# IBM Blockchain

## Three introductory labs

https://ibm.biz/BdsXmj

Lennart Frantzell, Syed Zaidi
October 30, 2016

# Blockchain agenda

1. Double-entry bookkeeping and the business background to the Blockchain
2. The Hyperledger Project, Blockchain architecture, concepts and terminology
3. Three Blockchain hands-on labs
4. Where do we go from here?

# Blockchain intro

https://www.ibm.com/developerworks/cloud/library/cl-blockchain-basics-intro-bluemix-trs/index.html

https://www.ibm.com/developerworks/cloud/library/cl-ibm-blockchain-101-quick-start-guide-for-developers-bluemix-trs/index.html **IBM Blockchain 101: Quick-start guide for developers**

# Section 1:
# Double-entry bookkeeping and the business background to the Blockchain

# Double-entry bookkeeping and the role of business ledgers

LEDGER

| ACCOUNT TYPE | CASH | | | | |
|---|---|---|---|---|---|
| TRANSACTION DATE | TRANSACTION DETAIL | REFERENCE | DEBIT | CREDIT | BALANCE |
| 1/1/16 | Expenses for Jan | Ref#1 | $100.00 | | $100.00 |
| 2/1/16 | Tax withheld | Ref#2 | | $110.00 | ($10.00) |

**Luca Bartolomeo de Pacioli**

*Portrait of Luca Pacioli*, traditionally attributed to Jacopo de' Barbari, 1495 (attribution controversial).[1]

- Business transactions typically involve various participants like buyers, sellers, and intermediaries (such as banks, auditors, or notaries) whose business agreements and contracts are recorded in ***business ledgers***.
- A business typically uses multiple ledgers to keep track of asset ownership and asset transfers between participants in its various lines of businesses.
- Ledgers are the systems of record (SORs) for a business's economic activities and interests.

# Problems with current business ledgers

Current business ledgers in use today are deficient in many ways:

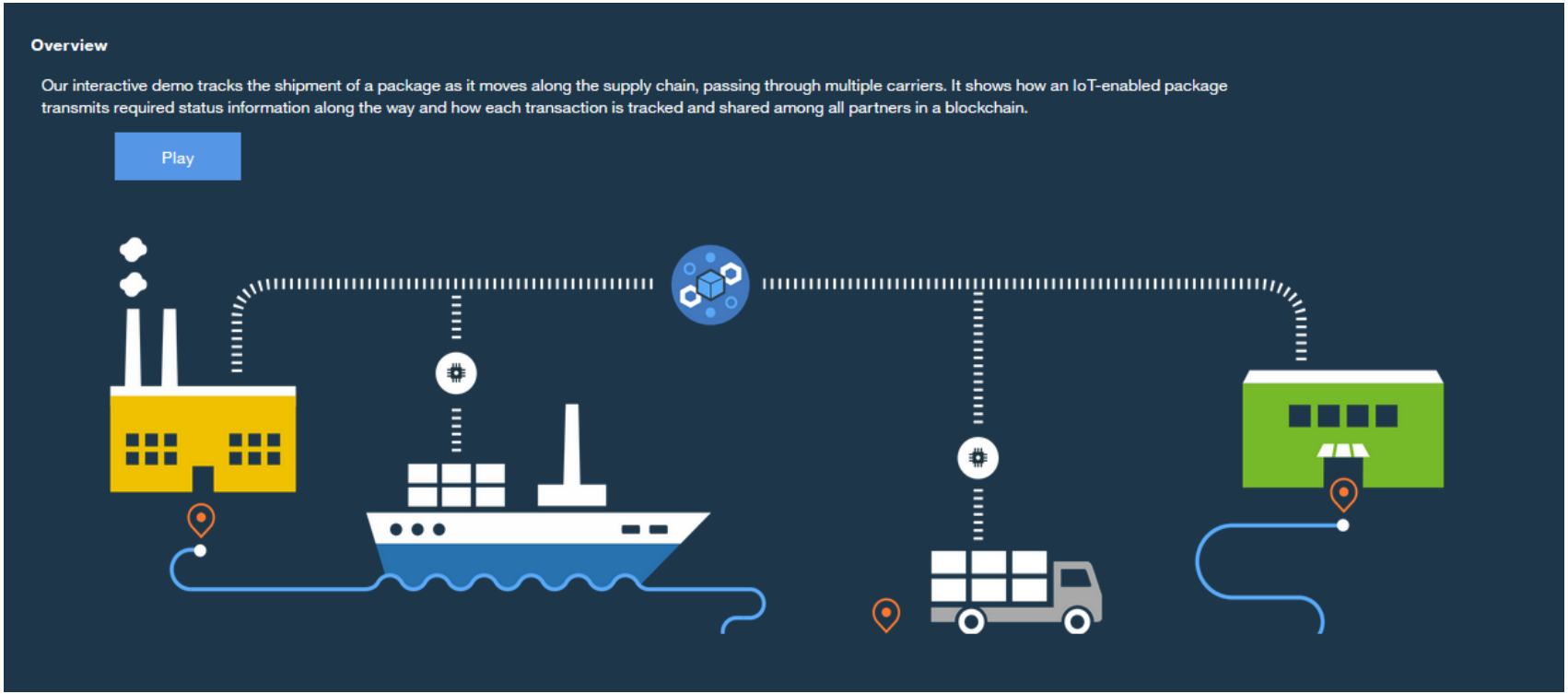
- They are inefficient, costly, non-transparent, and subject to fraud and misuse.
- These problems stem from centralized, trust-based, third-party systems,
- such as financial institutions, clearing houses, and other mediators of existing institutional arrangements.
- Centralized, trust-based ledger systems lead to bottlenecks and slowdowns of
- transaction settlements.
- Lack of transparency, as well as susceptibility to corruption and fraud, lead to disputes.
- Having to resolve disputes and possibly reverse transactions or provide Insurance for transactions is costly.

……

## Lets look at some of the problems that Blockchain can solve

https://www.youtube.com/watch?v=IgNfoQQ5Reg IBM Blockchain Car Lease Demo
https://www.youtube.com/watch?v=F0P7NM7d-ps Blockchain in IBM Global Financing
https://www.youtube.com/watch?v=TZjs-BYiE5I&feature=youtu.be Blockchain on IBM z Systems for Finance
https://www.youtube.com/watch?v=r0LsnzAe1Yg International Trade Solution on Blockchain
https://www.youtube.com/watch?v=B2eYh2Y7rJw&feature=youtu.be Trading Commerical Paper via blockchain with IBM Bluemix

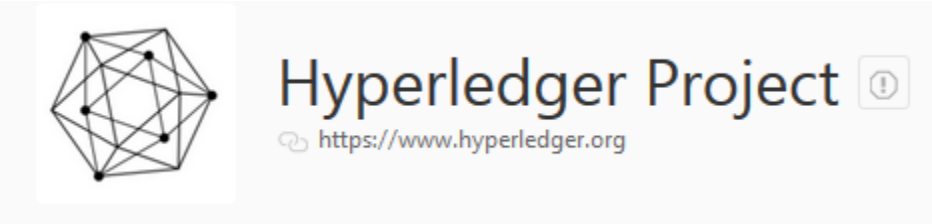
# Blockchain intro




https://www.ibm.com/internet-of-things/iot-news/announcements/private-blockchain/

# Section 2:
# The Hyperledger project, conceps and technologies

# The Hyperledger project




https://github.com/hyperledger

# The Hyperledger project

https://www.hyperledger.org/


https://twitter.com/Hyperledger

https://www.youtube.com/watch?v=EKa5Gh9whgU Building a blockchain for business with the Hyperledger Project

# The Hyperledger project

- A cross-industry collaborative effort, started in December 2015 by the Linux Foundation to support blockchain-based distributed ledgers

- It is focused on ledgers designed to support global business transactions, including major technological, financial, and supply chain companies, with the goal of improving many aspects of performance and reliability.

- The project aims to bring together a number of independent efforts to develop open protocols and standards, by providing a modular framework that supports different components for different uses. This would include a variety of blockchains with their own consensus and storage models, and services for identity, access control, and contracts.

https://twitter.com/Hyperledge
https://www.youtube.com/watch?v=EKa5Gh9whgU Building a blockchain for business with the Hyperledger Project

# The Hyperledger project

The Hyperledger Project defined a protocol specification known as **Open Blockchain Protocol Specification** to create a blockchain fabric for use in a variety of industry use-cases involving B2B and B2C transactions. The primary goals of this effort are:

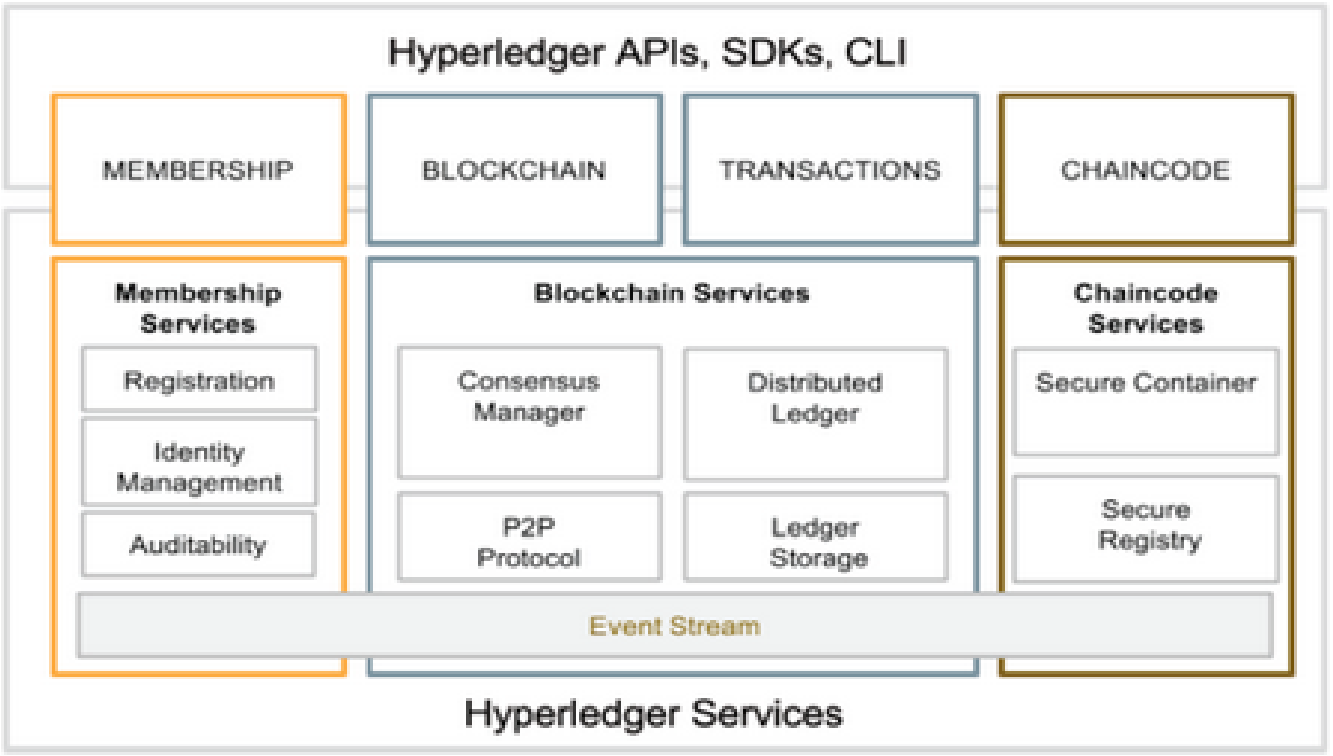
- Support a wide variety of industrial use cases with different requirements
- Comply with regulatory regimes that exist today
- Support
  - verified identities, private and confidential transactions
  - permissioned, shared ledger
  - performance, scaling, auditability, identity, security, and privacy
  - reduction of costly computations involved in proof of work

Hyperledger's Blockchain fabric use the following concepts as its underpinnings:

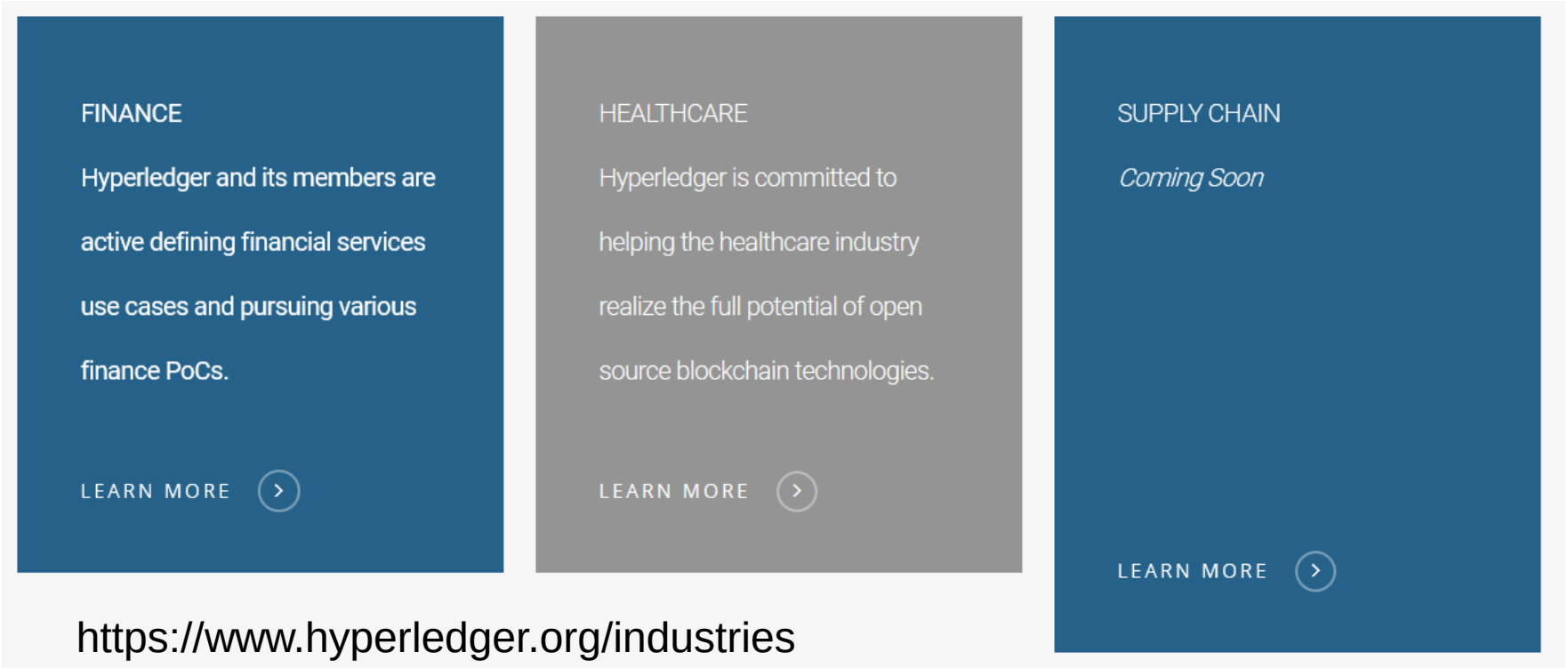
- Smart contracts
- Digital assets
- System of record repositories/stores
- Decentralized consensus-based network
- Pluggable consensus algorithms/models
- Cryptographic security

https://github.com/OpenAssets/open-assets-protocol

# The Hyperledger project



# The Hyperledger project




https://www.hyperledger.org/industries

# Concepts and technologies

https://www.ibm.com/blockchain/what_is_blockchain.html

**Before**

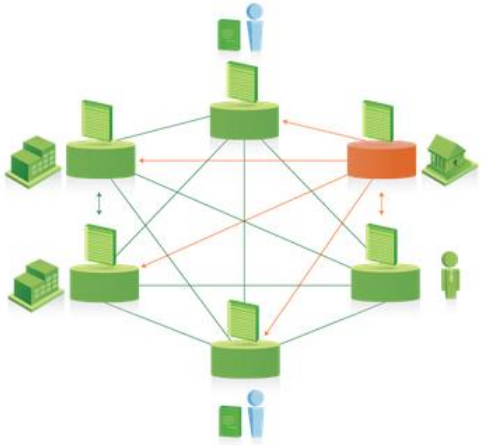
Asset ownership and transfer between businesses is currently inefficient, slow, costly and vulnerable to manipulation.
Everyone has their own ledger where discrepancies between business parties can increase settlement times.
A new way is needed for Internet-age market enablement.

# Concepts and technologies

https://www.ibm.com/blockchain/what_is_blockchain.html

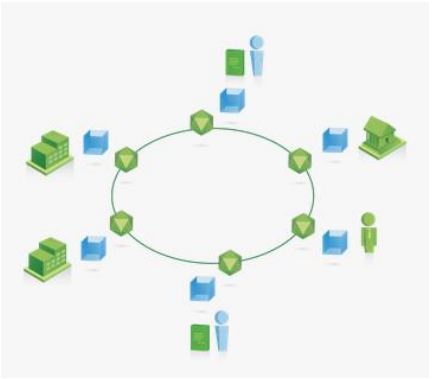
**Before**

Asset ownership and transfer between businesses is currently inefficient, slow, costly and vulnerable to manipulation.
Everyone has their own ledger where discrepancies between business parties can increase settlement times.
A new way is needed for Internet-age market enablement.



# Concepts and technologies

https://www.ibm.com/blockchain/what_is_blockchain.html



**After**

- Blockchain technologies can be used to share a ledger across
- the business network.
- The network will be private to the parties concerned, permissioned so only authorized parties are allowed to join
- Secure using cryptographic technology to ensure that participants only see what they are allowed to see.
- The shared ledger will be more robust, since it is replicated and distributed.
- All transactions against the ledger will require consensus across the network, where provenance of information is clear and transparent.
- Transactions will be immutable (unchangeable) and final.

# Concepts and technologies: Shared ledger

https://www.ibm.com/blockchain/what_is_blockchain.html

**Shared ledger**

It can act as a source of truth for businesses doing transactions on a blockchain:

- Records all transactions across the business network
- Is shared among participants
- Is replicated so each participant has their own copy
- Is permissioned, so participants see only appropriate transactions

Often, companies have multiple ledgers for multiple business networks in which they participate. It can be used for recording and totaling financial transactions.

