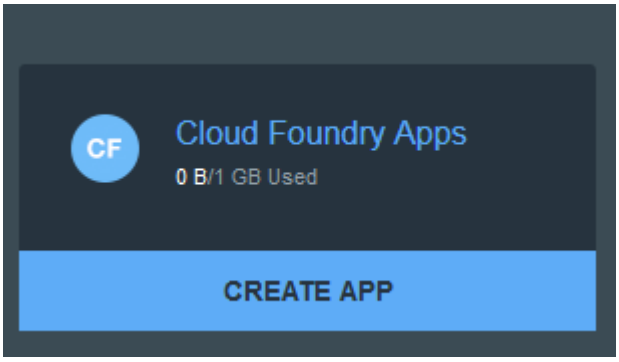
IBM Watson Machine Learning
General Discussion on Application Development

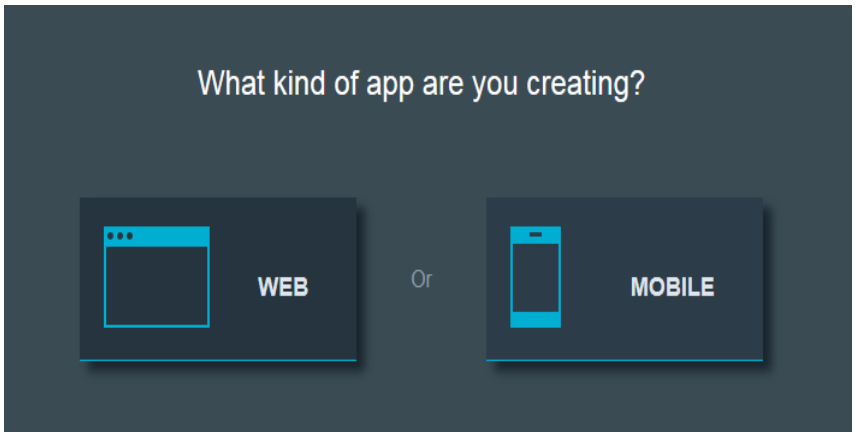
Preparing to develop the application

Bluemix makes it very easy to get a new application started. It is worth using the Bluemix **Create App** functionality at least once to get a good feel for what is involved.

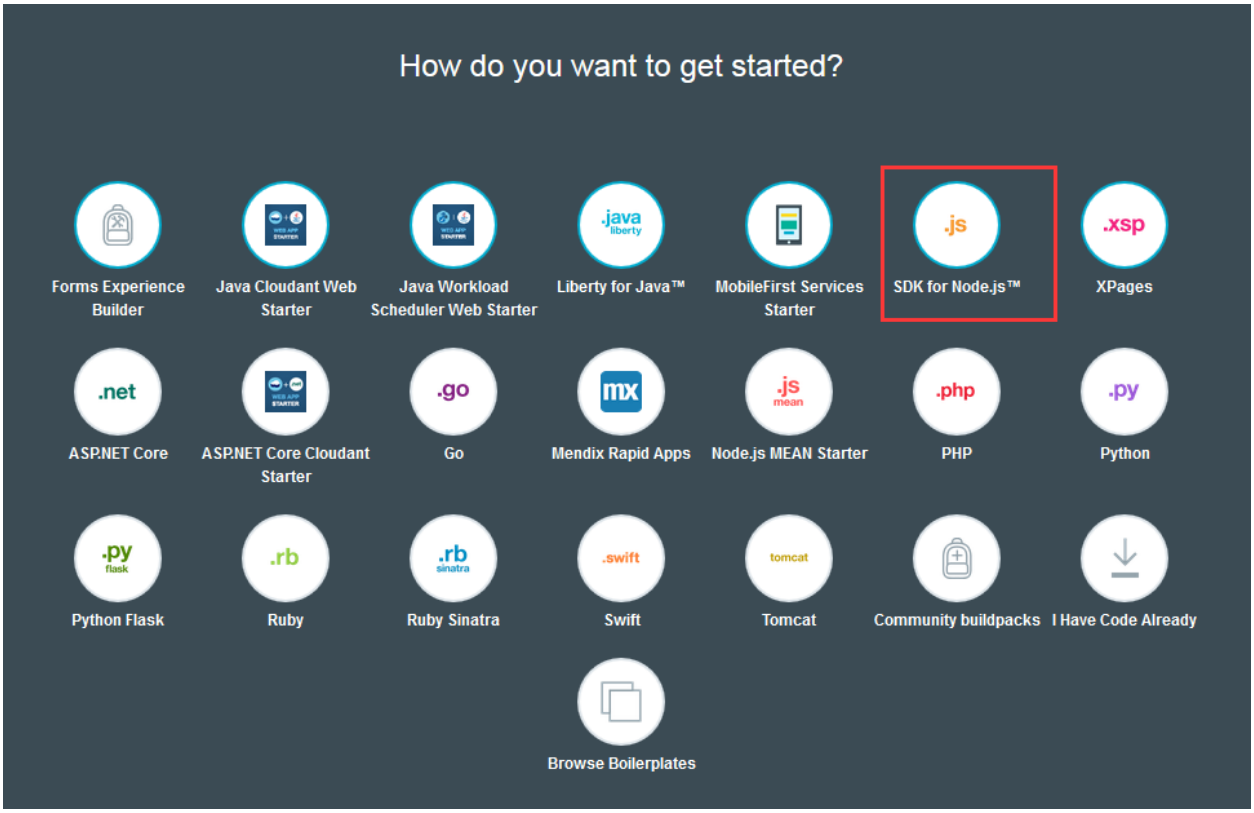
1. Go to the Bluemix web site. We'll be using the development instance in this example, which is at https://console.stage1.ng.bluemix.net/
2. Log in using your account details or sign up now for an IBM ID and access to Bluemix. It's free.
3. You may create an "org" and "space" for doing this sample development. The names used are up to you.
4. Once on the dashboard for your development space, click **Create App**.



