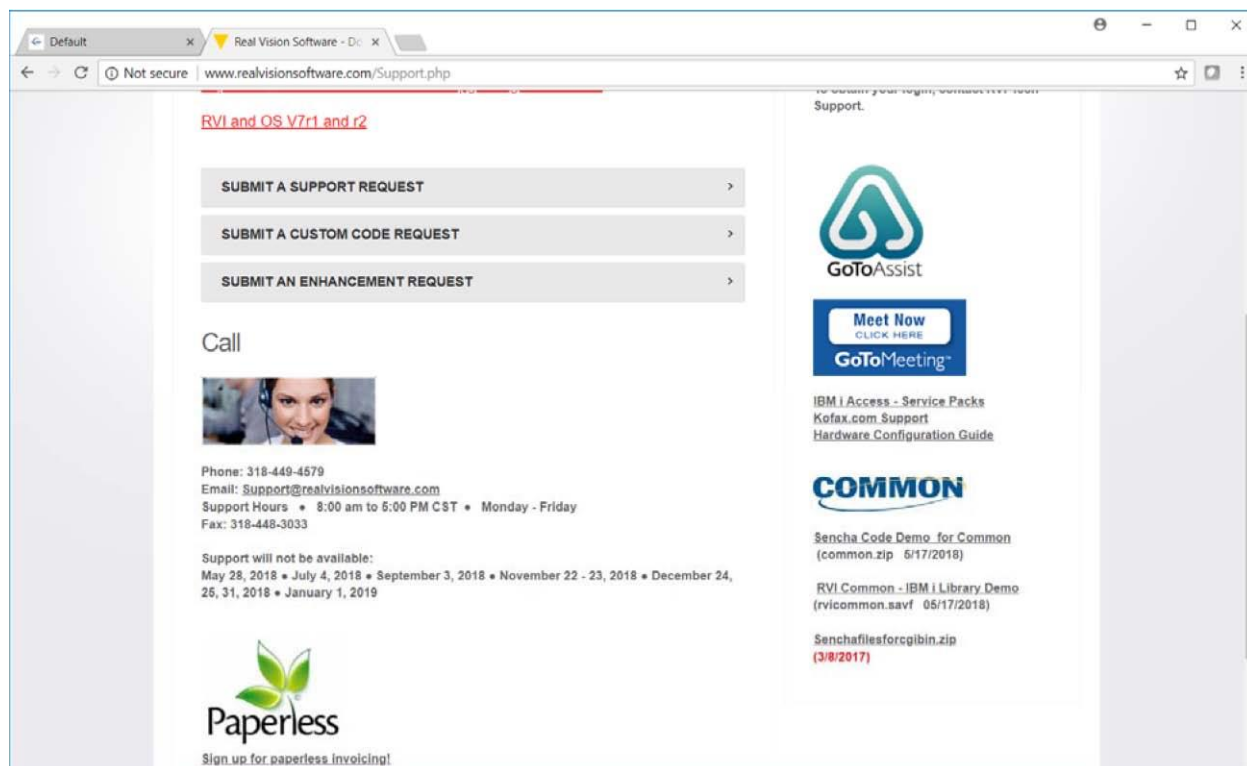


WWW.REALVISIONSOFTWARE.COM



Download the 3 things under Common Logo on the right

Common.Zip File goes on IBM I under /CGIBIN Directory RVICommon.SAVF FTP to IBM I and restore the library RVICOMMON Senchafilesforcgibin.zip Unzip this and put it under the /CGIBIN directory on IBM I Create Http Configuration with access to RVICOMMON library as example

shows...

Once completed you will now have several versions of the RVI Demo application to choose from.

