# Module 1: File Processing

In this module you will use Azure Storage, Azure Function, and Azure Cosmos DB to process data from a JSON file. The Azure Function will be triggered when a new JSON file is created in Azure storage. The Azure Function will read the data and populate records into an Amazon Cosmos DB table.

## Architecture Overview



A JSON file is uploaded to Azure storage. A [blob trigger](https://docs.microsoft.com/en-us/azure/azure-functions/functions-bindings-storage-blob#trigger) configured on an Azure Function is triggered when a blob is added in the Azure storage container. The Azure Function retrieves the file, processes it, and populates the Cosmos DB with the data from the JSON file.

## Implement Instructions

### Create Azure Resources

1. Log into the Azure Management Portal with an account that has permissions to deploy new Azure resources.
2. To deploy a customized template through the portal, click **New**, and search for **Template Deployment** until you can select it from the options.
3. Click **Template Deployment**.
4. Click **Create**.
5. You see several options for creating a template. Click **Build your own template in the editor**.
6. You now have a blank template that is available for customizing. Delete the JSON in the blank template, then paste in the JSON from the following file *“<sourcecodepath>/1\_FileProcessing/azuredeploy.json”*.
7. Click **Save**.
8. Enter a new name for the Resource group. For example: File-Processing
9. Check the **I agree to the terms and conditions stated above** checkbox.
10. Click **Purchase**.
11. After the Azure resources are deployed, continue with the steps below.

### Storage Account Configuration

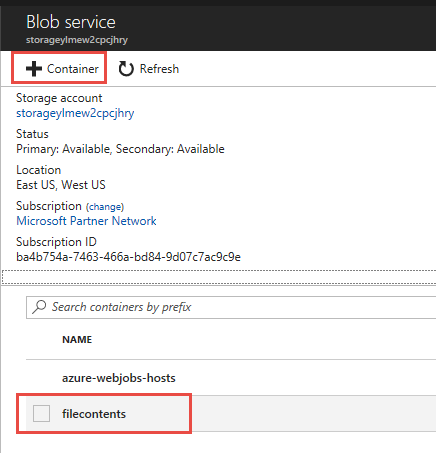
1. Go to the **Storage Account** created by the ARM template.

