




Deployment on Openshift:

Make account on Openshift and create node js project

 or

Matches search 'node' ([show all](#))

 **Node.js 0.10** 

Node.js is a platform built on Chrome's JavaScript runtime for easily building fast, scalable network applications. Node.js is perfect for data-intensive real-time applications that run across distributed devices.

<http://www.nodejs.org/>


☆ OpenShift maintained

🛡️ Receives automatic security updates

JAVASCRIPT NODEJS

Choose the name and press button Create application

Based On

Node.js 0.10 Cartridge 

Node.js is a platform built on Chrome's JavaScript runtime for easily building fast, scalable network applications. Node.js is perfect for data-intensive real-time applications that run across distributed devices.

<http://www.nodejs.org/>

☆ OpenShift maintained

🛡️ Receives automatic security updates

Public URL

OpenShift will automatically register this domain name for your application. You can add your own domain name later.

Cartridges **Node.js 0.10**

Applications are composed of cartridges - each of which exposes a service or capability to your code. All applications must have a web cartridge.



Scaling No scaling ▾

OpenShift automatically routes web requests to your web gear. If you allow your application to scale, we'll set up a load balancer and allocate more gears to handle traffic as you need it.

Region

- ☐ No preference
- ☒ **aws-us-east-1**
All gear sizes can be deployed to the US Region.
- ☐ **aws-eu-west-1**
WARNING: Small gears cannot be deployed to this region. Only production gears can be deployed to the EU Region (small, highcpu, medium, and large).
- ☐ **aws-ap-southeast-2**
WARNING: This region is reserved for Dedicated Node Service
- ☐ **aws-us-west-1**
WARNING: This region is reserved for Dedicated Node Service
- ☐ **aws-eu-central-1**
WARNING: This region is reserved for Dedicated Node Service

Gears within your application will run on servers in the specified region.

Back Create Application +1  

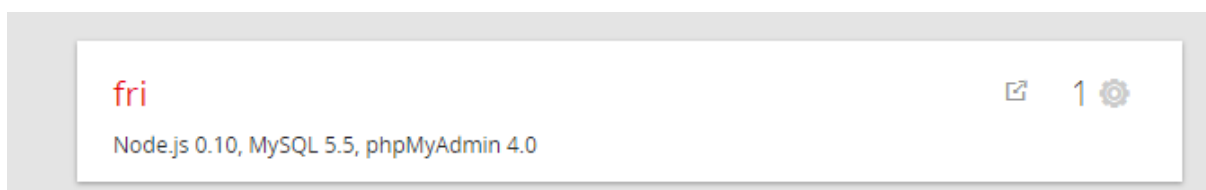
You will get instruction how to connect to it:

```
git clone ssh://587bbd1d89f5cf88c90001c7@fri-
spirala3.rhcloud.com/~/.git/fri.git/
```

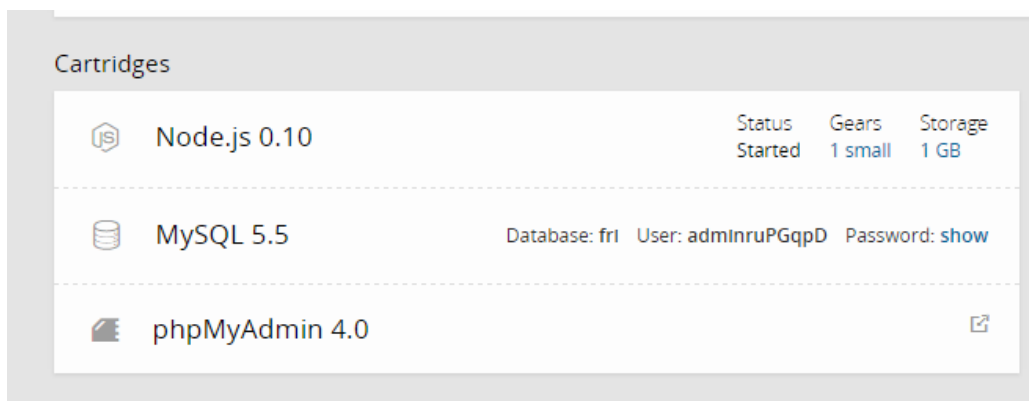
```
cd fri/
```

We will add code later and then push it. Let's deal with database. You can find application on following link:

<https://openshift.redhat.com/app/console/applications>



Add cartridges like on the picture



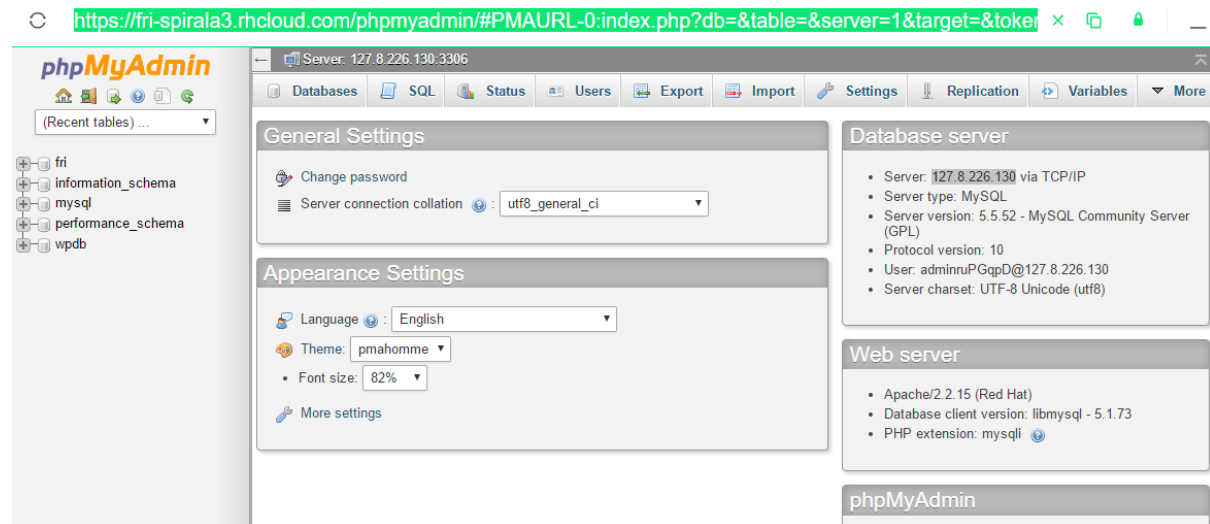
Now you can access database with given credentials

<https://fri-spirala3.rhcloud.com/phpmyadmin>

username: adminruPGqD

password: krp9blgyi9Vs

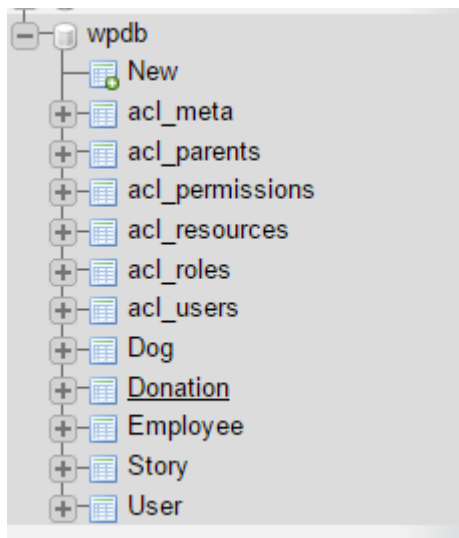
it will look like this



Now import the database phpMyAdmin. Press import and browse file db.sql:

A screenshot of the 'Importing into the current server' page in phpMyAdmin. The top navigation bar is the same as the previous screenshot. The main content area has a title 'Importing into the current server'. Below the title is a section 'File to Import:'. It contains a text box with the instruction: 'File may be compressed (gzip, bzip2, zip) or uncompressed. A compressed file's name must end in .[format].[compression]. Example: .sql.zip'. There are two radio buttons: 'Browse your computer:' and 'Select from the web server upload directory /tmp:'. The 'Browse your computer:' option is selected, and next to it is a 'Choose file' button and the text 'No file chosen (Max: 200MiB)'. Below the radio buttons is a 'Character set of the file:' dropdown set to 'utf-8'. There is a section 'Partial Import:' with a checked checkbox 'Allow the interruption of an import in case the script detects it is close to the PHP timeout limit. (This might be a good way to however it can break transactions.)'. Below this is a text box 'Number of rows to skip, starting from the first row:' with the value '0'. There is a section 'Format:' with a dropdown set to 'SQL'. At the bottom is a section 'Format-Specific Options:'.

The database is will be on the list:



Now we have to change connection string for database since it is not local anymore:

In models/index.js change connection string to the following (format of connection string is: 'mysql://username:password@host:port/db') :

```
var connection = new  
Sequelize('mysql://adminruPGqD:krp9blgyi9Vs@127.8.226.130:3306/wpdb', { omitNull: true });
```

Since we are also not on the same server anymore and openshift generated new server.js script put the code from bin/www inside it with these changes:

```
var app = require('./app');  
  
app.set('port', process.env.OPENSIFT_NODEJS_PORT || process.env.PORT || 8080);  
  
app.set('ip', process.env.OPENSIFT_NODEJS_IP || '127.0.0.1');  
  
var server = http.createServer(app).listen(app.get('port'), app.get('ip'), function(){  
  
  console.log('Express server listening on port ' + app.get('port'));  
  
});
```

```
app.set('port', process.env.OPENSIFT_NODEJS_PORT || process.env.PORT || 8080);  
app.set('ip', process.env.OPENSIFT_NODEJS_IP || '127.0.0.1');  
  
/**  
 * Create HTTP server.  
 */  
console.log("port" + app.get('port') + " ip " + app.get('ip'));  
  
var server = http.createServer(app).listen(app.get('port'), app.get('ip'), function(){  
  console.log('Express server listening on port ' + app.get('port'));  
});  
  
/**
```