5. In this example, we will create a NodeJS application. But any language capable of making REST service calls will do.
6. Choose **Web**.



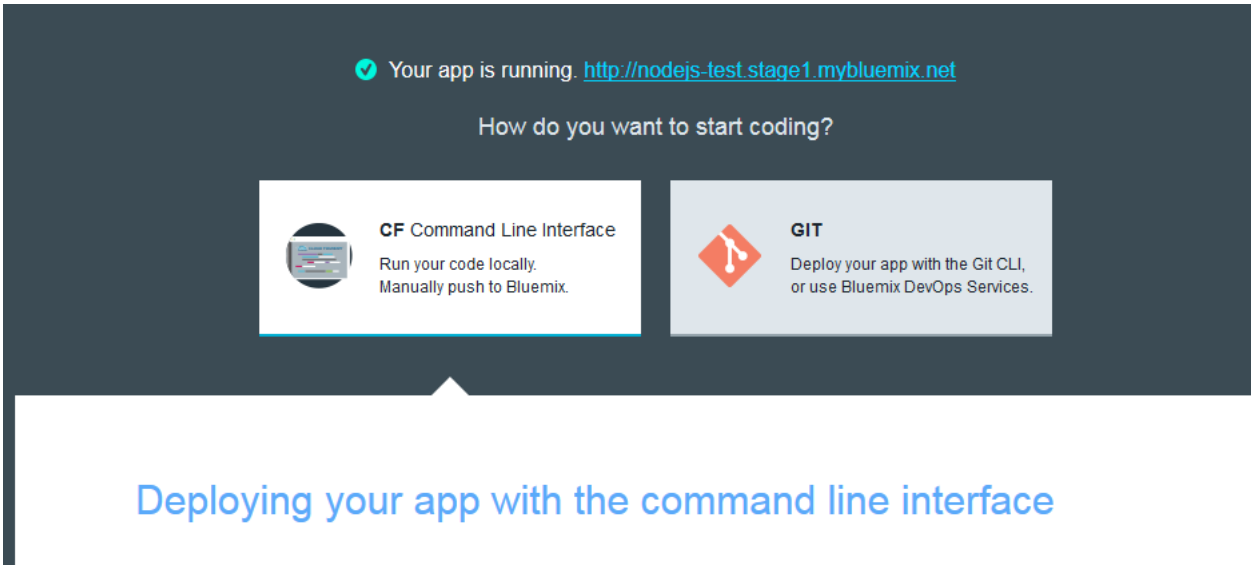
Then choose **SDK for Node.js**.



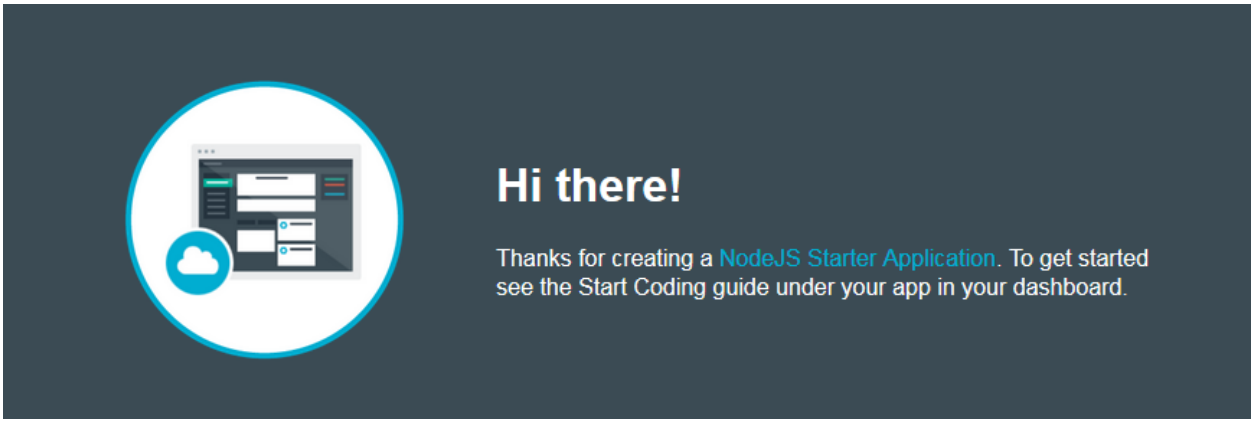
7. Then your application will be created and running. It's a simple NodeJS Starter Application. If you want more features, you can push your own NodeJS code onto this application by using one of the following methods:

   a) CF Command Line Interface

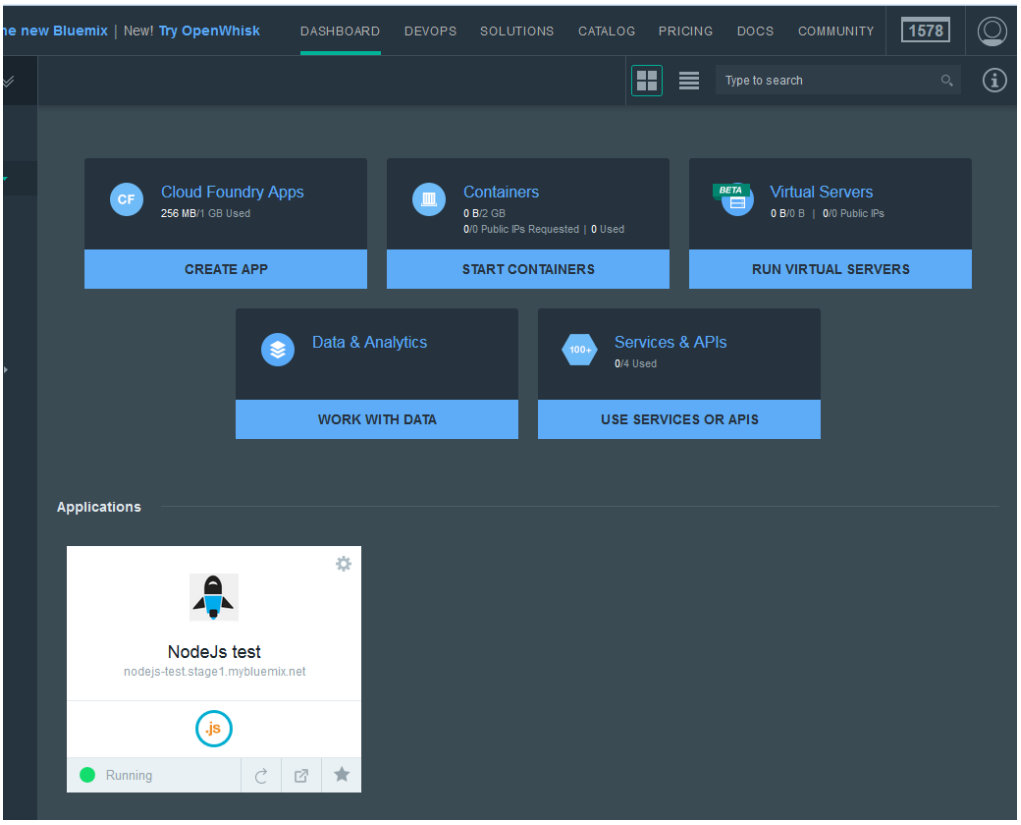
   b) GIT



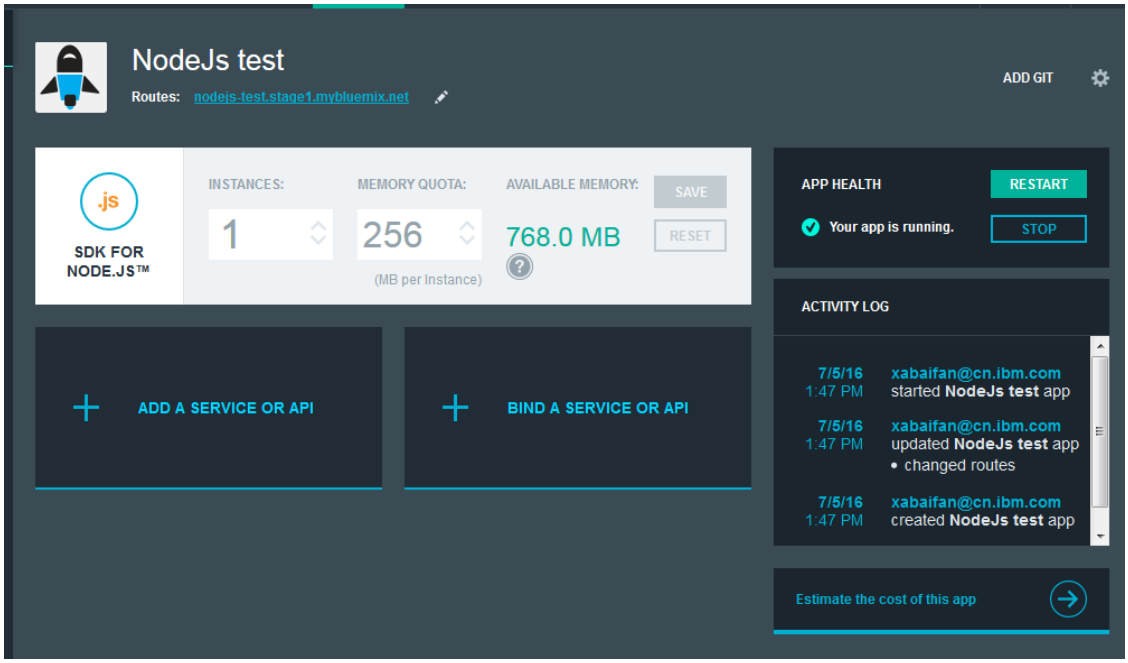
8. Click the URL for the newly deployed application (beside the **Your app is running**) to see the welcome page. This means your application is successfully deployed.



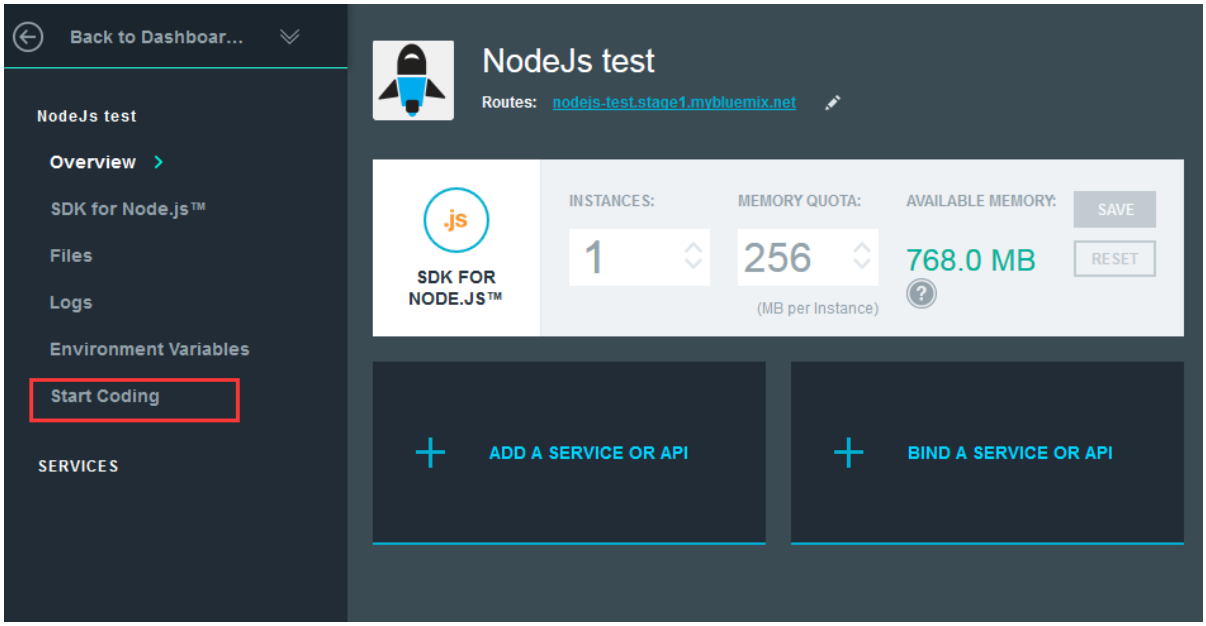
9. A number of things happen when you push **Create** on your new application. A viable "shell" application for NodeJS is created and "pushed" out to Bluemix. The DNS routing entries that let you get to this application are entered. Then the application is started (a basic "Hello World" application). You can see the application on your dashboard.