**For example:**



1. Create a container named **filecontents** under the Blob service.

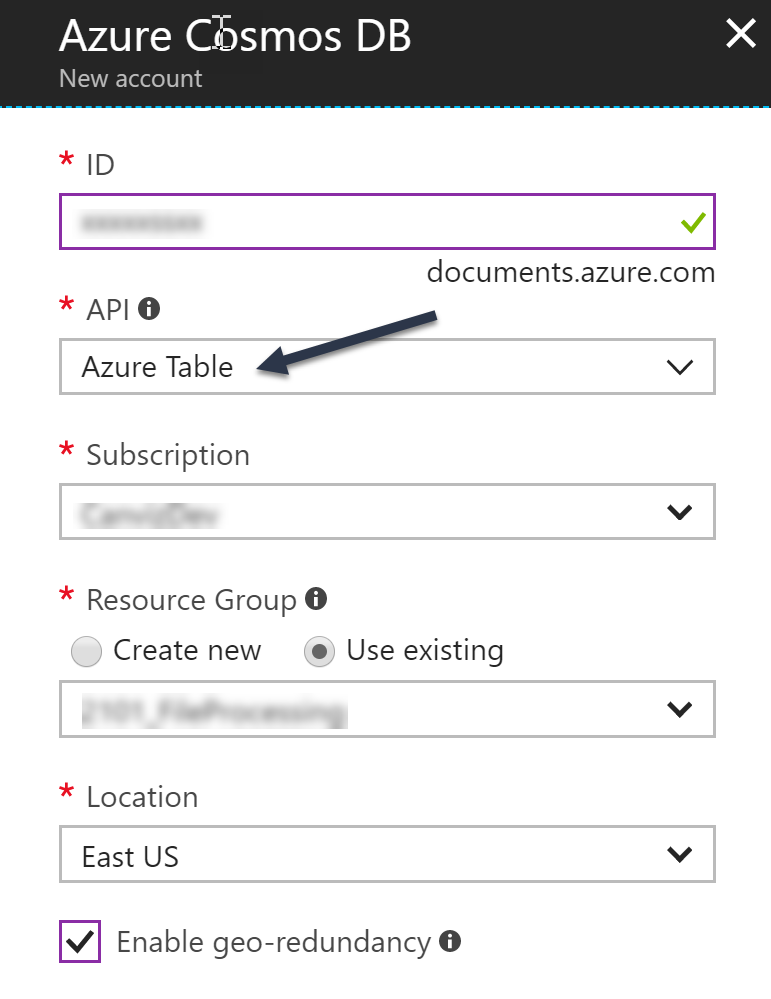
**For example:**



### CosmosDB Account Configuration

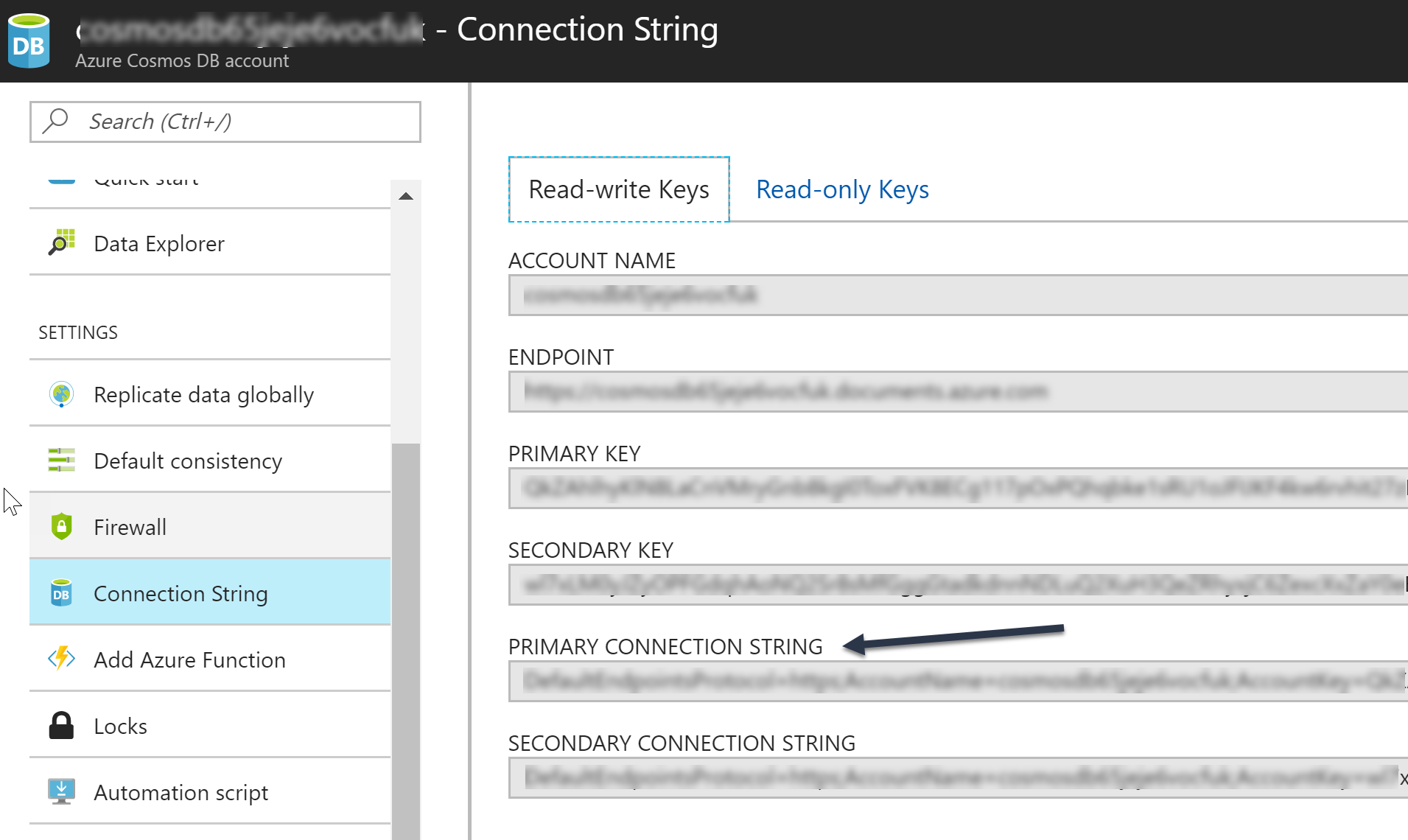
1. Create a new Azure Cosmos DB in the resource group.