Version 1

To run the RVI Demo version from Common 2018 you can open a web browser and put in:

<http://yourip:yourport/internet/common/rvidemo/app/index.html>

NOTE: To Create Your OutQue Files on demo program Addliblible RVICOMMON
Call MAKEWSPLXX

Version 2.1

To run the JsonP version of the RVI Demo from Common 2019 you can open a web browser and put in:

<http://yourip:yourport/internet/common/rvidemojsp/app/index.html>

NOTE: Make sure to put the corresponding php file under the root of the Windows/Linux Web Serrver.

Version 2.2

To run the NodeJs version of the RVI Demo you will first need to follow the Node Js installation listed below. Once Completed you can open a web browser and put in:

<http://yourip:yourport/internet/common/rvidemonodejs/app/index.html>

Version 2.3

To run the NodeJs/JsonP version of the RVI Demo you will first need to follow the Node Js installation listed below. Once Completed you can open a web browser and put in:

<http://yourip:yourport/internet/common/rvidemonjs/app/index.html>

NOTE FOR RUNNING VERSION 2.1 - 2.3: Make sure to edit the /cgibin/common/(projectName)/config/Settings.js file to match the corresponding WebSocket/JsonP IP addresses and Port numbers.

NOTE FOR RUNNING VERSION 2.2 OR 2.3: To use the CommonNode PF you will need to create a Sequence in SQL. You can do this by simple entering the command STRSQL and then pasting the following command into the prompt:

```
CREATE SEQUENCE COMMONSEQ  
START WITH 1  
INCREMENT BY 1  
MAXVALUE 10000  
CYCLE  
CACHE 24
```

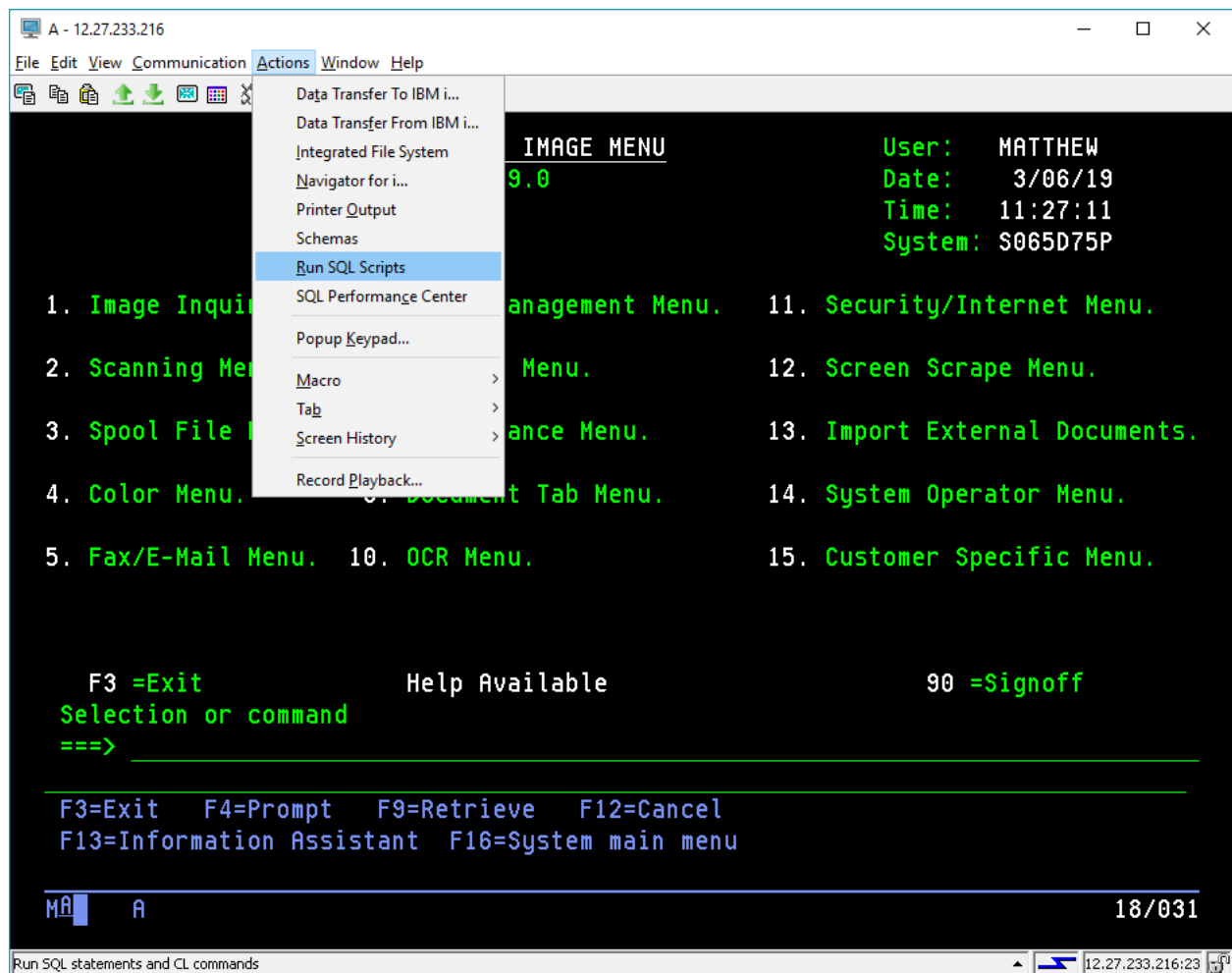
Enjoy good luck coding..