# Concepts and technologies: Smart contracts

https://www.ibm.com/blockchain/what_is_blockchain.html

**Smart contracts**

A smart contract can include a digital asset which is anything that has an owner and can be converted into value. Digital assets can be tangible or intangible. A smart contract can also include a digital representation of a set of business rules:

- Is embedded in the blockchain
- Is executed in a transaction
- Is verifiable, signed, and encoded in a programming language
- For example, it defines conditions under which corporate bond transfer occurs

# Concepts and technologies: Consensus

https://www.ibm.com/blockchain/what_is_blockchain.html

**Consensus**

Entries in the ledger are synchronized to all ledgers in the network. Consensus ensures that these shared ledgers are exact copies, and lowers the risk of fraudulent transactions since tampering would have to occur across many places at the exact same time.

- All parties agree to the transaction and validate it via the peer network.
- Rules can also be established to validate transactions.
- This trusted and trustless participation makes commitment possible at a low cost.

IBM Blockchain uses a “pluggable” consensus system to meet the needs of different industry segments.

# Concepts and technologies: Privacy and Confidentiality

https://www.ibm.com/blockchain/what_is_blockchain.html

**Privacy and confidentiality**

- Ability to protect records with a personal digital signature —
  - the blockchain generates a private and public key to seal that record.
- It is encrypted, hashed, and sent to the network of validating nodes
- Unique IDs for customer, invoice and reference numbers
- Although the ledger is shared, sometimes participants require:
  - Private transactions
- Identities that cannot be linked to specific transactions
- Transactions need to be authenticated, and cryptography is central to these processes.

# Concepts and technologies: Privacy and Confidentiality

https://www.ibm.com/blockchain/what_is_blockchain.html

**Privacy and confidentiality**

- Ability to protect records with a personal digital signature —
  - the blockchain generates a private and public key to seal that record.
- It is encrypted, hashed, and sent to the network of validating nodes
- Unique IDs for customer, invoice and reference numbers
- Although the ledger is shared, sometimes participants require:
  - Private transactions
- Identities that cannot be linked to specific transactions
- Transactions need to be authenticated, and cryptography is central to these processes.

# Concepts and technologies: Chaincode

https://github.com/IBM-Blockchain/learn-chaincode

Chaincode is Go (GoLang) or Java code that enables users to interact with a blockchain network. Whenever you 'invoke' a transaction on the network, you are calling a function in chaincode that reads and writes values to the ledger.

# Chaincode Hello World

https://github.com/ALT-F1/chaincode-helloworld/blob/master/helloworld.go

```
/*
The MIT License (MIT)
*/
....
// HelloWorld example simple Chaincode implementation
type HelloWorld struct {
}

// Init is called during Deploy transaction after the container has been
func (t *HelloWorld) Init(stub *shim.ChaincodeStub, function string, args []string) ([]byte, error) {
        .....
}
// Invoke is called for every Invoke transactions. The chaincode may change its state variables
func (t *HelloWorld) Invoke(stub *shim.ChaincodeStub, function string, args []string) ([]byte, error) {
        fmt.Printf("HelloWorld in Invoke with function %s!\n", function)
        if function != "invoke" {
                return nil, errors.New("Invalid invoke function name. Expecting \"invoke\"")
        }
        ......}
```

# Chaincode Hello World

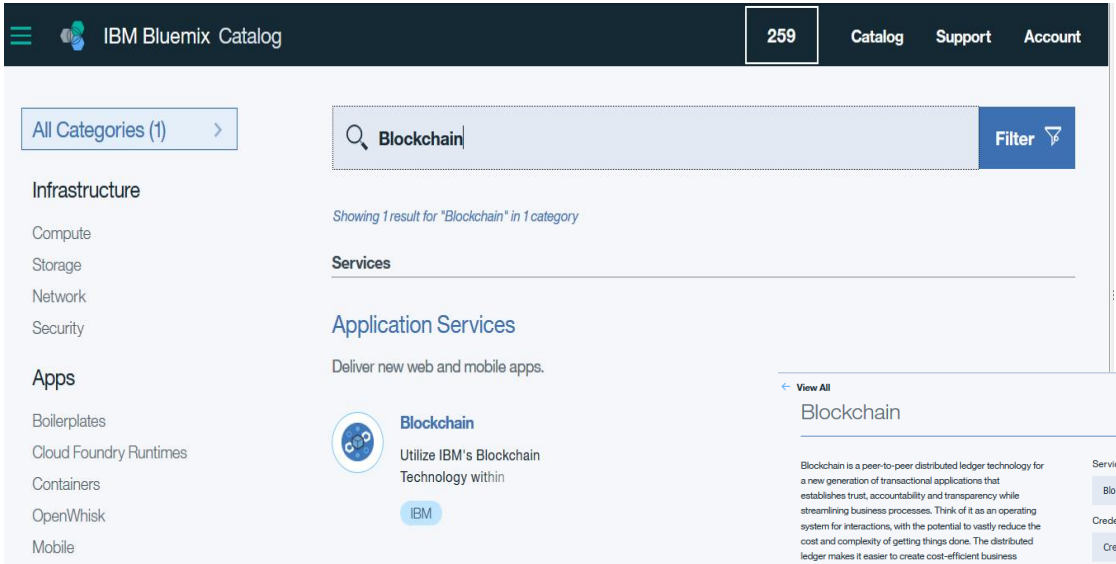
https://github.com/ALT-F1/chaincode-helloworld/blob/master/helloworld.go

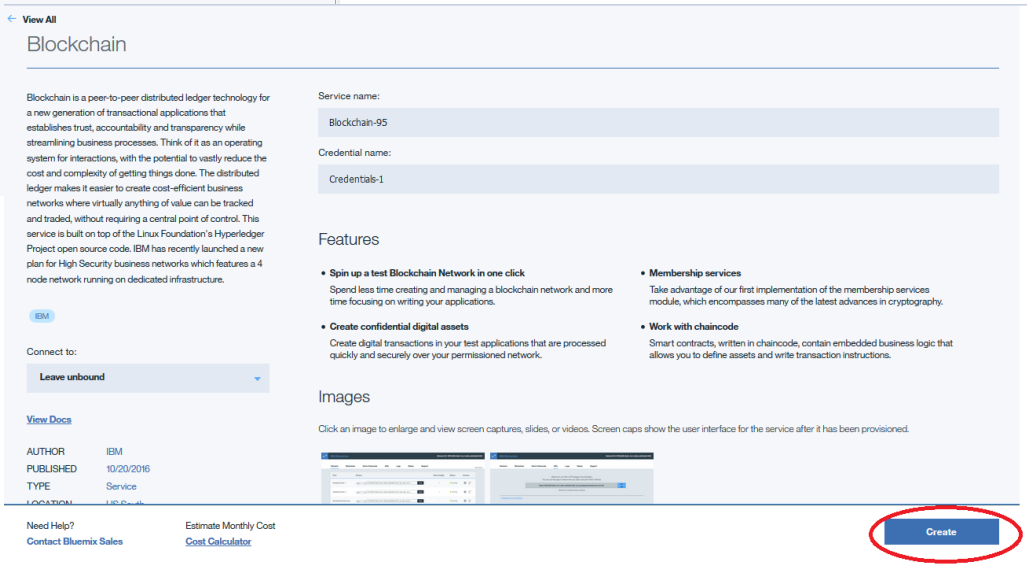
```
// Query is called for Query transactions. The chaincode may only read
// (but not modify) its state variables and return the result
func (t *HelloWorld) Query(stub *shim.ChaincodeStub, function string, args []string) ([]byte, error) {
	myLogger.Debug("HelloWorld Query is called!\n")
	if function != "query" {
		return nil, errors.New("Invalid query function name. Expecting \"query\"")
	}
.....
}

func main() {
	myLogger.Debug("HelloWorld Main\n")
	err := shim.Start(new(HelloWorld))
	if err != nil {
		fmt.Printf("Error starting HelloWorld chaincode: %s\n", err)
	}
}
```