**For example:**



1. After the Cosmos DB is created, copy the primary connection string to a text file. You will use the connection string in subsequent steps.

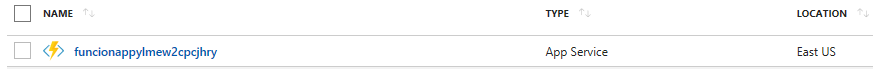
**For example:**



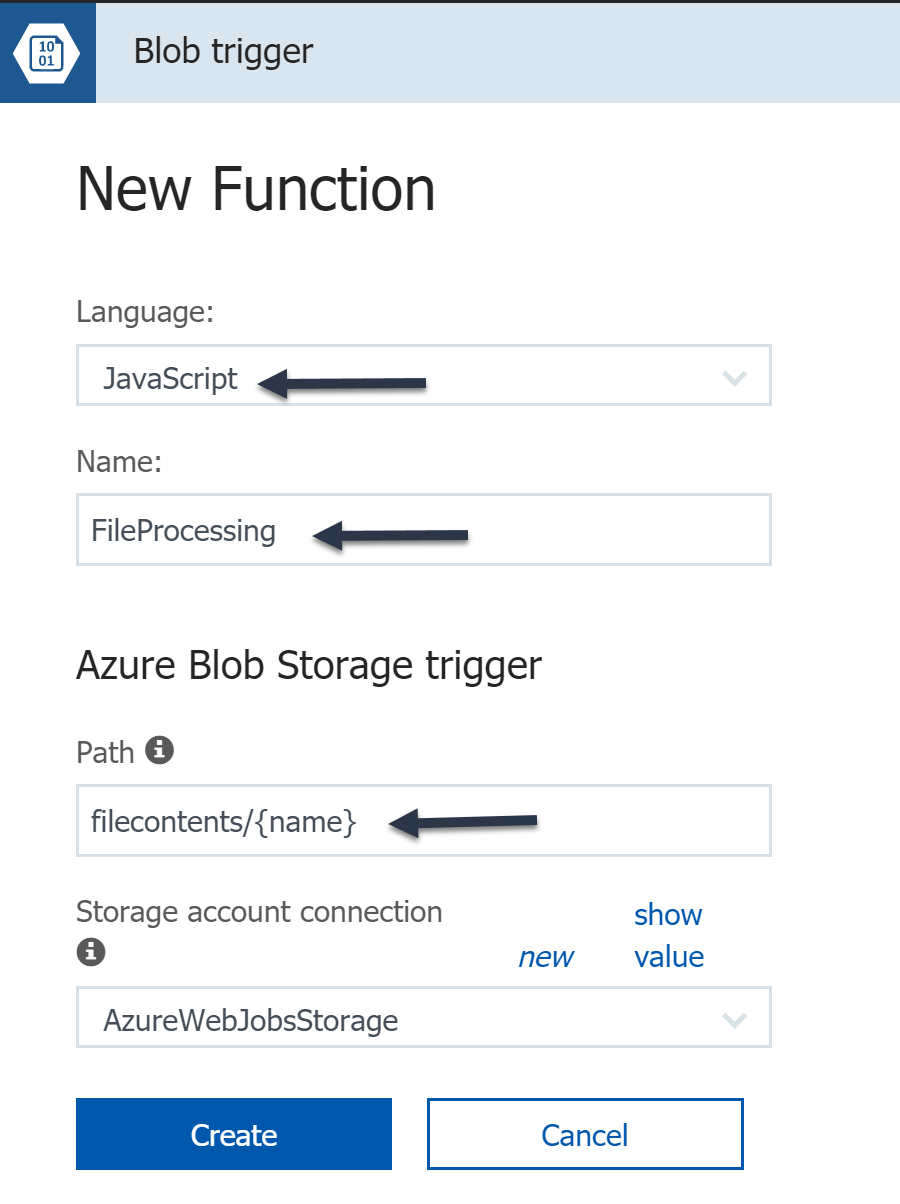
### Azure Function Configuration

1. Go to the **Azure Function** **App Service** created by the ARM template.

**For example:**

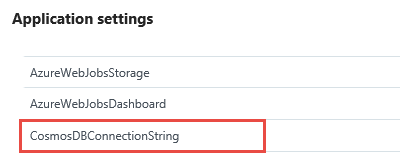
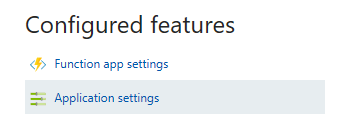


1. Create a Blob trigger function named FileProcessing. See the Create a Blob storage triggered function in [this article](https://docs.microsoft.com/en-us/azure/azure-functions/functions-create-storage-blob-triggered-function) to see how to do it. Use the settings in the screenshot below to create the New Function.