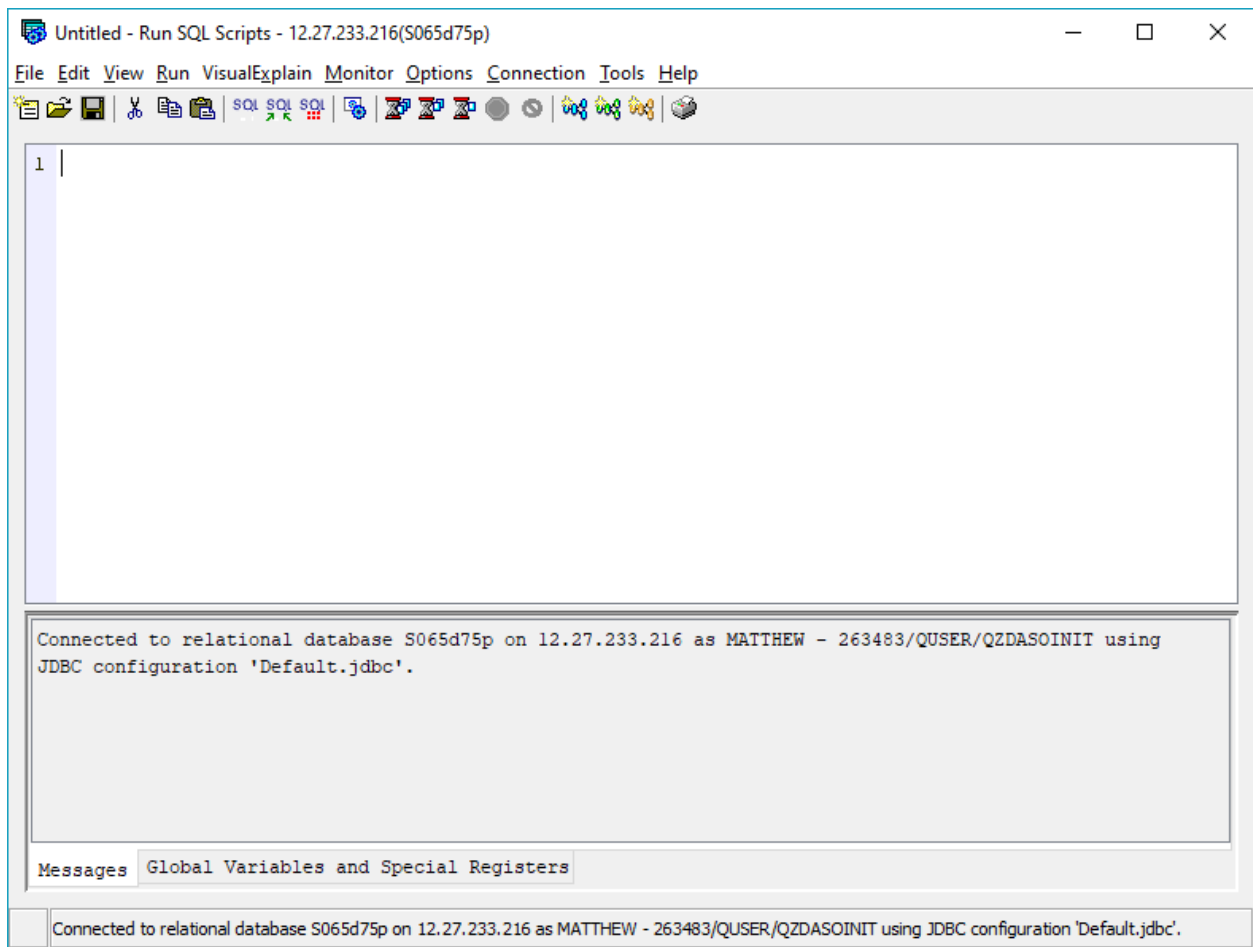
NODE JS INSTALLATION FOR IBMI

Open Source Info: <https://bitbucket.org/ibmi/opensource/src/master/docs/yum/>

Step 1: First we need to install “yum” which is IBM’s resource to fetch open source tools from within the IBM I command line. This will require Access Client Solutions (ACS). If ACS is not installed, please refer to the link above for install instructions.

- First we will click the “Actions” menu option from ACS and select “Run SQL Scripts”





- Next you will want to paste his SQL Script into the window to install “yum”:

```
create or replace table qtemp.ftpcmd(cmd char(240)) on replace delete rows;  
create or replace table qtemp.ftplog(line char(240)) on replace delete rows;
```

```
insert into qtemp.ftpcmd(CMD) values  
  ('anonymous anonymous@example.com')  
  , ('namefmt 1')  
  , ('lcd /tmp')  
  , ('cd /software/ibmi/products/pase/rpms')  
  , ('bin')  
  , ('get README.md (replace)')  
  , ('get bootstrap.tar.Z (replace)')  
  , ('get bootstrap.sh (replace)')  
with nc  
;
```