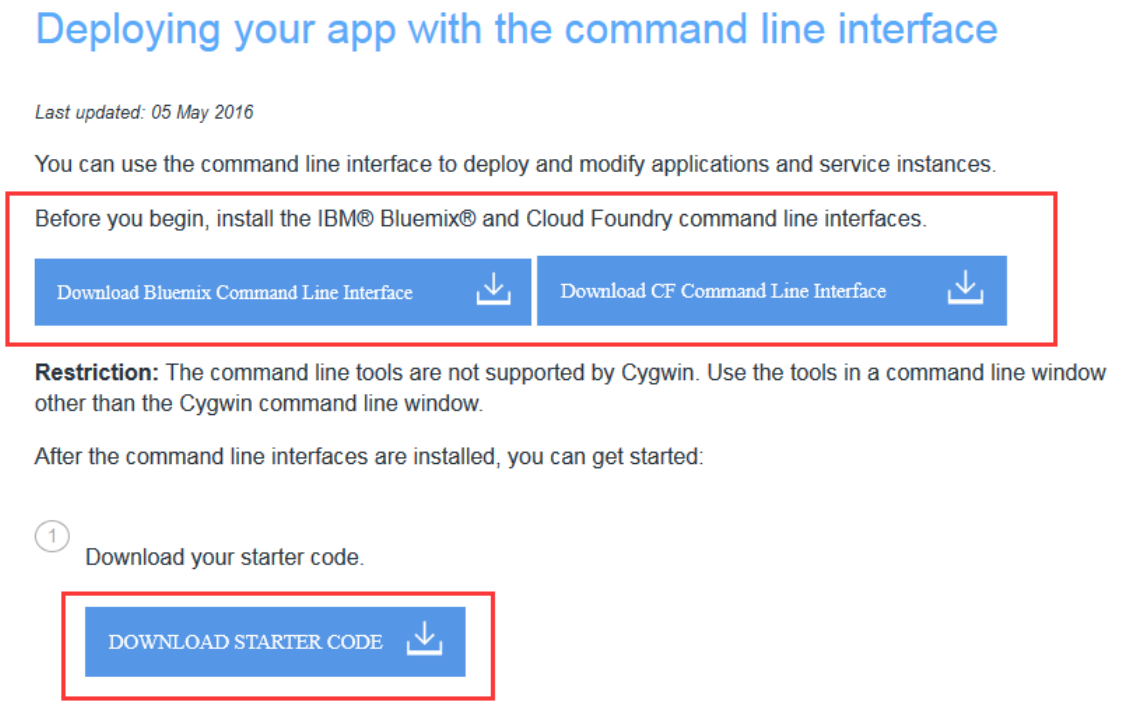


10. To see more details, click the application you just deployed.

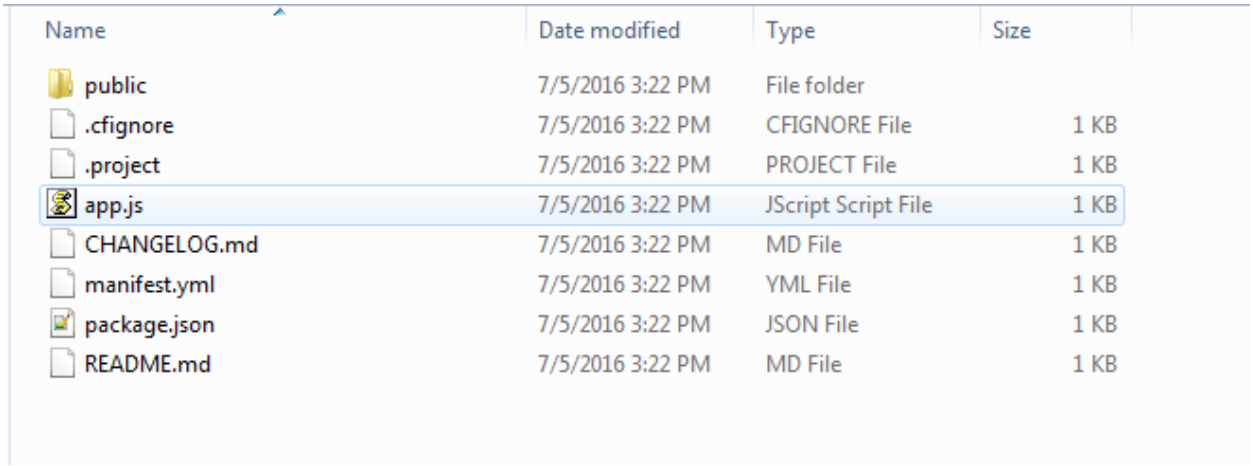


11. The fun begins when you download this application and work with it on your desktop. When you select **Start Coding** on our application, you get help in installing the CF command line tool and downloading the sample application by clicking **Download the starter code**. But before you do that, install the Bluemix and CF command line interface, and add their installation directory to your Environment Variable "%Path%."





