

In the Bluemix UI navigate to your application overview page for the NodeRED application

Select **Edit code** in the Continuous Delivery section to launch the Bluemix DevOps Services page for this application

select **bluemix-setting.js** from the file tree in the editor window

find the definition of the functionGlobalContext object

We are going to add a CRC checking package. To make the package available you define an object attribute with a name and then the require statement as the value:

```
1 functionGlobalContext: {  
2   crc:require('crc')  
3 },
```

select **File** then **Save** to save the bluemix-setting.js file

Now we need to bring the crc package into the application, so need to define it as a dependency in the **package.json** file. Don't forget to add the comma to the preceding line.

```
1 "dependencies": {  
2   ...  
3   "node-red-node-random": "0.0.6",  
4   "crc": "3.x"  
5 },
```

Select **File** then **Save** to save the file

Switch into the Git section of the editor.

Enter a commit message and ensure both bluemix-setting.js and package.json are shown and selected in the Working Directory Changes section then hit **Commit**.

In the Outgoing section you should now have your 1 committed change, so select **PUSH ALL** to send it to the master branch. This will automatically start the build and deploy process for your application.

Once the application has restarted you have access to the package from a function node:

```
1 var crc = global.get('crc');  
2 msg.payload = crc.crc32('hello').toString(16);  
3 return msg;
```