123,
70.             "optime" : {
71.                 "ts" : Timestamp(1459669953, 1),
72.                 "t" : NumberLong(1)
73.             },
74.             "optimeDate" : ISODate("2016-04-03T07:52:33Z"),
75.             "infoMessage" : "could not find member to sync from",
76.             "electionTime" : Timestamp(1459669952, 2),
77.             "electionDate" : ISODate("2016-04-03T07:52:32Z"),
78.             "configVersion" : 1,
79.             "self" : true
80.         }
81.     ],
82.     "ok" : 1
83. }
84. rs0:PRIMARY> db.createCollection("cust");
85. { "ok" : 1 }
86. rs0:PRIMARY> show collections;
87. cust
88. rs0:PRIMARY> use mydb;
89. switched to db mydb
90. rs0:PRIMARY> db.createCollection("cust");
91. { "ok" : 1 }
92. rs0:PRIMARY> show collections;
93. cust

1. rs0:PRIMARY> db.cust.insert({id:1,name:'sriram',loction:'ayodya'})
2. WriteResult({ "nInserted" : 1 })
3. rs0:PRIMARY> db.cust.find();
4. { "_id" : ObjectId("5700cca43133bfac6bf15d07"), "id" : 1, "name" : "sriram", "loction" : "ayodya" }
5. rs0:PRIMARY>
6. rs0:PRIMARY> db.cust.find().pretty();
7. {
8.     "_id" : ObjectId("5700cca43133bfac6bf15d07"),
9.     "id" : 1,
10.    "name" : "sriram",
11.    "loction" : "ayodya"
12. }
13. rs0:PRIMARY> db.cust.insert({id:2,name:'lakshman',loction:'ayodya'})
14. WriteResult({ "nInserted" : 1 })
15. rs0:PRIMARY> db.cust.find().pretty();
16. {
17.     "_id" : ObjectId("5700cca43133bfac6bf15d07"),
18.     "id" : 1,
19.     "name" : "sriram",
20.     "loction" : "ayodya"
21. }
22. {
23.     "_id" : ObjectId("5700cf5f3133bfac6bf15d08"),
24.     "id" : 2,
25.     "name" : "lakshman",
26.     "loction" : "ayodya"
27. }

```

```

28. rs0:PRIMARY> db.cust.findOne({id:1})
29. {
30.     "_id" : ObjectId("5700cca43133bfac6bf15d07"),
31.     "id" : 1,
32.     "name" : "sriram",
33.     "loction" : "ayodya"
34. }
35. rs0:PRIMARY> db.cust.findOne({id:2})
36. {
37.     "_id" : ObjectId("5700cf5f3133bfac6bf15d08"),
38.     "id" : 2,
39.     "name" : "lakshman",
40.     "loction" : "ayodya"
41. }

```