Openshift also generated new package.js so add dependencies from our application package.js:

```

11
12   "engines": {
13     "node": ">= 0.6.0",
14     "npm": ">= 1.0.0"
15   },
16   "scripts": {
17     "start": "node ./server.js",
18     "test": "mocha --timeout 10000"
19   },
20   "dependencies": {
21     "acl": "^0.4.9",
22     "acl-sequelize": "^0.1.2",
23     "bcrypt-nodejs": "0.0.3",
24     "body-parser": "~1.15.2",
25     "bootstrap": "^4.0.0-alpha.6",
26     "chai": "^3.5.0",
27     "chai-http": "^2.0.1",
28     "connect-flash": "^0.1.1",
29     "cookie-parser": "~1.4.3",
30     "debug": "~2.2.0",
31     "email-validator": "^1.0.7",
32     "express": "~4.14.0",
33     "express-session": "^1.14.2",
34     "jade": "~1.11.0",
35     "less": "^2.7.2",
36     "mocha": "^3.2.0",
37     "morgan": "~1.7.0",
38     "mysql": "^2.12.0",
39     "passport": "^0.3.2",
40     "passport-local": "^1.0.0",
41     "sequelize": "^3.29.0",
42     "serve-favicon": "~2.3.0",
43     "winston": "^2.3.0"
44   },
45   "devDependencies": {},
46   "bundleDependencies": [],
47
48   "private": true,
49   "main": "server.js"
50 }
51

```

We are done with changes and we can push the code with following commands:

git add .

git commit -m 'My changes'

git push origin master

While pushing you can see if there are some errors and problems while starting application:

```

remote: npm info prepublish OpenShift-Sample-App@1.0.0
remote: npm info ok
remote: Preparing build for deployment
remote: Deployment id is 839d7f86
remote: Activating deployment
remote: Starting MySQL 5.5 cartridge
remote: Starting PHPMyAdmin cartridge
remote: Starting NodeJS cartridge
remote: Sun Jan 15 2017 12:17:16 GMT-0500 (EST): Starting application 'fri' ...
remote: Waiting for application port (8080) become available ...
remote: Application 'fri' failed to start (port 8080 not available)
remote: -----
remote: Git Post-Receive Result: failure
remote: Activation status: failure
remote: Activation failed for the following gears:
remote: 587a671a7628e11a87000215 (Error activating gear: CLIENT_ERROR: Failed to execute: 'control
remote: #<IO:0x00000000189f390>
remote: #<IO:0x00000000189f318>
remote: )
remote: Deployment completed with status: failure
remote: postreceive failed
To ssh://587a671a7628e11a87000215@fri-spirala3.rhcloud.com/~/.git/fri.git/
 8acd643..cf154cd master -> master

```

If you have errors you can see log with the following command:

rhc tail -a fri

And there you can find where is the problem, for example like this:

```

memory, and will not scale past a single process.
/var/lib/openshift/587a671a7628e11a87000215/app-root/runtime/repo/routes/index.js:15
  var tsFormat = () => (new Date()).toLocaleTimeString();
  ^
SyntaxError: Unexpected token )
    at Module._compile (module.js:439:25)
    at Object.Module._extensions..js (module.js:474:10)
    at Module.load (module.js:356:32)
    at Function.Module._load (module.js:312:12)
    at Module.require (module.js:364:17)
    at require (module.js:380:17)
    at Object.<anonymous> (/var/lib/openshift/587a671a7628e11a87000215/app-root/runtime/repo/app.js:32:1)
    at Module._compile (module.js:456:26)
    at Object.Module._extensions..js (module.js:474:10)
    at Module.load (module.js:356:32)
    at Function.Module._load (module.js:312:12)
    at Module.require (module.js:364:17)
    at require (module.js:380:17)
    at Object.<anonymous> (/var/lib/openshift/587a671a7628e11a87000215/app-root/runtime/repo/server.js:7:11)
    at Module._compile (module.js:456:26)
    at Object.Module._extensions..js (module.js:474:10)
    at Module.load (module.js:356:32)
    at Function.Module._load (module.js:312:12)
    at Function.Module.runMain (module.js:497:10)
    at startup (node.js:119:16)

```

Fix the code until you get success:

```

remote: npm info ok
remote: Preparing build for deployment
remote: Deployment id is bd12714d
remote: Activating deployment
remote: Starting MySQL 5.5 cartridge
remote: Starting PHPMyAdmin cartridge
remote: Starting NodeJS cartridge
remote: Sun Jan 15 2017 12:55:09 GMT-0500 (EST): Starting application 'fri' ...
remote: Waiting for application port (8080) become available ...
remote: Found 127.8.226.129:8080 listening port
remote: -----
remote: Git Post-Receive Result: success
remote: Activation status: success
remote: Deployment completed with status: success
To ssh://587a671a7628e11a87000215@fri-spirala3.rhcloud.com/~/.git/fri.git/
 9eff8c2..8ca9729 master -> master
PS D:\FRI\1. semester\Web programming\openshift\fri>

```