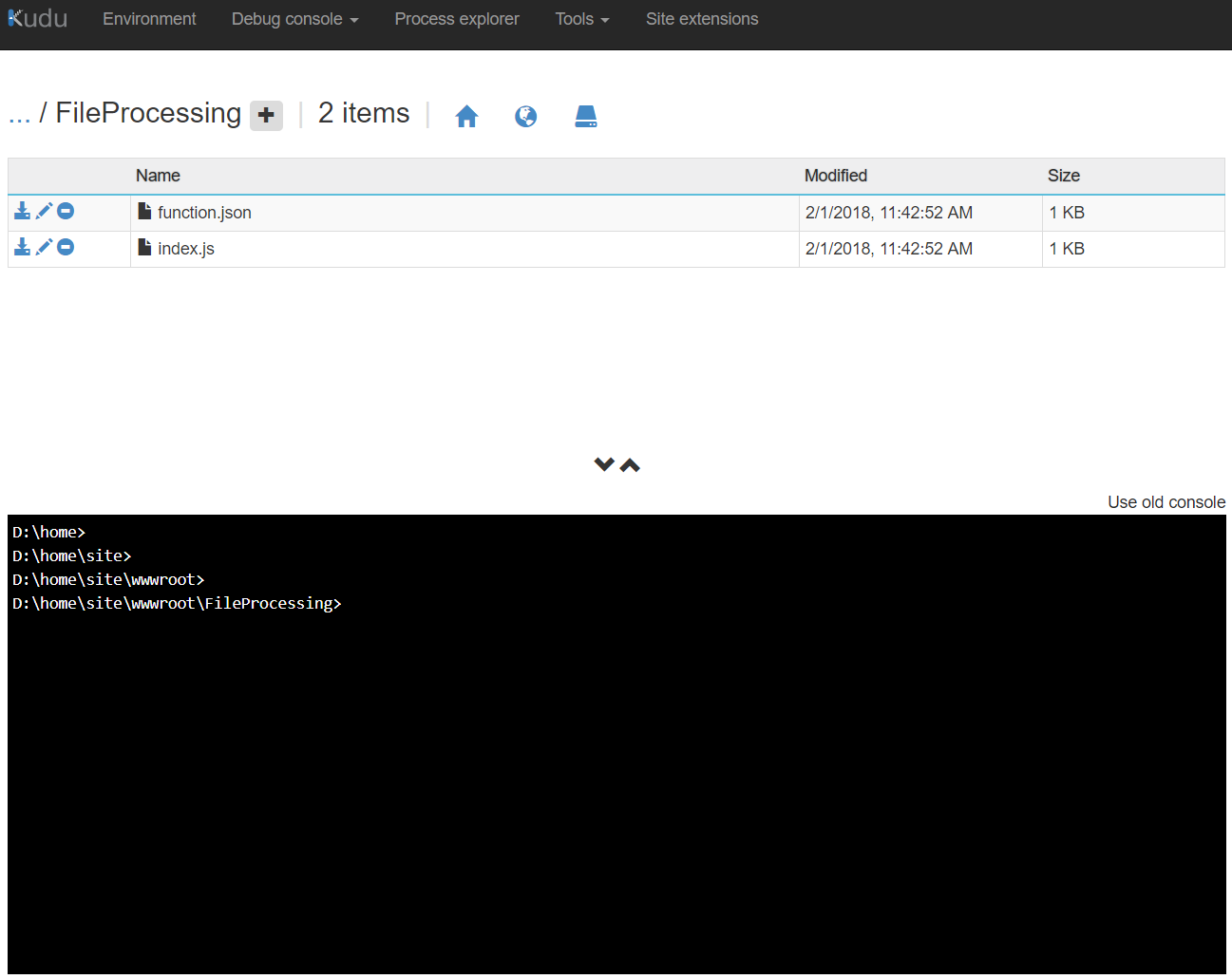


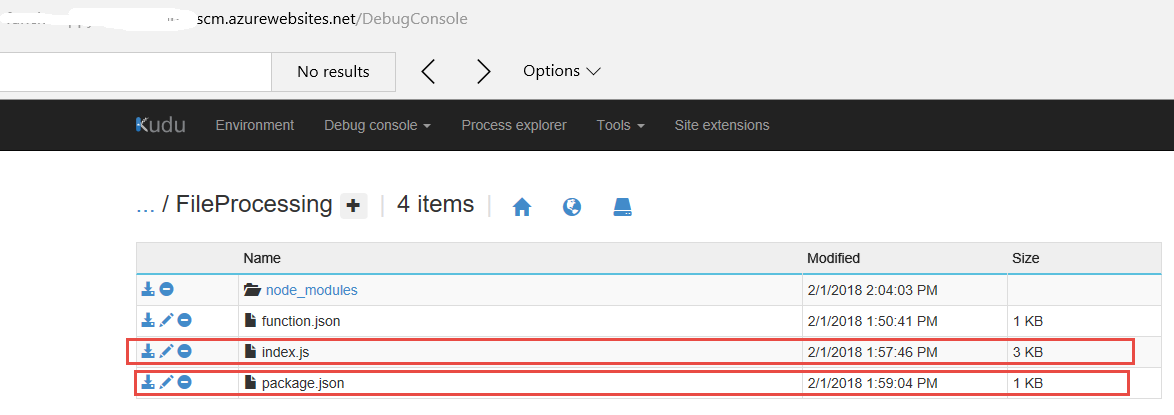
1. Go to the application settings and find the setting named **CosmosDBConnectionString**. Paste the connection string value you copied in a previous step into the CosmosDBConnectionString value.