```

1. rs0:PRIMARY> db.cust.update({id:1},{id:1,name:"raghuram",{upsert:true}});
2. WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
3. rs0:PRIMARY> db.cust.find().pretty();
4. {
5.     "_id" : ObjectId("5700d4c93133bfac6bf15d0b"),
6.     "id" : 1,
7.     "name" : "raghuram"
8. }
9. {
10.    "_id" : ObjectId("5700d4d53133bfac6bf15d0c"),
11.    "id" : 2,
12.    "name" : "lakshman",
13.    "loction" : "ayodhya"
14. }
15. rs0:PRIMARY> db.cust.update({id:1},{id:1,name:"raghuram",locations:"ayodhya",{upsert:true}});
16. WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
17. rs0:PRIMARY> db.cust.find().pretty();
18. {
19.     "_id" : ObjectId("5700d4c93133bfac6bf15d0b"),
20.     "id" : 1,
21.     "name" : "raghuram",
22.     "locations" : "ayodhya"
23. }
24. {
25.     "_id" : ObjectId("5700d4d53133bfac6bf15d0c"),
26.     "id" : 2,
27.     "name" : "lakshman",
28.     "loction" : "ayodhya"
29. }
30. rs0:PRIMARY> db.cust.insert({id:3,name:'bharatha',loction:'ayodya'})
31. WriteResult({ "nInserted" : 1 })
32. rs0:PRIMARY> db.cust.find().pretty();
33. {
34.     "_id" : ObjectId("5700d4c93133bfac6bf15d0b"),
35.     "id" : 1,
36.     "name" : "raghuram",
37.     "locations" : "ayodhya"
38. }
39. {
40.     "_id" : ObjectId("5700d4d53133bfac6bf15d0c"),
41.     "id" : 2,
42.     "name" : "lakshman",
43.     "loction" : "ayodhya"
44. }
45. {
46.     "_id" : ObjectId("5700d60b3133bfac6bf15d0d"),
47.     "id" : 3,
48.     "name" : "bharatha",
49.     "loction" : "ayodya"
50. }
51. rs0:PRIMARY> db.cust.remove({id:3});
52. WriteResult({ "nRemoved" : 1 })
53. rs0:PRIMARY> db.cust.find().pretty();
54. {

```

```

55.         "_id" : ObjectId("5700d4c93133bfac6bf15d0b"),
56.         "id" : 1,
57.         "name" : "raghuras",
58.         "locations" : "ayodhya"
59.     }
60. {
61.         "_id" : ObjectId("5700d4d53133bfac6bf15d0c"),
62.         "id" : 2,
63.         "name" : "lakshman",
64.         "location" : "ayodhya"
65.     }
66. rs0:PRIMARY> db.cust.remove({});
67. WriteResult({ "nRemoved" : 2 })
68. rs0:PRIMARY> db.cust.find().pretty();
69. rs0:PRIMARY>

```

- Non-Blocking Code Example

```

1. C:\Program Files\nodejs\node_modules\npm\node_modules>npm install fs
2. npm@3.8.3 C:\Program Files\nodejs\node_modules\npm
3. +-- connect@3.4.1
4. +-- express@4.13.4 extraneous
5. +-- fs@0.0.2 extraneous
6. +-- http@0.0.0 extraneous
7. `-- server@0.0.3 extraneous
8.
9.
10. C:\Program Files\nodejs\node_modules\npm\node_modules>E:
11.
12. E:\work\GitHub\Projects\nodeJSApps>dir
13. Volume in drive E is SPACE
14. Volume Serial Number is BAD5-CB77
15.
16. Directory of E:\work\GitHub\Projects\nodeJSApps
17.
18. 04/03/2016 03:32 PM <DIR>      .
19. 04/03/2016 03:32 PM <DIR>      ..
20. 04/03/2016 03:32 PM          16 cust.csv
21. 04/03/2016 03:32 PM        191 fs.js
22. 04/03/2016 02:57 PM <DIR>      node_modules
23. 04/03/2016 03:17 PM        490 staticPage.js
24. 04/03/2016 02:56 PM <DIR>      test
25.                3 File(s)        697 bytes
26.                4 Dir(s) 165,199,478,784 bytes free
27.
28. E:\work\GitHub\Projects\nodeJSApps>node fs.js
29. NodeJS fs module.
30. 1,sriram,ayodhya
31.
32. E:\work\GitHub\Projects\nodeJSApps>

```

- Express

## Express Overview

Express is a minimal and flexible Node.js web application framework that provides a robust set of features to develop web and mobile applications. It facilitates a rapid development of Node based Web applications. Following are some of the core features of Express framework:

- Allows setting up middleware's to respond to HTTP Requests.
- Defines a routing table which is used to perform different action based on HTTP Method and URL.
- Allows to dynamically rendering HTML Pages based on passing arguments to templates.

## Installing Express

```
1. C:\Program Files\nodejs\node_modules\npm\node_modules>npm install express
2. npm@3.8.3 C:\Program Files\nodejs\node_modules\npm
3. +-- connect@3.4.1
4. +-- express@4.13.4 extraneous
5. +-- http@0.0.0 extraneous
6. +-- server@0.0.3 extraneous
```

There are following important modules which you should install along with express:

- body-parser - This is a node.js middleware for handling JSON, Raw, Text and URL encoded form data.
- cookie-parser - Parse Cookie header and populate req.cookies with an object keyed by the cookie names.
- multer - This is a node.js middleware for handling multipart/form-data.