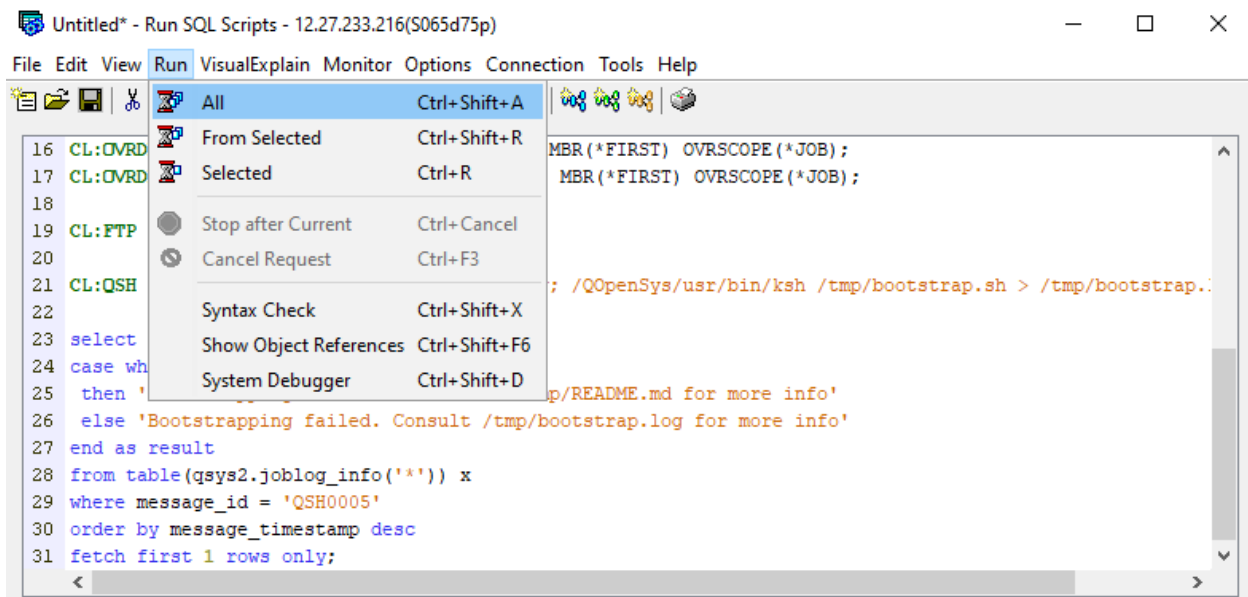
```
CL:OVRDBF FILE(INPUT) TOFILE(QTEMP/FTPCMD) MBR(*FIRST) OVRSCOPE(*JOB);  
CL:OVRDBF FILE(OUTPUT) TOFILE(QTEMP/FTPLOG) MBR(*FIRST) OVRSCOPE(*JOB);
```

```
CL:FTP RMTSYS('public.dhe.ibm.com');
```

```
CL:QSH CMD('touch -C 819 /tmp/bootstrap.log; /QOpenSys/usr/bin/ksh
/tmp/bootstrap.sh > /tmp/bootstrap.log 2>&1');
```

```
select
case when (message_tokens = X'00000000')
  then 'Bootstrapping successful! Review /tmp/README.md for more info'
  else 'Bootstrapping failed. Consult /tmp/bootstrap.log for more info'
end as result
from table(qsys2.joblog_info('*')) x
where message_id = 'QSH0005'
order by message_timestamp desc
fetch first 1 rows only;
```

- Next you will want to select the “Run” menu option on the top and then select the “All” option.



- If completed successfully, once script is finished running in the console you will see a message that says “Bootstrapping Successful!”. You have successfully installed “yum”.

```

16 CL:QVRDBF FILE(INPUT) TOFILE(QTEMP/FTPCMD) MBR(*FIRST) OVRSCOPE(*JOB);
17 CL:QVRDBF FILE(OUTPUT) TOFILE(QTEMP/FTPLOG) MBR(*FIRST) OVRSCOPE(*JOB);
18
19 CL:FTP RMTSYS('public.dhe.ibm.com');
20
21 CL:QSH CMD('touch -C 819 /tmp/bootstrap.log; /QOpenSys/usr/bin/ksh /tmp/bootstrap.sh > /tmp/bootstrap.log');
22
23 select
24 case when (message_tokens = X'00000000')
25 then 'Bootstrapping successful! Review /tmp/README.md for more info'
26 else 'Bootstrapping failed. Consult /tmp/bootstrap.log for more info'
27 end as result
28 from table(qsys2.joblog_info('*')) x
29 where message_id = 'QSH0005'
30 order by message_timestamp desc
31 fetch first 1 rows only;

```

RESULT

```

Bootstrapping successful! Review /tmp/README.md for more info

```

Done: 1 rows retrieved.

```

select case when (message_tokens = X'00000000') then 'Bootstrapping successful! Review /tmp/README.m...

```

Messages Global Variables and Special Registers

Connected to relational database S065d75p on 12.27.233.216 as MATTHEW - 263483/QUSER/QZDASOINIT using JDBC configuration 'Default.jdbc'.

Step 2: Now that we have “yum” installed we can now retrieve the open source files from IBM through the command line.