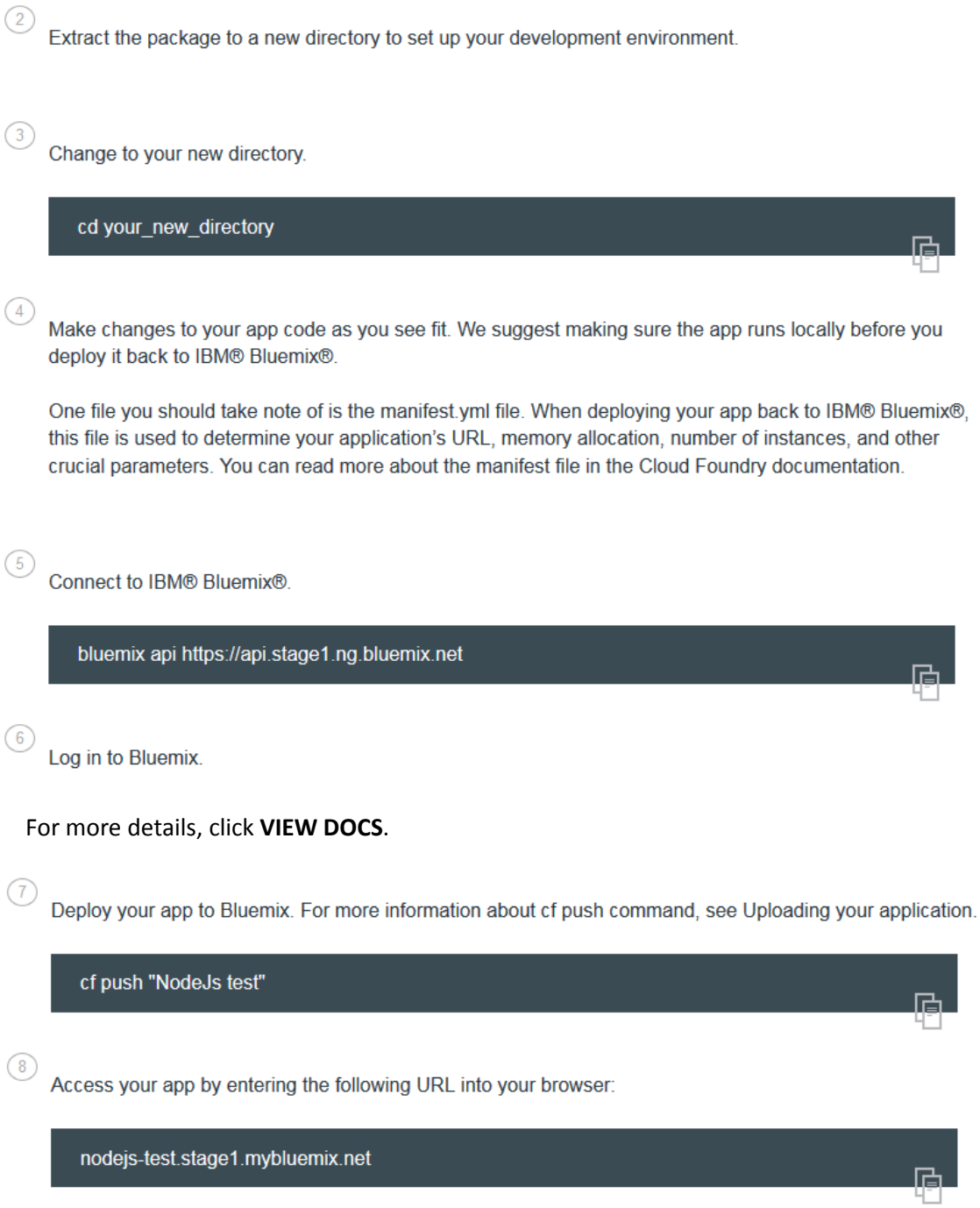
What is in this application bundle when we download it?



We get the basic framework for a NodeJS application (*app.js* and *package.json*) and the *manifest.yml* used when we push this application to Bluemix. There is also a *README.txt* file to help you take your next steps in building a NodeJS application.

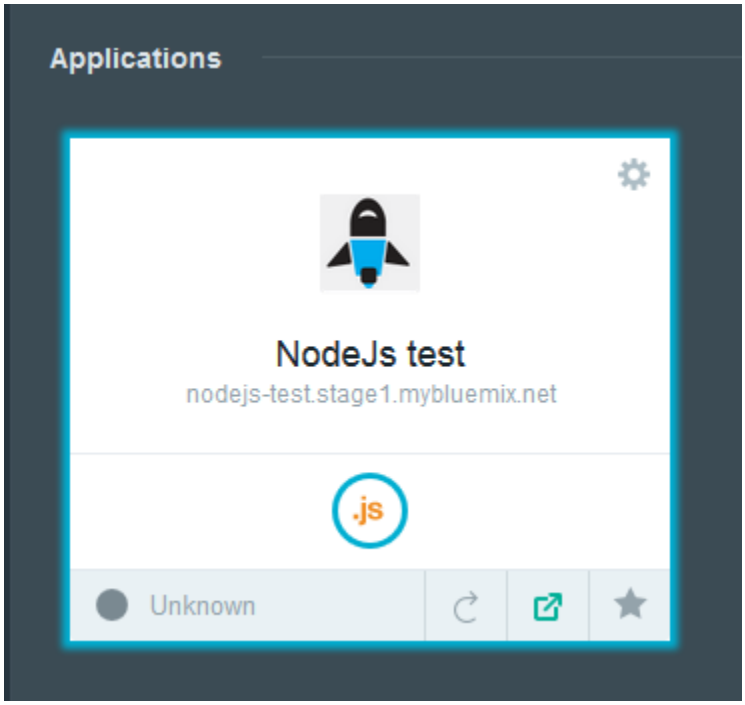
Pushing the modified application back to Bluemix

The following comments (under the **Download the starter code** button described in step 11 previously) show how to push back the application after modifying it.



The entire bundle is transferred up to Bluemix and you will be notified as it goes through all phases of deployment and finally restarts.