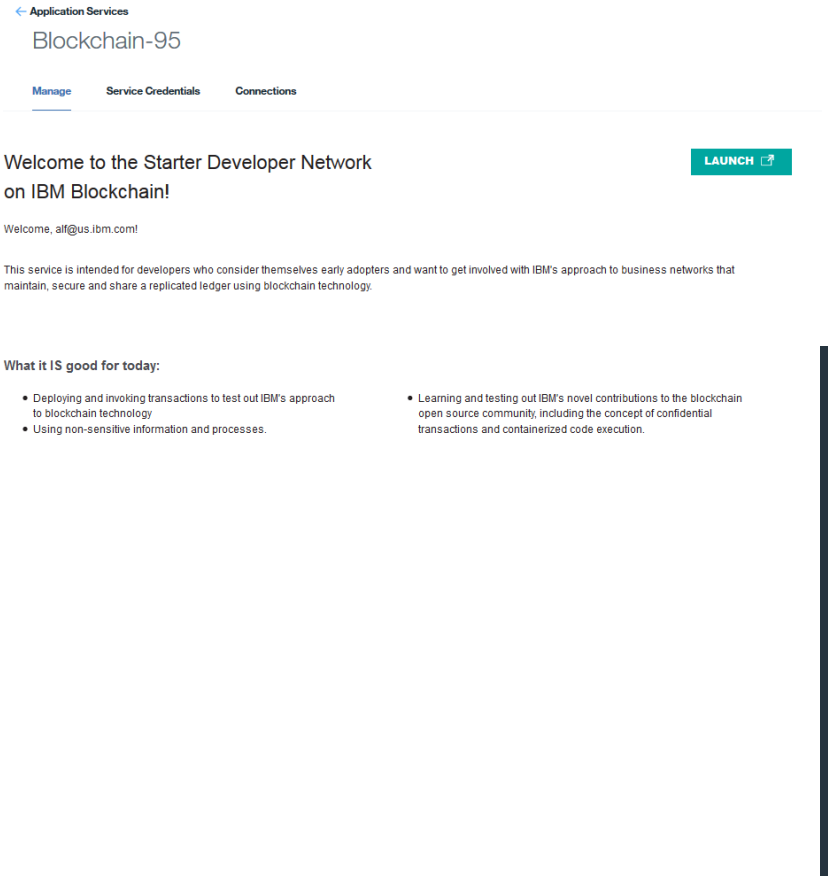
# Section 3:
# Blockchain on Bluemix

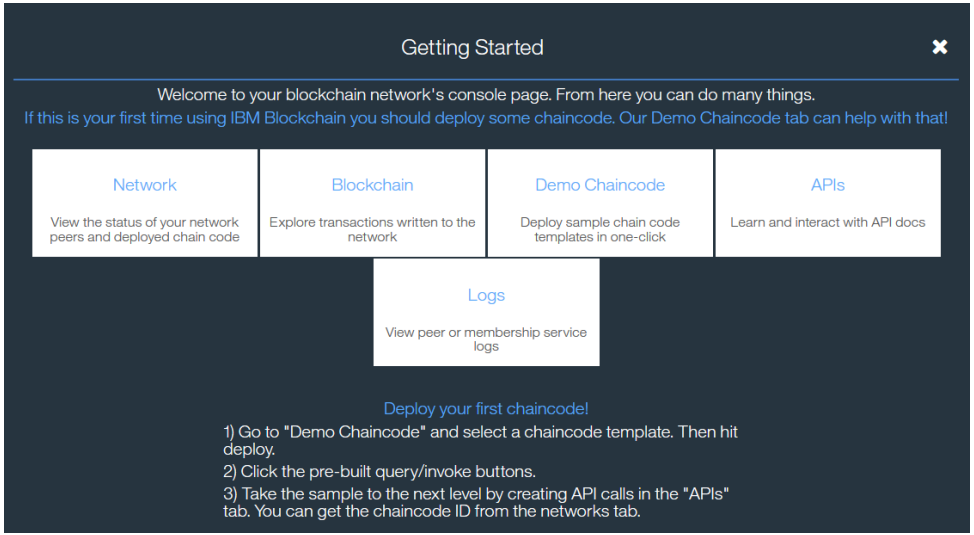
# Blockchain in Bluemix Catalog



# Blockchain in Bluemix Catalog



# Blockchain in Bluemix Catalog

## Blockchain

Blockchain is a peer-to-peer distributed ledger technology for a new generation of transactional applications that establishes trust, accountability and transparency while streamlining business processes. Think of it as an operating system for interactions, with the potential to vastly reduce the cost and complexity of getting things done. The distributed ledger makes it easier to create cost-efficient business networks where virtually anything of value can be tracked and traded, without requiring a central point of control. This service is built on top of the Linux Foundation's Hyperledger Project open source code. IBM has recently launched a new plan for High Security business networks which features a 4 node network running on dedicated infrastructure.

IBM

Connect to:

Leave unbound

View Docs

| | |
|---|---|
| AUTHOR | IBM |
| PUBLISHED | 10/20/2016 |
| TYPE | Service |
| LOCATION | US South |

Service name:

Blockchain-w9

Credential name:

Credentials 1

### Features

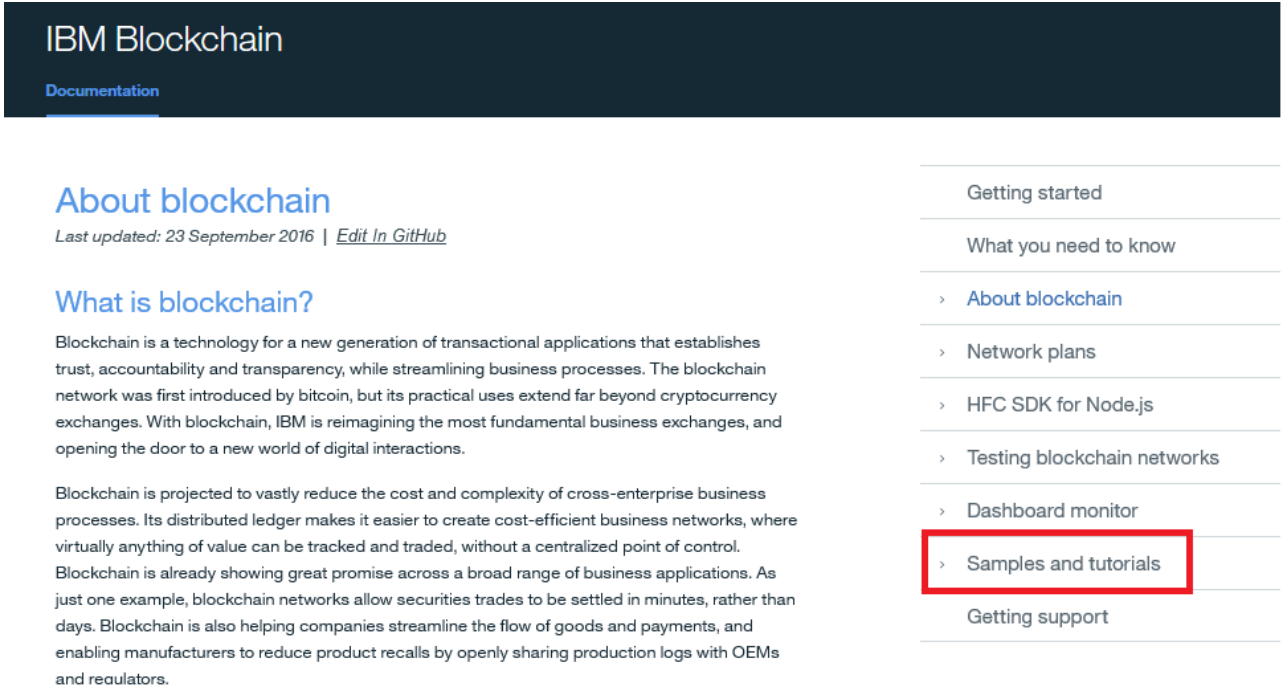
- **Spin up a test Blockchain Network in one click**
  Spend less time creating and managing a blockchain network and more time focusing on writing your applications.
- **Membership services**
  Take advantage of our first implementation of the membership services module, which encompasses many of the latest advances in cryptography.
- **Create confidential digital assets**
  Create digital transactions in your test applications that are processed quickly and securely over your permissioned network.
- **Work with chaincode**
  Smart contracts, written in chaincode, contain embedded business logic that allows you to define assets and write transaction instructions.

### Images

Click an image to enlarge and view screen captures, slides, or videos. Screen caps show the user interface for the service after it has been provisioned.

# Blockchain in Bluemix Catalog

https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_overview.html



# Blockchain in Bluemix Catalog – sample apps the tutorials

https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_tutorials.html

## Sample apps and tutorials

*Last updated: 5 October 2016* | *Edit In GitHub*

The following samples demonstrate how applications and chaincode function in an IBM Blockchain network. To learn more about the Hyperledger Fabric v0.5 code, which underpins your blockchain network, visit the Fabric Docs section of the Linux Foundation's Hyperledger Project.

To experience chaincode applications in action, you can immediately deploy the Marbles, Commercial Paper or Car Lease demo below (click a Deploy to Bluemix button). Or continue reading to explore the Hello Chaincode tutorial.

- Deploy to Bluemix **Marbles**
- Deploy to Bluemix **Commercial Paper**
- Deploy to Bluemix **Car Lease**

## Using the Hello 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:

# Lab 1 Marbles

https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_tutorials.html

View your app

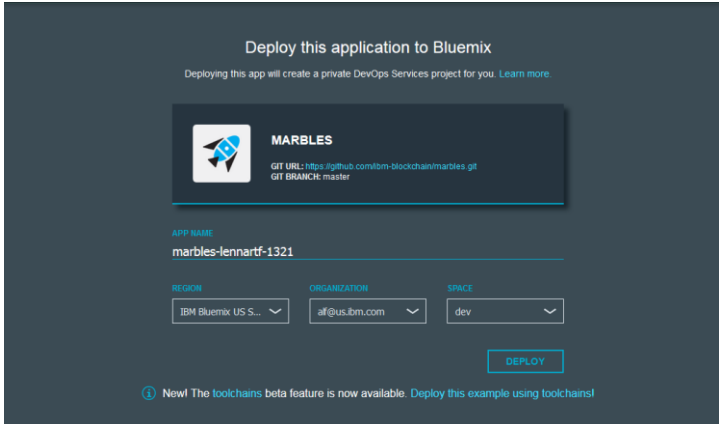
## Sample apps and tutorials

*Last updated: 5 October 2016* | *Edit In GitHub*

The following samples demonstrate how applications and chaincode function in an IBM Blockchain network. To learn more about the Hyperledger Fabric v0.5 code, which underpins your blockchain network, visit the Fabric Docs section of the Linux Foundation's Hyperledger Project.