```
1. E:\work\GitHub\Projects\nodeJSApps>npm install express --save
2. npm WARN saveError ENOENT: no such file or directory, open 'E:\work\GitHub\Projects\nodeJSApps\package.json'
3. E:\work\GitHub\Projects\nodeJSApps
4. +-- express@4.13.4
5.   +-- accepts@1.2.13
6.     | +-- mime-types@2.1.10
7.     | | +-- mime-db@1.22.0
8.     | | +-- negotiator@0.5.3
9.     +-- array-flatten@1.1.1
10.    +-- content-disposition@0.5.1
11.    +-- content-type@1.0.1
12.    +-- cookie@0.1.5
13.    +-- cookie-signature@1.0.6
14.    +-- debug@2.2.0
15.    | +-- ms@0.7.1
16.    +-- depd@1.1.0
17.    +-- escape-html@1.0.3
18.    +-- etag@1.7.0
19.    +-- finalhandler@0.4.1
20.    | +-- unpipe@1.0.0
21.    +-- fresh@0.3.0
22.    +-- merge-descriptors@1.0.1
23.    +-- methods@1.1.2
24.    +-- on-finished@2.3.0
25.    | +-- ee-first@1.1.1
26.    +-- parseurl@1.3.1
27.    +-- path-to-regexp@0.1.7
28.    +-- proxy-addr@1.0.10
29.    | +-- forwarded@0.1.0
30.    | +-- ipaddr.js@1.0.5
31.    +-- qs@4.0.0
32.    +-- range-parser@1.0.3
33.    +-- send@0.13.1
34.    | +-- destroy@1.0.4
35.    | +-- http-errors@1.3.1
36.    | | +-- inherits@2.0.1
37.    | | +-- mime@1.3.4
38.    | | +-- statuses@1.2.1
39.    +-- serve-static@1.10.2
40.    +-- type-is@1.6.12
41.    | +-- media-typer@0.3.0
42.    +-- utils-merge@1.0.0
43.    +-- vary@1.0.1
44.
45. npm WARN enoent ENOENT: no such file or directory, open 'E:\work\GitHub\Projects\nodeJSApps\package.json'
46. npm WARN nodeJSApps No description
47. npm WARN nodeJSApps No repository field.
48. npm WARN nodeJSApps No README data
49. npm WARN nodeJSApps No license field.
50.
51. E:\work\GitHub\Projects\nodeJSApps>dir
52. Volume in drive E is SPACE
53. Volume Serial Number is BAD5-CB77
54.
55. Directory of E:\work\GitHub\Projects\nodeJSApps
56.
57. 04/03/2016  04:34 PM    <DIR>          .
58. 04/03/2016  04:34 PM    <DIR>          ..
59. 04/03/2016  03:32 PM                16 cust.csv
60. 04/03/2016  03:48 PM             595 expressOne.js
61. 04/03/2016  03:32 PM             191 fs.js
62. 04/03/2016  04:34 PM    <DIR>          node_modules
63. 04/03/2016  03:17 PM             490 staticPage.js
```