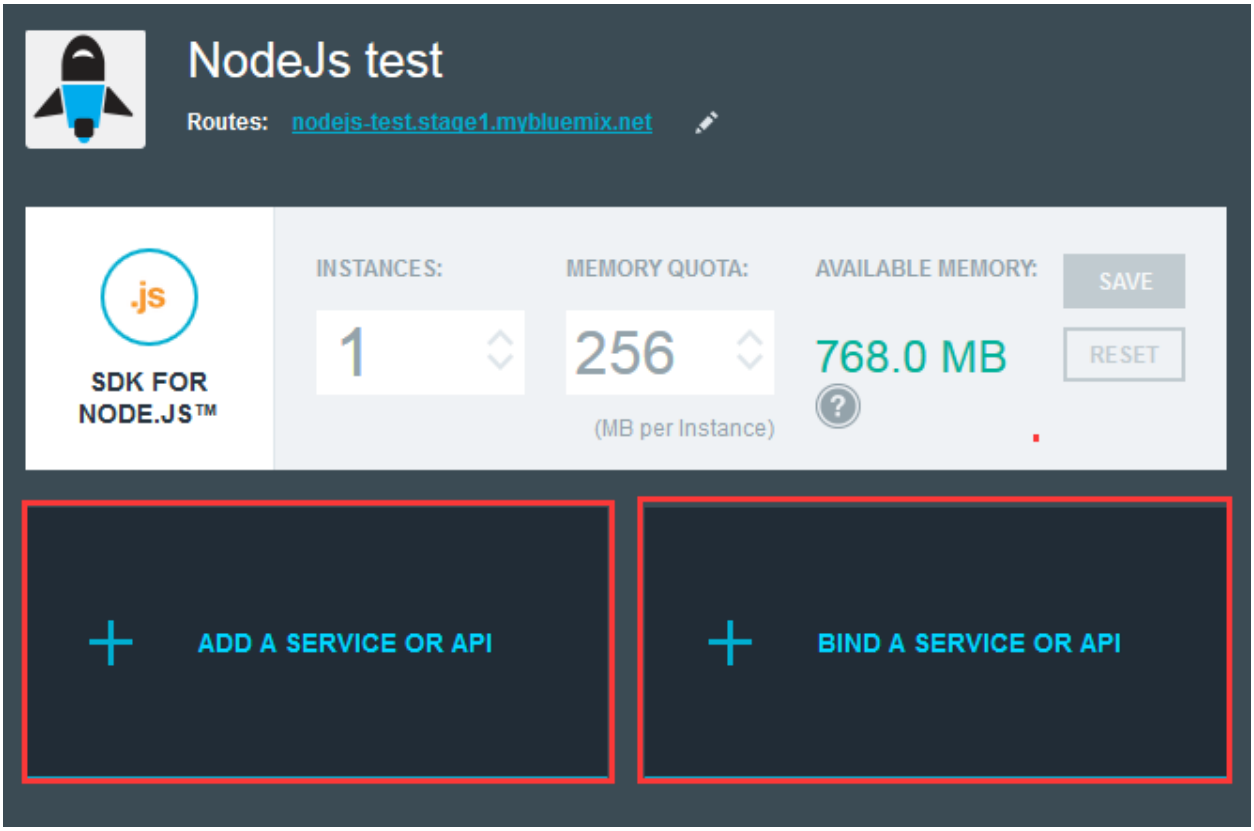
Once restarted, you can launch the application from the Bluemix dashboard by clicking the **Open app in a new page** button.



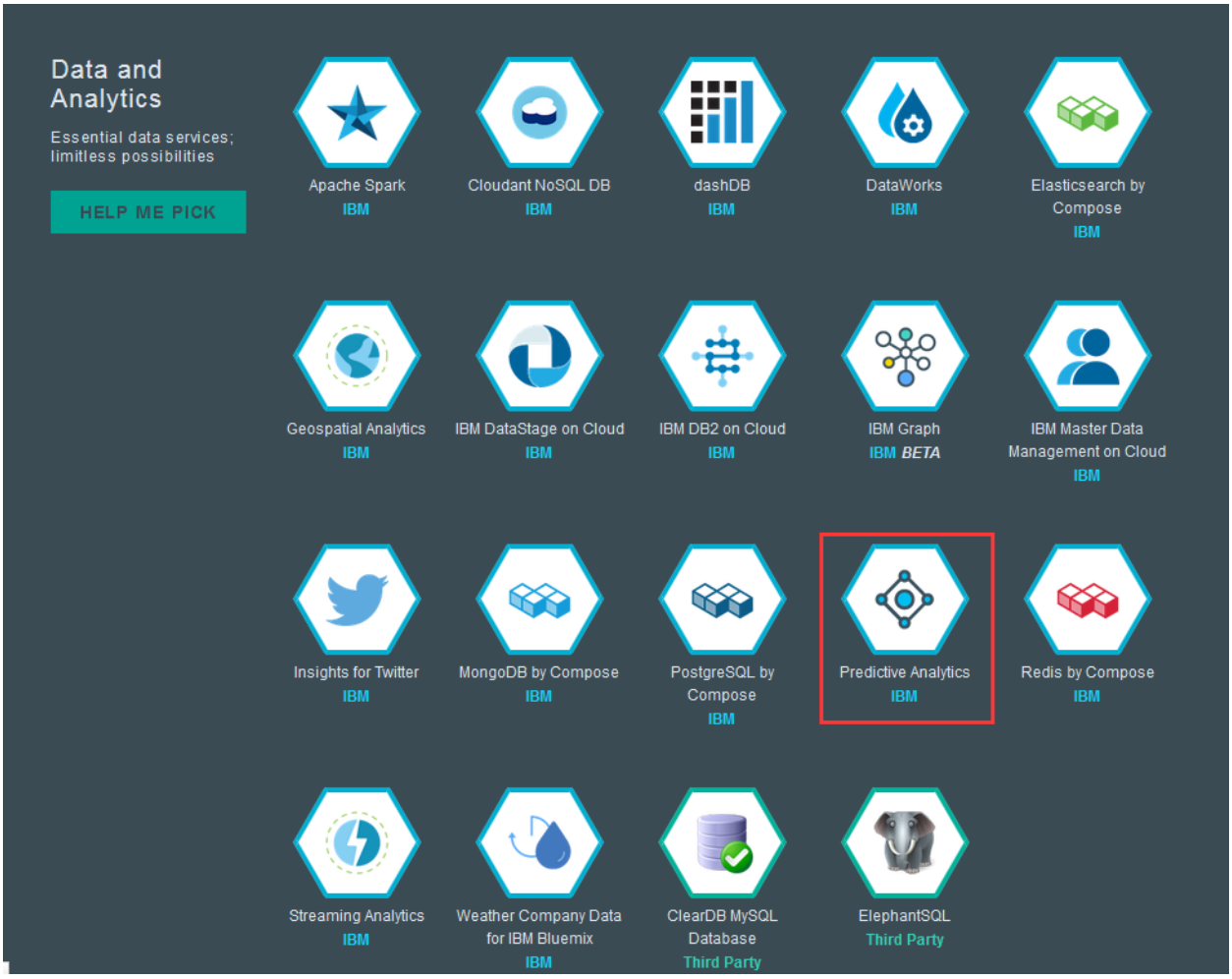
Testing your Bluemix application during development