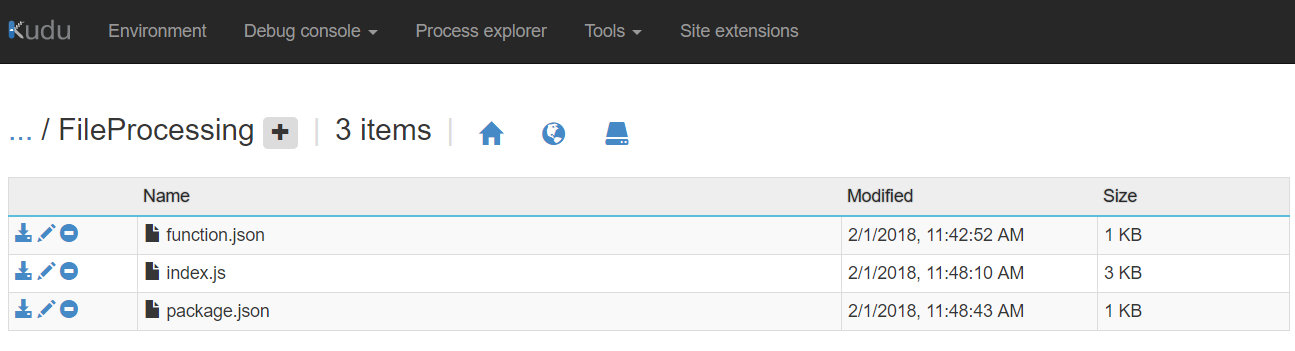
**For example:**



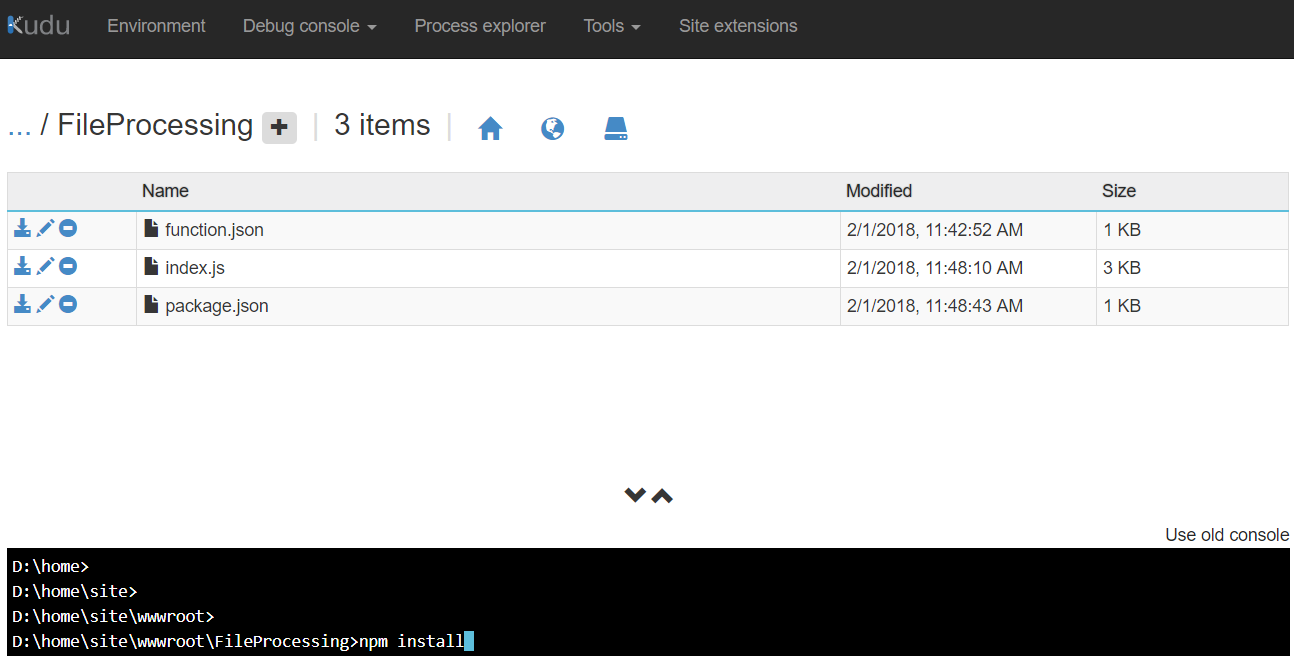
1. Save the settings.
2. Refer to [this article](https://docs.microsoft.com/en-us/azure/azure-functions/functions-reference#fileupdate) to open a debug console and navigate to the D:\home\site\wwwroot\FileProcessing folder.

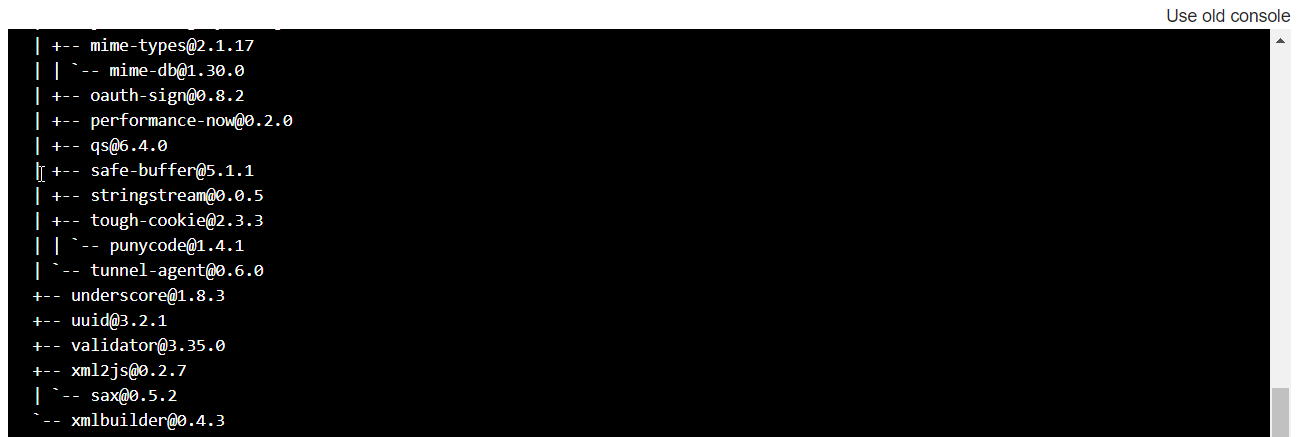


1. Replace the index.js file shown in the screenshot with the **<sourcecodepath>/assets/index.js** file in this sample’s source code. You can easily do this by dragging and dropping the index.js file from your local file system into the web page. 
2. Add the package.json file in this sample’s source code. It is found in this file path. **<sourcecodepath>/assets/package.json**. You can easily do this by dragging and dropping the package.js file from your local file system into the web page.