```

64.          4 File(s)          1,292 bytes
65.          3 Dir(s)  165,150,875,648 bytes free
66.
67. E:\work\GitHub\Projects\nodeJSApps>npm install body-parser --save
68. npm WARN saveError ENOENT: no such file or directory, open 'E:\work\GitHub\Projects\nodeJSApps\package.json'
69. E:\work\GitHub\Projects\nodeJSApps
70. `-- body-parser@1.15.0
71.    +-- bytes@2.2.0
72.    +-- http-errors@1.4.0
73.    +-- iconv-lite@0.4.13
74.    +-- qs@6.1.0
75.    `-- raw-body@2.1.6
76.       `-- bytes@2.3.0
77.
78. npm WARN enoent ENOENT: no such file or directory, open 'E:\work\GitHub\Projects\nodeJSApps\package.json'
79. npm WARN nodeJSApps No description
80. npm WARN nodeJSApps No repository field.
81. npm WARN nodeJSApps No README data
82. npm WARN nodeJSApps No license field.
83.
84. E:\work\GitHub\Projects\nodeJSApps>npm install multer --save
85. npm WARN saveError ENOENT: no such file or directory, open 'E:\work\GitHub\Projects\nodeJSApps\package.json'
86. E:\work\GitHub\Projects\nodeJSApps
87. `-- multer@1.1.0
88.    +-- append-field@0.1.0
89.    +-- busboy@0.2.13
90.    | +-- dicer@0.2.5
91.    | | `-- streamsearch@0.1.2
92.    | | `-- readable-stream@1.1.13
93.    | | +-- core-util-is@1.0.2
94.    | | +-- isarray@0.0.1
95.    | | `-- string_decoder@0.10.31
96.    +-- concat-stream@1.5.1
97.    | +-- readable-stream@2.0.6
98.    | | +-- isarray@1.0.0
99.    | | +-- process-nextick-args@1.0.6
100.    | | `-- util-deprecate@1.0.2
101.    | `-- typedarray@0.0.6
102.    +-- mkdirp@0.5.1
103.    | `-- minimist@0.0.8
104.    +-- object-assign@3.0.0
105.    `-- xtend@4.0.1
106.
107. npm WARN enoent ENOENT: no such file or directory, open 'E:\work\GitHub\Projects\nodeJSApps\package.json'
108. npm WARN nodeJSApps No description
109. npm WARN nodeJSApps No repository field.
110. npm WARN nodeJSApps No README data
111. npm WARN nodeJSApps No license field.

```

## expressOne.js

```

1.  var express = require('express');
2.  var app = express();
3.  var fs = require("fs");
4.  var cust_data;
5.  app.get('/', function (req, res) {
6.    res.send('Welcome Express World!');
7.  })
8.
9.  fs.readFile('cust.csv', function (err, data) {
10.    if (err) return console.error(err);
11.    cust_data = data.toString();
12.    console.log(data.toString());
13.  });
14.  app.get('/cust', function (req, res) {
15.    res.send(cust_data);
16.  })
17.
18.  var server = app.listen(8081, function () {
19.
20.    var host = server.address().address
21.    var port = server.address().port
22.
23.    console.log("Express, listening at http://%s:%s", host, port)
24.  })
25.
26. E:\work\GitHub\Projects\nodeJSApps>dir
27. Volume in drive E is SPACE
28. Volume Serial Number is BAD5-CB77
29.
30. Directory of E:\work\GitHub\Projects\nodeJSApps
31.
32. 04/03/2016  04:34 PM    <DIR>      .

```

```
33. 04/03/2016 04:34 PM <DIR> ..
34. 04/03/2016 03:32 PM      16 cust.csv
35. 04/03/2016 03:48 PM      595 expressOne.js
36. 04/03/2016 03:32 PM      191 fs.js
37. 04/03/2016 04:34 PM <DIR> node_modules
38. 04/03/2016 03:17 PM      490 staticPage.js
39.          4 File(s)      1,292 bytes
40.          3 Dir(s)  165,148,434,432 bytes free
41.
42. E:\work\GitHub\Projects\nodeJSApps>node expressOne.js
43. Express, listening at http://:::8081
44. 1,sriram,ayodhya
```