To experience chaincode applications in action, you can immediately deploy the Marbles, Commercial Paper or Car Lease demo below (click a Deploy to Bluemix button). Or continue reading to explore the Hello Chaincode tutorial.

- Deploy to Bluemix **Marbles**
- Deploy to Bluemix **Commercial Paper**
- Deploy to Bluemix **Car Lease**

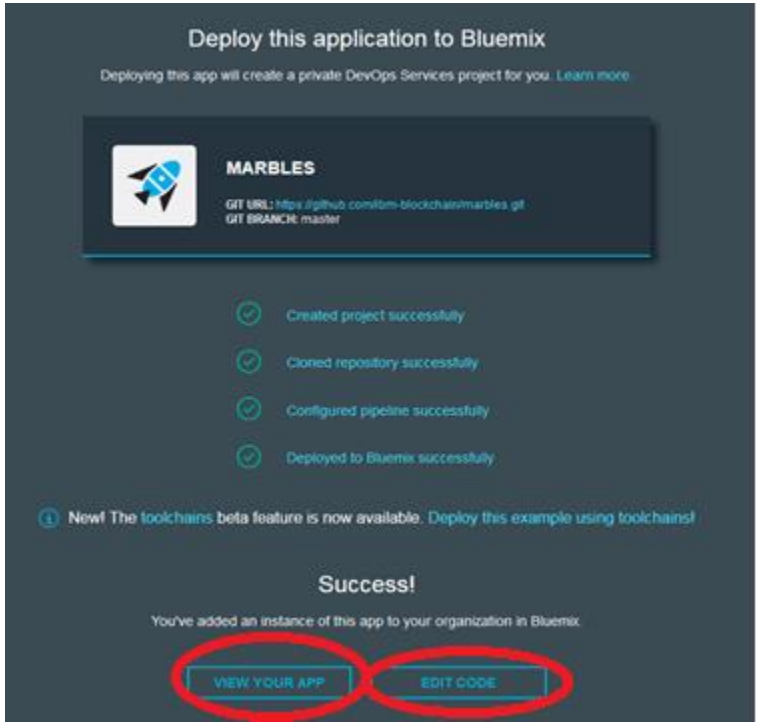


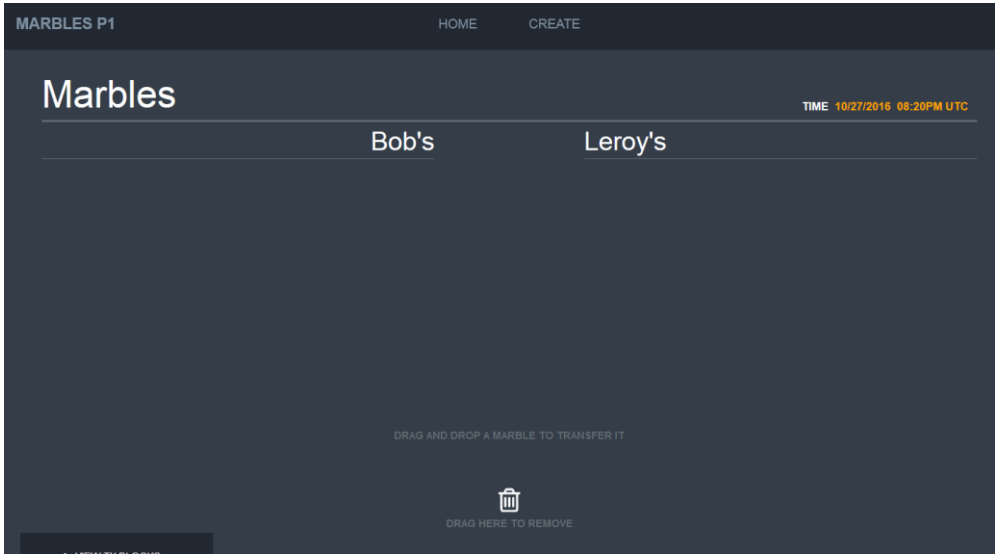

https://github.com/IBM-Blockchain/marbles/blob/master/tutorial_part1.md

# Lab 1 Marbles – View Your App

https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_tutorials.html

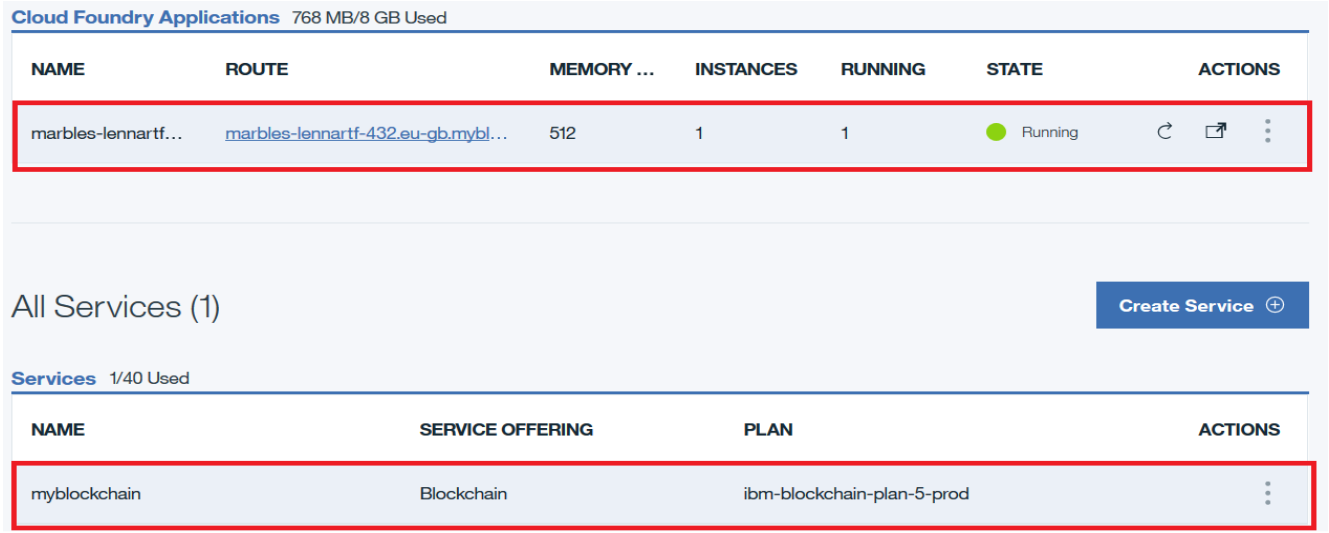
View your app



# Lab 1 Marbles – The underlying architecture

https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_tutorials.html




https://github.com/ibm-blockchain/marbles-chaincode

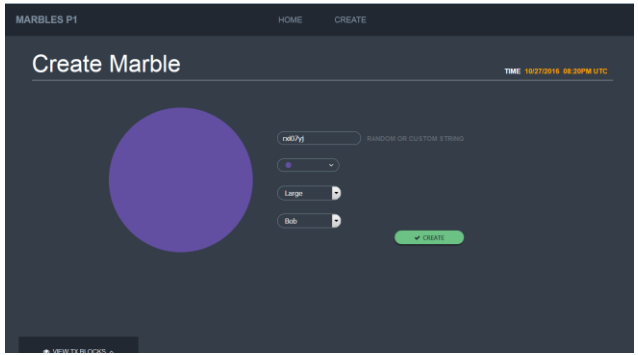
Hyperledger Framework

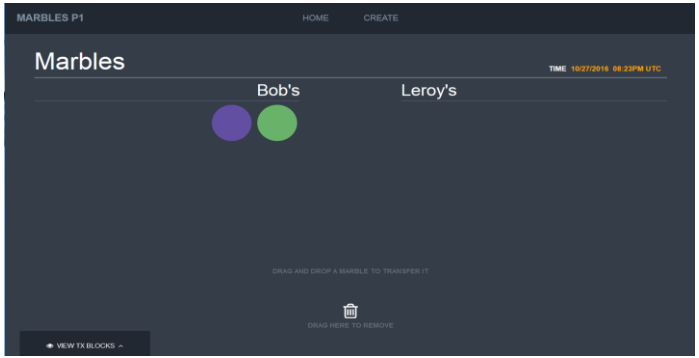
https://github.com/hyperledger/fabric/tree/master/docs

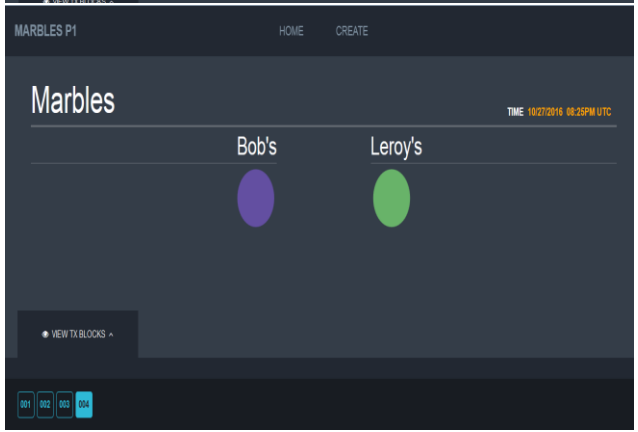
# Lab 1 Marbles – View Your App

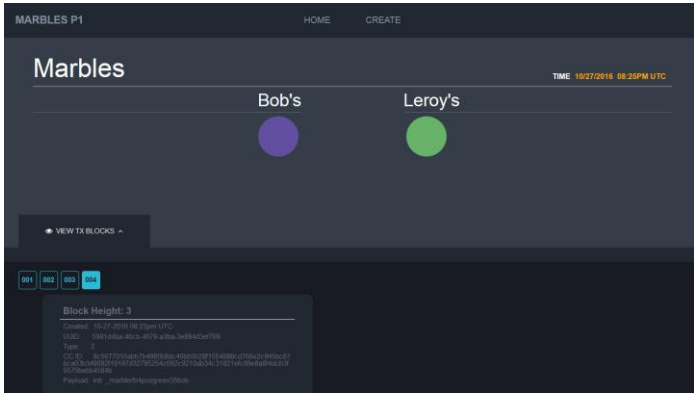
https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_tutorials.html