- First, we want to enter into SSH on the green screen by entering CALL QP2TERM from the command line. This will bring you into SSH.

A terminal window titled "A - 12.27.233.216" with a menu bar (File, Edit, View, Communication, Actions, Window, Help) and a toolbar. The terminal content is as follows:

```
/QopenSys/usr/bin/-sh

$

===>

F3=Exit    F6=Print    F9=Retrieve  F11=Truncate/Wrap
F13=Clear   F17=Top     F18=Bottom   F21=CL command entry

MA A 18/007
```

- Next we will want to change directory to the location where the “yum” package resides by entering “cd /QopenSys/pkg/bn” and press enter. From here you will now be able to use the yum command.

The same terminal window as above, but now showing the execution of the 'cd' command:

```
/QopenSys/usr/bin/-sh

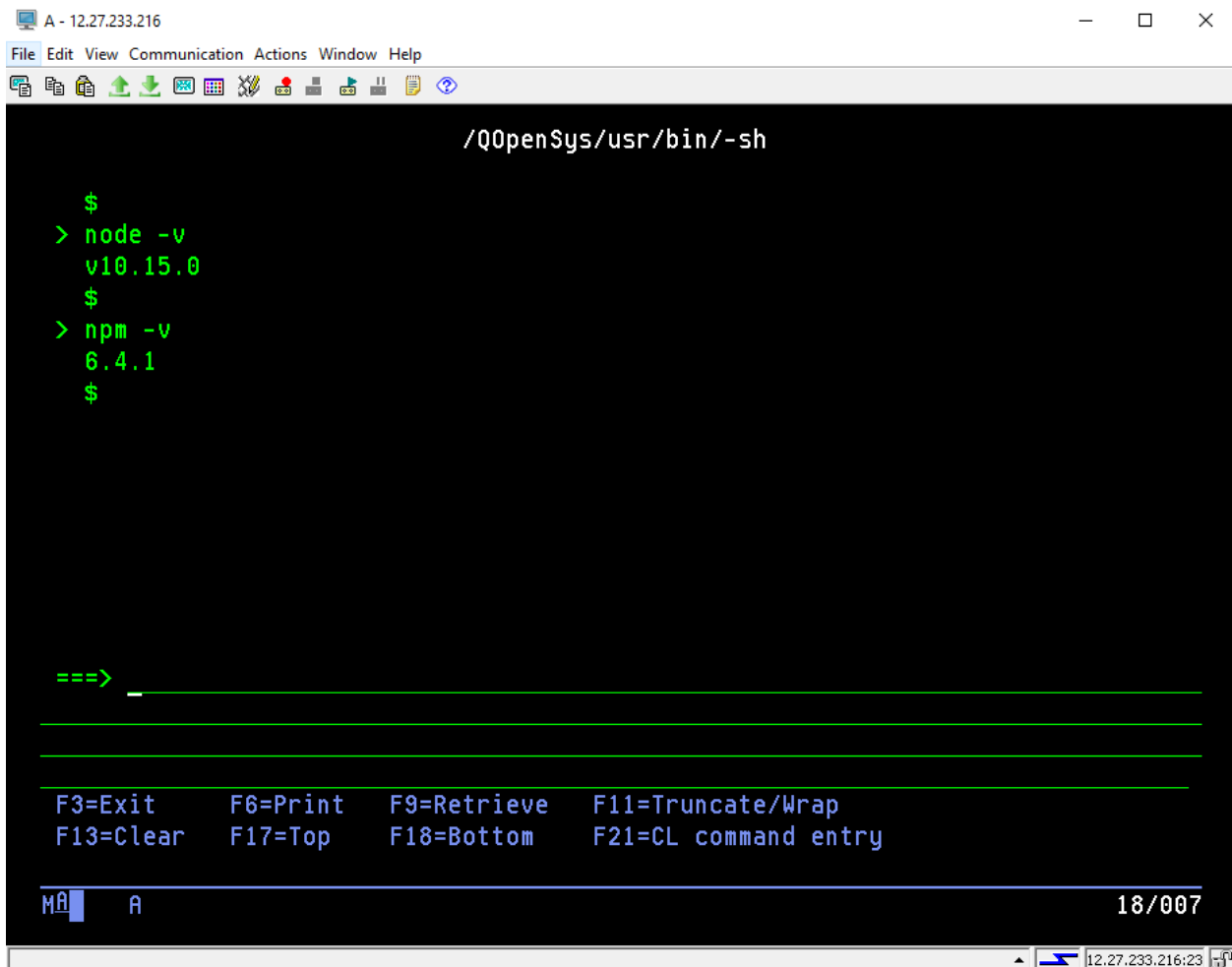
$
> cd /QopenSys/pkg/bn
$

===>

F3=Exit    F6=Print    F9=Retrieve  F11=Truncate/Wrap
F13=Clear   F17=Top     F18=Bottom   F21=CL command entry

MA A 18/007
```


