



Lab: Create and deploy chaincode to the IBM Blockchain service on Bluemix

Overview

This lab introduces you to chaincode development by showing you how to create a blockchain service in IBM® Bluemix® and how to deploy your first chaincode to the new service.

Tip: This lab shows some screen captures in the IBM Bluemix interface in the classic view. If you log in to Bluemix and want to work in the classic view, click the avatar in the upper right and select **Switch to Classic** at the bottom of the avatar window.

Important: Because the IBM Blockchain service is in beta, it might be temporarily unavailable or at capacity. If you experience problems in the lab when trying to create the IBM Blockchain service or accessing the IBM Blockchain service dashboard, try accessing the dashboard later.

Prerequisites

To complete this lab, be sure you complete the previous two labs “Transfer assets in a business network” and “Explore the IBM Blockchain service on Bluemix.”

You also need:

- A [Bluemix account](#)
- Firefox or Chrome browser

If you want to edit and compile chaincode yourself, you need:

- A GitHub account and tools from <https://github.com/>
- The Go programming language compiler from <https://golang.org/dl/>

Step 1. Create a new IBM Blockchain service on Bluemix

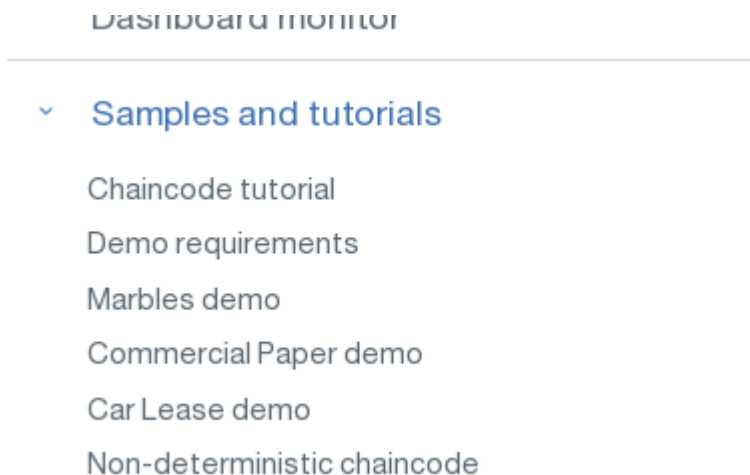
1. Create a new IBM Blockchain service instance in Bluemix for your chaincode. From a browser, log in to <http://www.bluemix.net>.
2. Click **CATALOG** from the top bar.
3. Open the **Application Services** section and click **Blockchain**. Review the service information.




4. Click **Create** to create a new blockchain service. Use the default settings.
The service welcome page provides details about IBM Blockchain Network Service Test Environment. Review this page for more information. For more information about blockchain, see [About blockchain](#).
5. Click the three dots on the right side of the service welcome page and select **View Docs**.



6. Click **Samples and tutorials** in the right navigation.



7. Click **Chaincode tutorial**.

 IBM Bluemix Docs

24

← IBM Blockchain

Documentation

Learn chaincode tutorial

This tutorial guides you through using basic building blocks to code an elementary chaincode application. You will incrementally build a working chaincode that creates generic assets for exchanging on a network. Then you will interact with your chaincode through the network API. After completing this tutorial, you will be able to answer the following questions:

- What is chaincode?
- How do I implement chaincode?
- What dependencies exist?
- What are the major functions?
- How do I pass different values to my arguments?
- How do I securely enroll a user on my network?
- How do I compile my chaincode?
- How do I interact with my chaincode through the REST API?

▼ Samples and tutorials

[Chaincode tutorial](#)

[Demo requirements](#)

[Marbles demo](#)

[Commercial Paper demo](#)

[Car Lease demo](#)

[Non-deterministic chaincode](#)

Getting support

Continue to the next page in this lab and **Step 2** for instructions on how to proceed through the chaincode tutorial.

Step 2. Follow the Bluemix tutorial to edit your first chaincode

There are two ways you can approach this tutorial:

If you are a developer who wants to write chaincode

You should follow all the steps. Install the Go compiler and use a Git account and tools to change and deploy your code. If you have created your Blockchain service after November 10th, you will have a Hyperledger Fabric version that is based on 0.6 and you should follow the steps in the tutorial for the Hyperledger Fabric 0.6 codebase, which will use version 2.0 of the learn chaincode project. At a high level, the tutorial guides you through these steps:

1. Set up the environment.
2. Set up GitHub and fork the example “learn chaincode” repository.
3. Clone the “learn chaincode” fork to your local system; be sure you clone the forked version and not the original.
4. Edit the chaincode_start.go file to add a new capability to invoke and query the chaincode.
5. Commit your local changes with the command `git commit -a`
6. Upload your changes back to your forked repository with the command `git push`

Tip: If you accidentally clone the original “Hello Chaincode” repository, you can use the following command to point to your fork instead of the original:

```
git remote set-url origin https://github.com/<yourGitID>/learn-chaincode
```

7. Interact with your changes by using the Swagger API to deploy, invoke, and query the new chaincode that you made.

Tip: You can use gopkg.in to provide a URL for the chaincode deploy that uses a GitHub branch version:

```
http://gopkg.in/<yourGitID>/learn-chaincode.v2/finished
```

If you only want to see how chaincode works

You don't need to follow the steps for setting up Go and using Git to make changes. Instead, read the earlier sections of the tutorial and simply deploy and work with the finished version of the chaincode directly when you get to the section in the tutorial “Deploying your chaincode.” Use this path if you have created the Blockchain service after November 10, 2016:

```
http://gopkg.in/ibm-blockchain/learn-chaincode.v2/finished
```