View your app



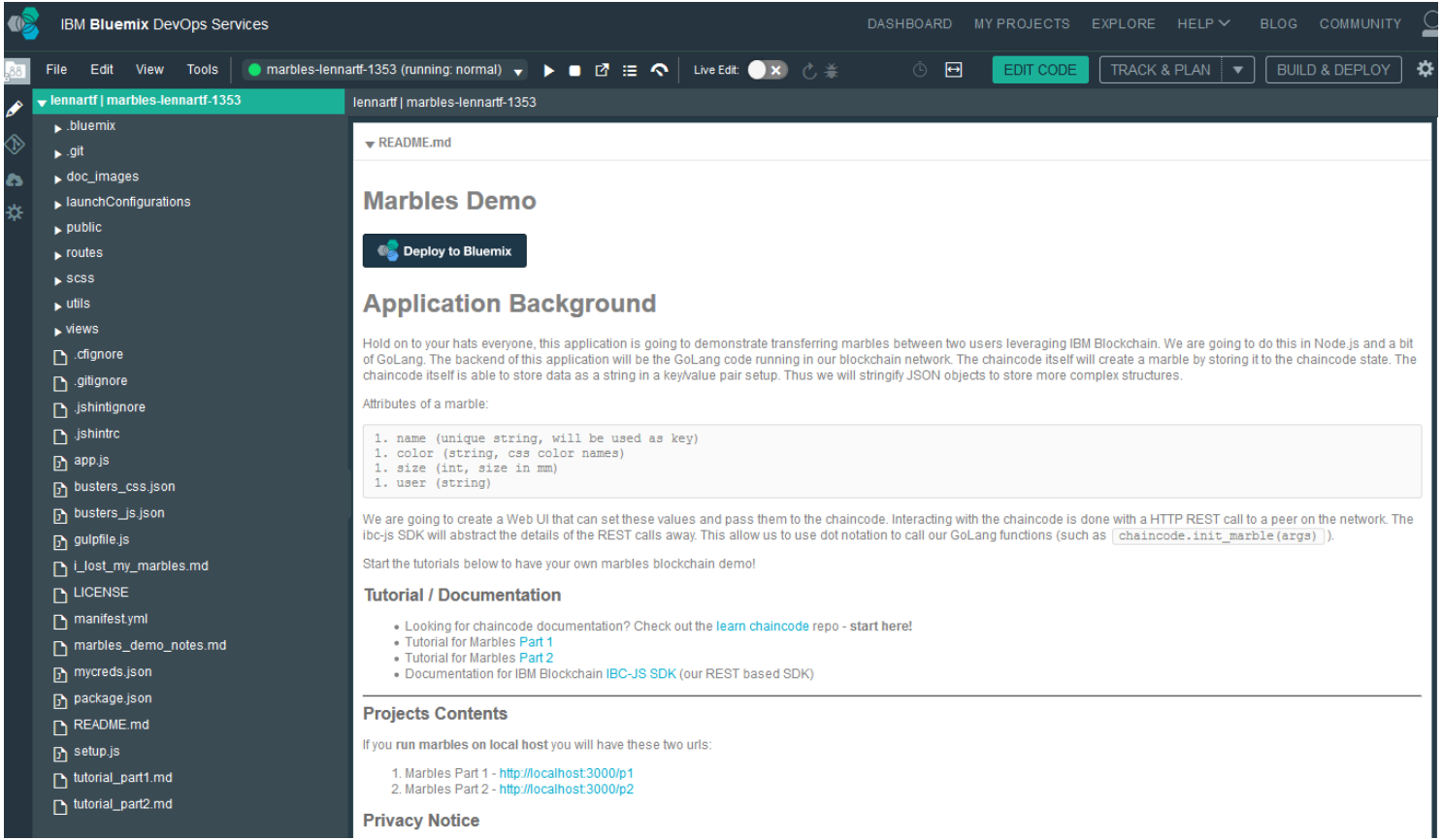



# Lab 1 Marbles – Edit Code

https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_tutorials.html



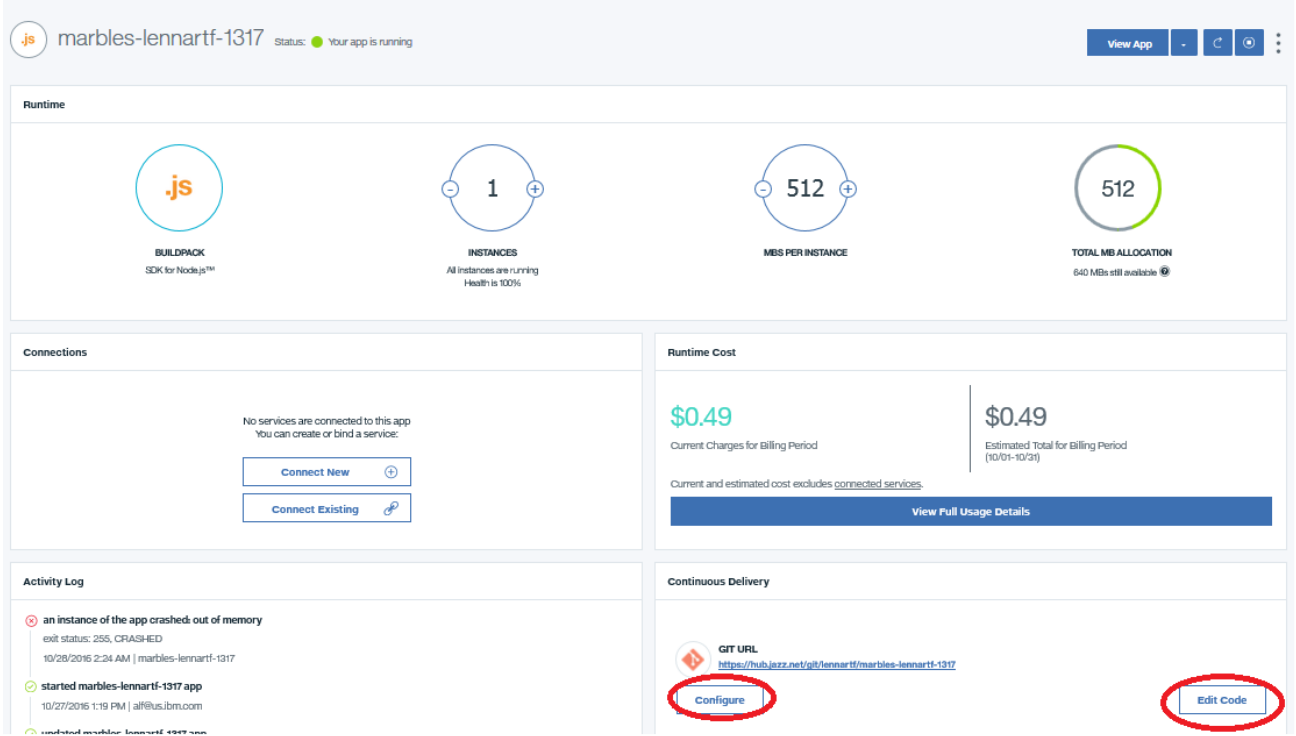
# Lab 1 Marbles – Bluemix console

https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_tutorials.html

| | | | | | | |
|---|---|---|---|---|---|---|
| manifesttest.yml | | 128 | 1 | 1 | Running | |
| marbles-lennartf-1317 | marbles-lennartf-1317.mybluemix.net | 512 | 1 | 1 | Running | |
| MobileFoundation-r... | mobilefoundation-rk-fe-server.myblu... | 1024 | 1 | 1 | Running | |
| NodeRedNI | NodeRedNI.mybluemix.net | 512 | 1 | 1 | Running | |
| outthink | outthink.mybluemix.net | 128 | 1 | 1 | Running | |