- Now we will run the several suggested commands from the IBM open source doc listed above.
 - `yum install python3-pip python3-ibm_db python3-itoolkkit`
 - `yum install python3-numpy python3-pandas python3-scikit-learn python3-scipy`
 - `yum install nodejs10`
 - `yum group install "Development tools"`
- Once Finished installing Node 10 and all its dependencies we will run the nodever module to set Node 10 as a global variable. You can do this by running this command:
 - `nodever 10`
- To test that you have successfully installed node and the npm (which is the package manager for node) you can run these 2 commands:
 - `node -v`
 - `npm -v`



A terminal window titled "A - 12.27.233.216" with a menu bar (File, Edit, View, Communication, Actions, Window, Help) and a toolbar. The terminal prompt is `/QOpenSys/usr/bin/-sh`. The user enters `node -v` and receives `v10.15.0`. Then the user enters `npm -v` and receives `6.4.1`. Below the terminal output, there is a legend for function keys: `F3=Exit`, `F6=Print`, `F9=Retrieve`, `F11=Truncate/Wrap`, `F13=Clear`, `F17=Top`, `F18=Bottom`, and `F21=CL command entry`. The terminal status bar shows "MA A" on the left and "18/007" on the right. The window title bar at the bottom right shows "12.27.233.216:23".

```
A - 12.27.233.216
File Edit View Communication Actions Window Help
/QOpenSys/usr/bin/-sh

$
> node -v
v10.15.0
$
> npm -v
6.4.1
$

===>

F3=Exit   F6=Print   F9=Retrieve  F11=Truncate/Wrap
F13=Clear  F17=Top    F18=Bottom   F21=CL command entry

MA A 18/007
12.27.233.216:23
```

- If this is displayed after the last couple steps, then you have successfully installed Node version 10 and NPM version 6.4.1

Step 3: We will now need to install a couple Node Modules from the NPM for our node application to work. We will be installing the “ws” package which is for our WebSocket server, the “idb-connector” package which is used to connect to the database, and also the “forever” package which will allow the node script to run behind the scenes. Go to your home directory and create a directory called node. You can do this by executing the command “mkdir /home/node”(Important: This is the location we will expect to run the NodeJS scripts. Do not create node in any other location).

- After creating the “node” directory we want to change directory to the “node” directory by executing this command: “cd /home/node”
- We can then install these node modules by running these simple commands:
 - npm install ws
 - npm install idb-connector
 - npm install forever

Step 4: Now that we have these modules installed we are ready to run the NodeJS script.