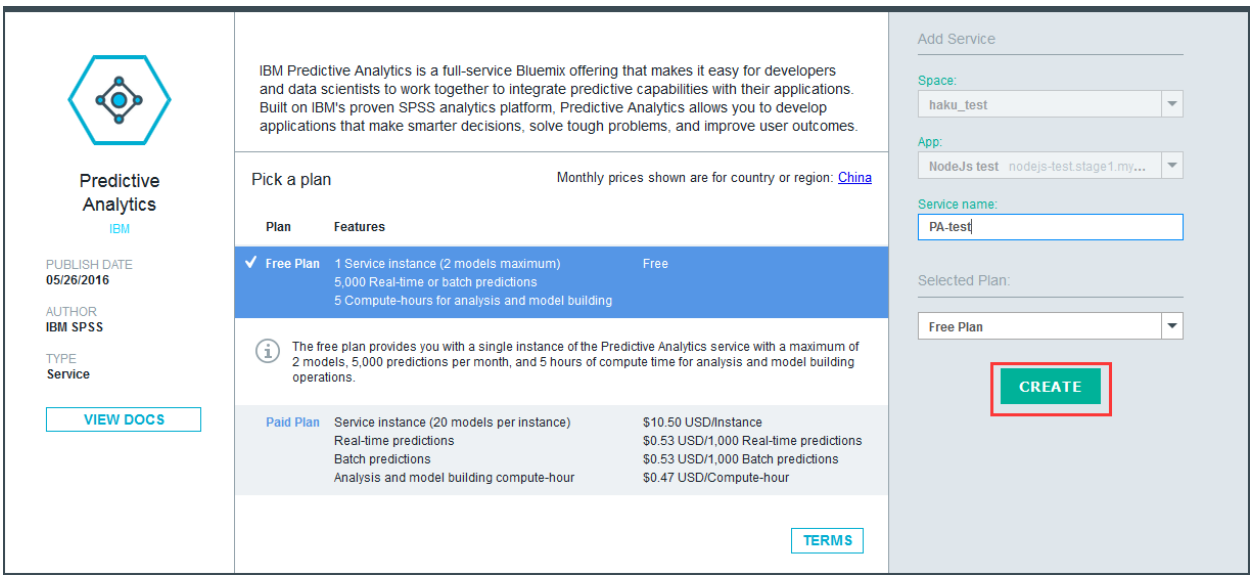
You may provision an instance of the IBM Machine Learning service at any point in time and get the connectivity information you need to test your application from your desktop. You may also push your application back up to Bluemix at any point in time and test it there. To test your application from your desktop, you will need a compatible development environment. In the example we've discussed here, this would mean a NodeJS server install.

Use the **ADD A SERVICE OR API** or **BIND A SERVICE OR API** button on your Bluemix application to connect it to your service instance. **ADD A SERVICE OR API** adds a new service and binds it to your application. **BIND A SERVICE OR API** binds an existing service or API to your application.

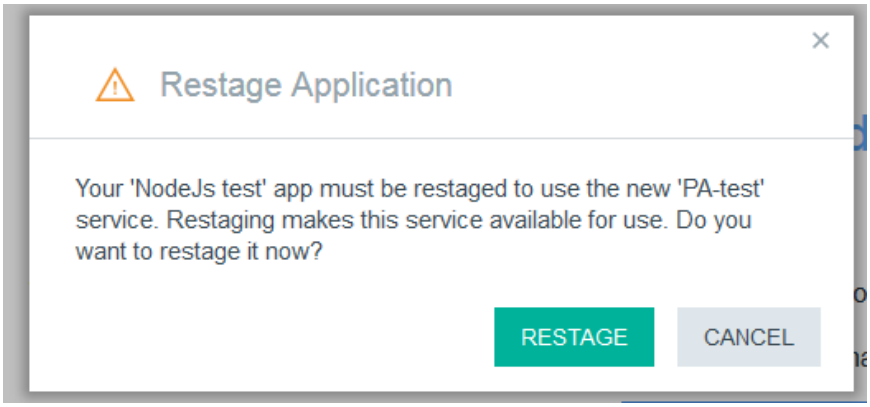


1. Since this is the first time, click **ADD A SERVICE OR API**.
2. In the **Data and Analytics** category, find **IBM Watson Machine Learning**.