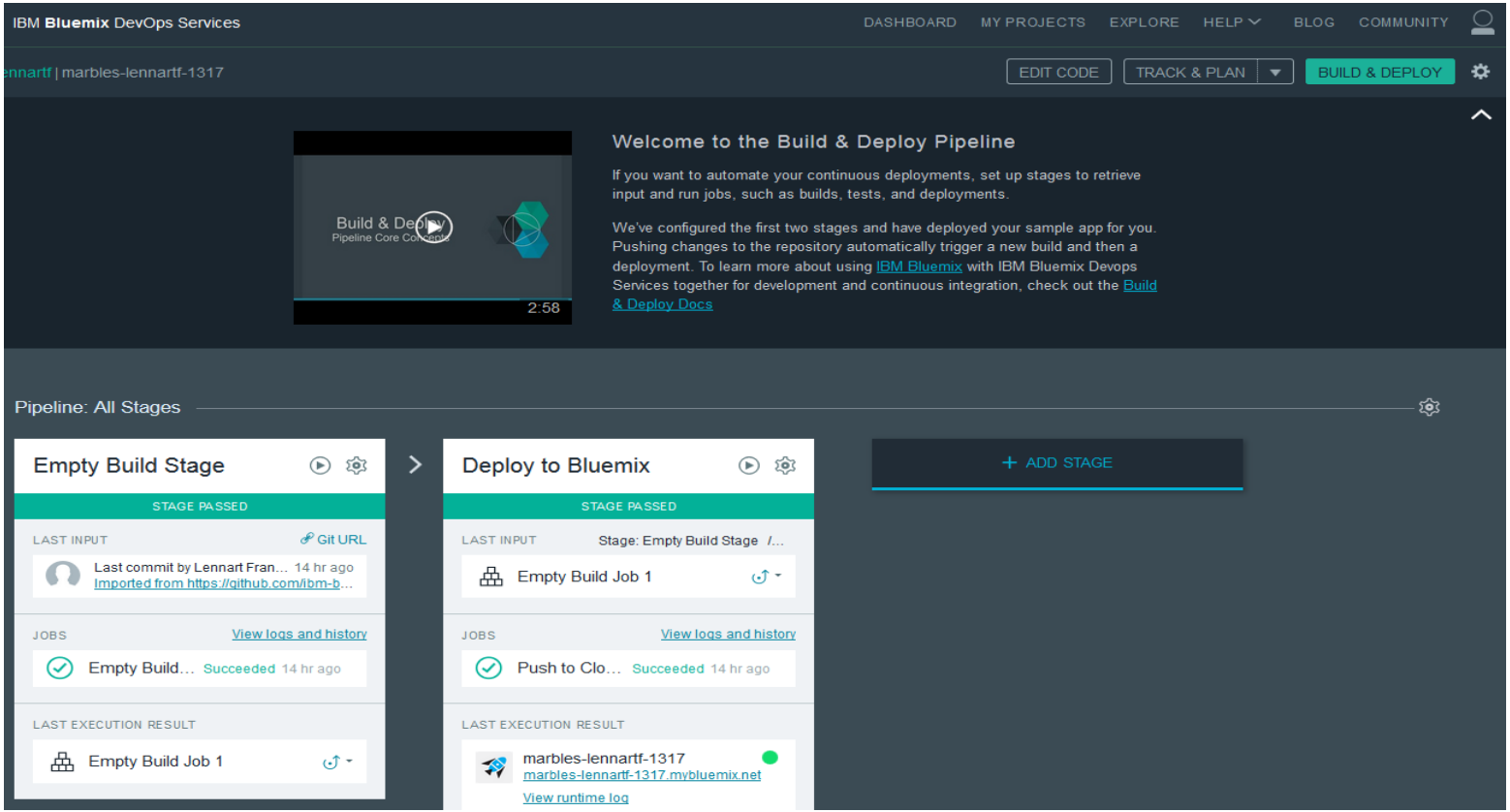
# Lab 1 Marbles – Bluemix console

https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_tutorials.html



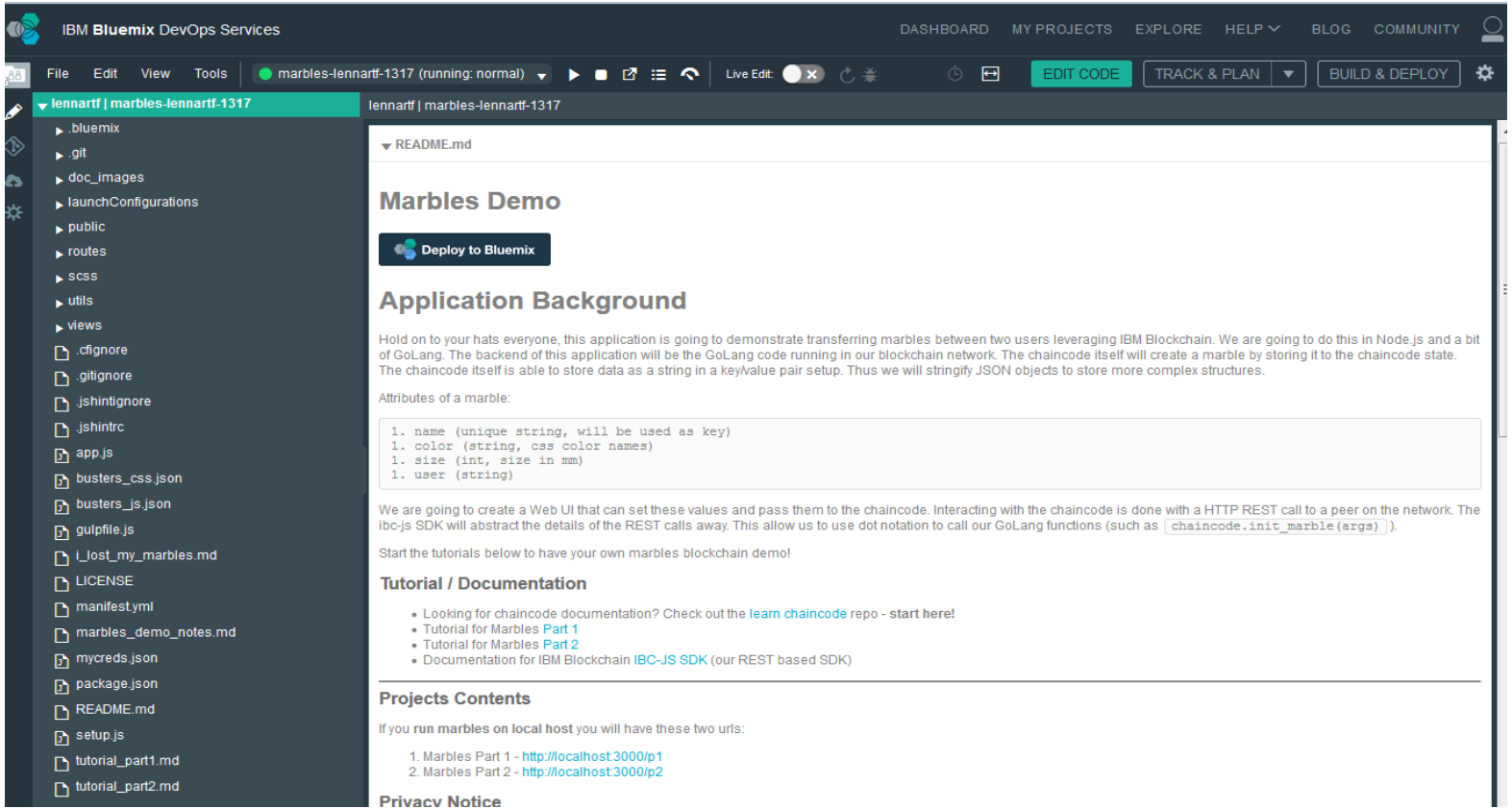
# Lab 1 Marbles – Devops Services Build and Deploy