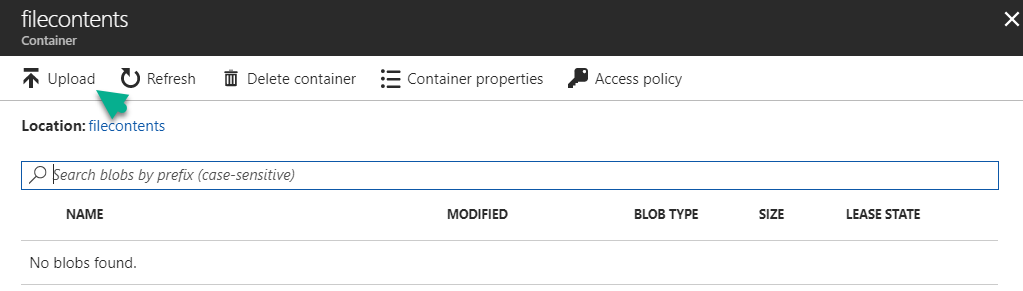
1. In the debug console, make sure you are in the FileProcessing folder.
2. Execute the **npm install** command in the debug console (in the FileProcessing folder) to install the required packages.

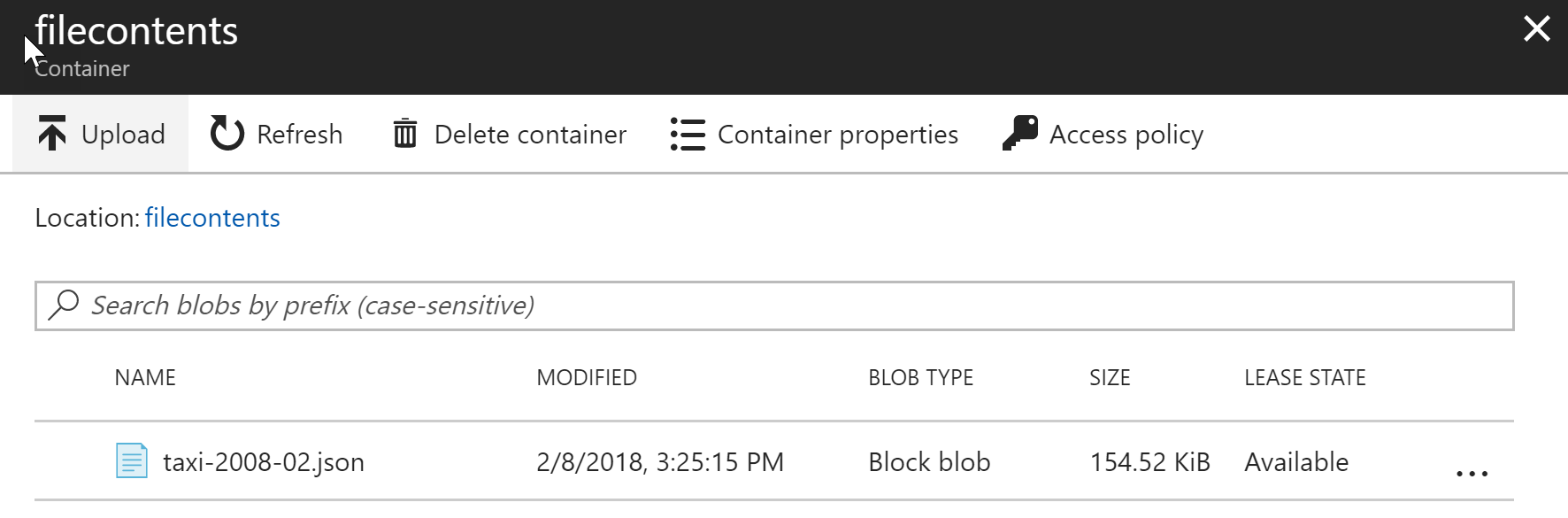




## Validate Installation and Configuration

1. Upload the **<sourcecodepath>/assets/taxi-2008-02.json** file to the **filecontents** container in the Azure Blob service you just created.





1. Finally, go to Cosmos DB Data explorer to view the data from the JSON file in the Cosmos DB.

**For example:**