- You will then need to grab a copy of the as400common.js file for the following steps. Put the as400common.js file inside of the node directory you just created.
- Next we will change directories to the forever directory to execute our forever command. “cd /home/node/node_modules/forever/bin”
- Now that we are in the forever directory we can now execute the forever start command. “forever start /home/node/as400common.js”. You can now exit out of QSH once the forever process is finished.
- To check and make sure your job has started successfully, WRKUSRJOB and you should see something like this in the job log.

```

  QPADEV00B6  MATTHEW  BATCHI  ACTIVE  PGM-node
  QPADEV00B6  MATTHEW  BATCHI  ACTIVE  PGM-node

```

NODE JS INSTALLATION FOR WINDOWS

Refer to <https://nodejs.org/en/> for Documentation and Downloads.

Step 1: Install NodeJs on the server you are wishing to connect to the project. You can use the link above to grab the most current version of NodeJs.

Step 2: Open the CMD line as Admin and install the few packages needed for the the NodeJs script provided (WinCommon.js) to run correctly. These include:

- "npm install ws"
- "npm install mssql"

Step 3: Next you will want to create a directory on the C: drive called node and put the WinCommon.js script provided inside of that folder. Open the script and provide the necessary information to connect to the database of your choice.

Step 4: Lastly you will want to change directory to the newly created node directory from the command line using "cd c:\node" and then enter "forever start WinCommon.js". Your WebSocket server on your windows machine should now be running.

NODE JS INSTALLATION FOR UBUNTU 18.04

Refer to <https://nodejs.org/en/> for Documentation and Downloads.

Step 1: Install Nodejs and npm from the Ubuntu repository on the server you are wishing to connect to the project. You can use the link above to grab the most current version of Nodejs.

Step 2: The following packages needed for the Nodejs script provided(linuxCommon.js) to run correctly. The information provided in this manual is written for Ubuntu; however, the same instructions apply for any Ubuntu-based distribution including Kubuntu, Linux Mint, and Elementary OS. Furthermore, to install Nodejs and npm on a different Linux distribution, consult your Linux distribution manual in detail.

¹Open the Linux Terminal by pressing Ctrl + Alt + T and before starting the installation make sure the to be logged in as a user with sudo privileges.

- I. Start by updating the packages list by typing:

```
$ sudo apt update
```

- II. Install nodejs using the apt package manager

```
$ sudo apt install nodejs npm
```

The Nodejs executable from Ubuntu repositories is named Nodejs instead of node.

- III. To verify the installation, type the following command:

```
$ nodejs --version
```

Output:

```
v8.10.0
```

- IV. To be able to install npm packages you also need to install npm, the Nodejs packages manager. Type:

```
$ sudo apt install npm
```

- V. Verify the installation by typing:

```
$ npm --version
```

Output:

```
3.5.2
```

¹ The documentation on how to install Nodejs and node package manager is open-source and can be found with a simple search or my following this link: <https://linuxize.com/post/how-to-install-node-js-on-ubuntu-18.04/>

Step 3: Do not attempt to use “sudo npm install [something]”, this could potentially give an error and make your life more difficult². Before we can run our script, we need to type in the Ubuntu Terminal the following commands.

- “npm install ws”
- “npm install mysql”
- “npm install forever”

Step 4: Next, inside the Ubuntu file system you will want to create a new directory under the ~/home/user directory called node and put the linuxCommon.js script provided inside of that folder. You can do it by typing:

```
$ sudo mkdir /home/user/NewFolder.
```

You could create the node directory anywhere inside the file system. If you want to create the folder on your Documents directory the location will be:

```
$ sudo mkdir /home/user/Documents/NewFolder
```

Where “user” is your username and “NewFolder” is the name of the folder – node – you are creating. Open the script and provide the necessary information to connect to the database of your choice.

Step 5: Lastly you will want to change directory to the newly created node directory from the command line by typing:

```
$ cd /home/user/Documents/node
```

and then enter:

```
$ forever start linuxCommon.js
```

Your WebSocket server on your Linux distribution machine should now be running.

² This article provides information on why to avoid using “sudo npm install [something]”, follow this link: <https://medium.com/@ExplosionPills/dont-use-sudo-with-npm-5711d2726aa3>