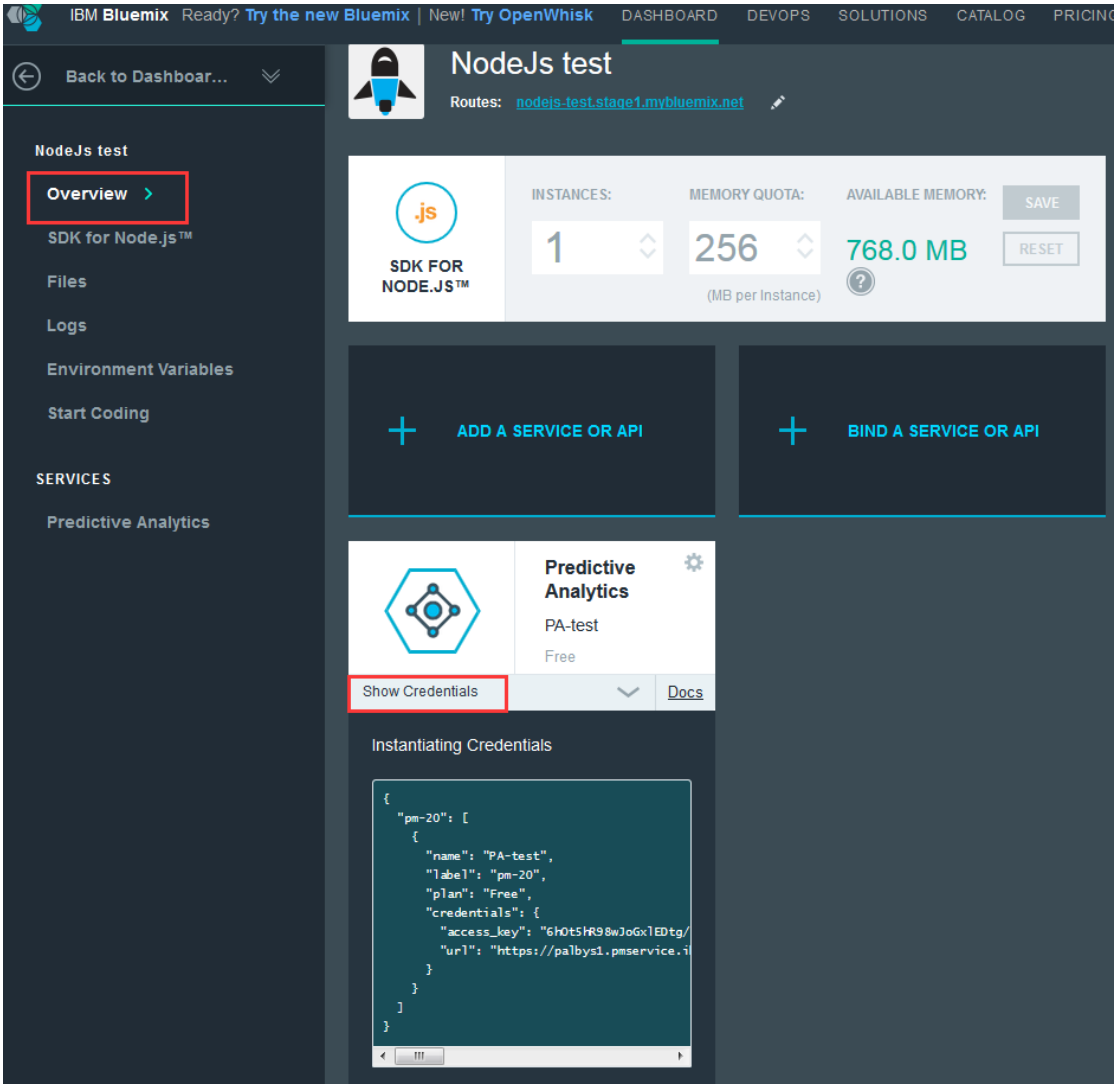

Data and Analytics
Essential data services; limitless possibilities
HELP ME PICK
Apache Spark
IBM
Cloudant NoSQL DB
IBM
dashDB
IBM
DataWorks
IBM
Elasticsearch by Compose
IBM
Geospatial Analytics
IBM
IBM DataStage on Cloud
IBM
IBM DB2 on Cloud
IBM
IBM Graph
IBM BETA
IBM Master Data Management on Cloud
IBM
Insights for Twitter
IBM
MongoDB by Compose
IBM
PostgreSQL by Compose
IBM
Predictive Analytics
IBM
Redis by Compose
IBM
Streaming Analytics
IBM
Weather Company Data for IBM Bluemix
IBM
ClearDB MySQL Database
Third Party
ElephantSQL
Third Party


Predictive Analytics
IBM
PUBLISH DATE
05/26/2016
AUTHOR
IBM SPSS
TYPE
Service
VIEW DOCS
IBM Predictive Analytics is a full-service Bluemix offering that makes it easy for developers and data scientists to work together to integrate predictive capabilities with their applications. Built on IBM's proven SPSS analytics platform, Predictive Analytics allows you to develop applications that make smarter decisions, solve tough problems, and improve user outcomes.
Pick a plan
Monthly prices shown are for country or region: China
Plan
Features
Free Plan
1 Service instance (2 models maximum)
5,000 Real-time or batch predictions
5 Compute-hours for analysis and model building
Free
The free plan provides you with a single instance of the Predictive Analytics service with a maximum of 2 models, 5,000 predictions per month, and 5 hours of compute time for analysis and model building operations.
Paid Plan
Service instance (20 models per instance)
Real-time predictions
Batch predictions
Analysis and model building compute-hour
$10.50 USD/Instance
$0.53 USD/1,000 Real-time predictions
$0.53 USD/1,000 Batch predictions
$0.47 USD/Compute-hour
TERMS
Add Service
Space:
haku_test
App:
NodeJs test
nodejs-test.stage1.my...
Service name:
PA-test
Selected Plan:
Free Plan
CREATE

3. Set your service name and fee plan, and then click **CREATE**. Now the service is created and bound to your application. You may click RESTAGE to restage your application and let the bound service work.



4. Go to the application **Overview** page to see the bound service. Once the application is bound, you can click the **Show Credentials** button on the bound service icon to obtain the connectivity information you need to test your application from your desktop.



5. Modify the NodeJS example *app.js* file to use either the Bluemix-provided environment variables VCAP_APP_HOST, VCAP_APP_PORT, and VCAP_SERVICES, or values you set to test from your desktop.

6. Read the documentation in the *Sample1* and *Sample2* folders for information about how *Sample1* and *Sample2* work.