https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_tutorials.html



# Lab 1 Marbles – DevOps Services Edit Code – Build and Deploy

https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_tutorials.html

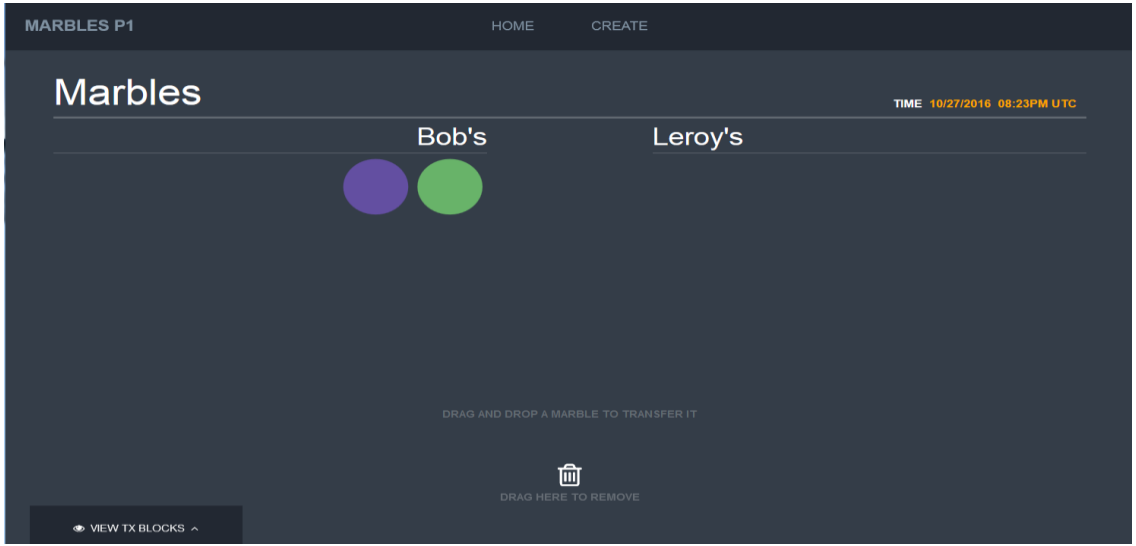


# Lab 1 Marbles – Console

https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_tutorials.html

# Lab 1 Marbles – View Your App

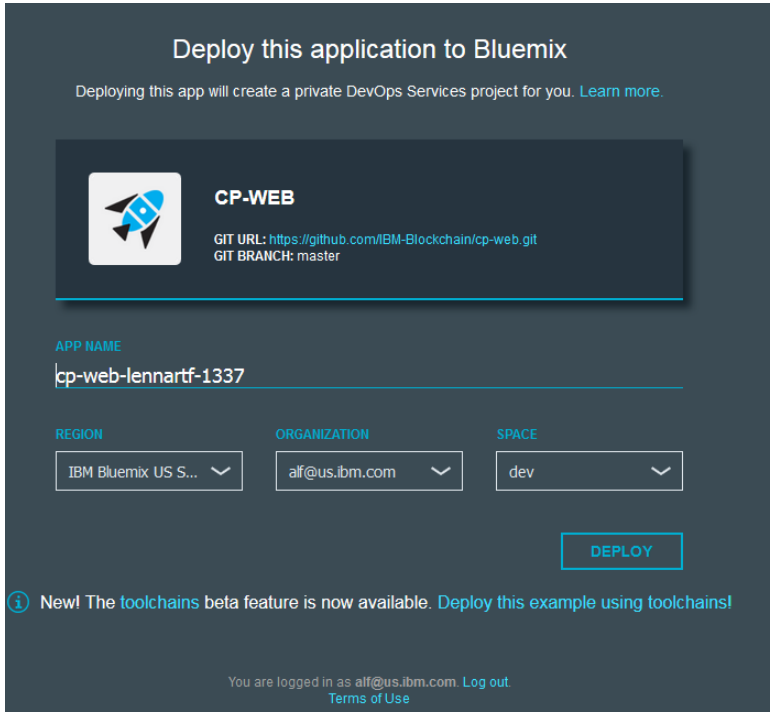
https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_tutorials.html

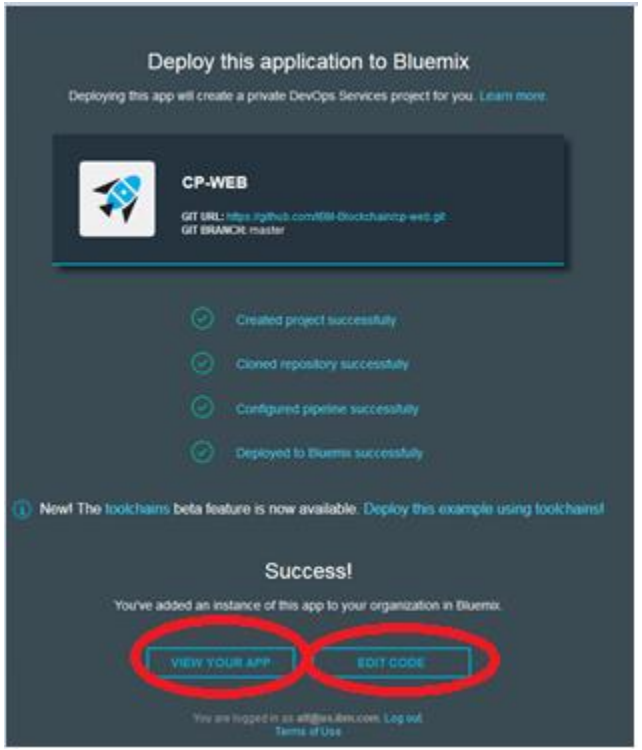


# Lab 2 Commercial Paper

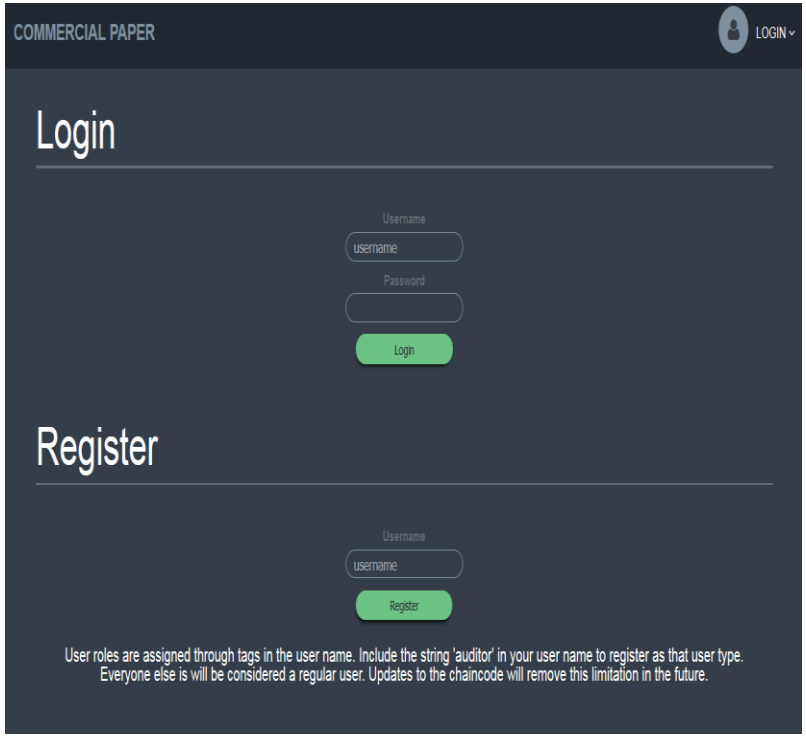
https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_tutorials.html

View your app

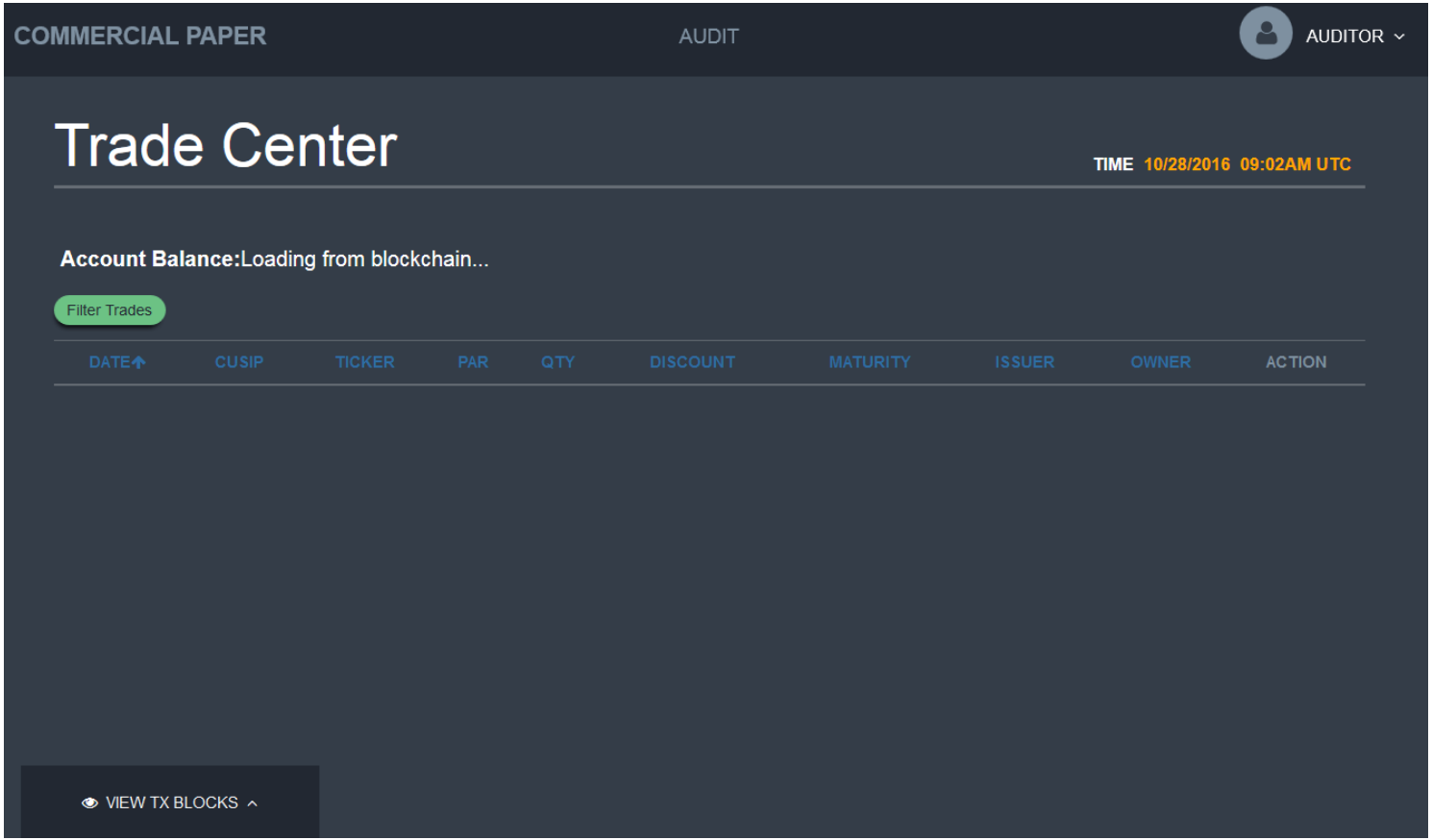


# Lab 2 Commercial Paper – View Your App



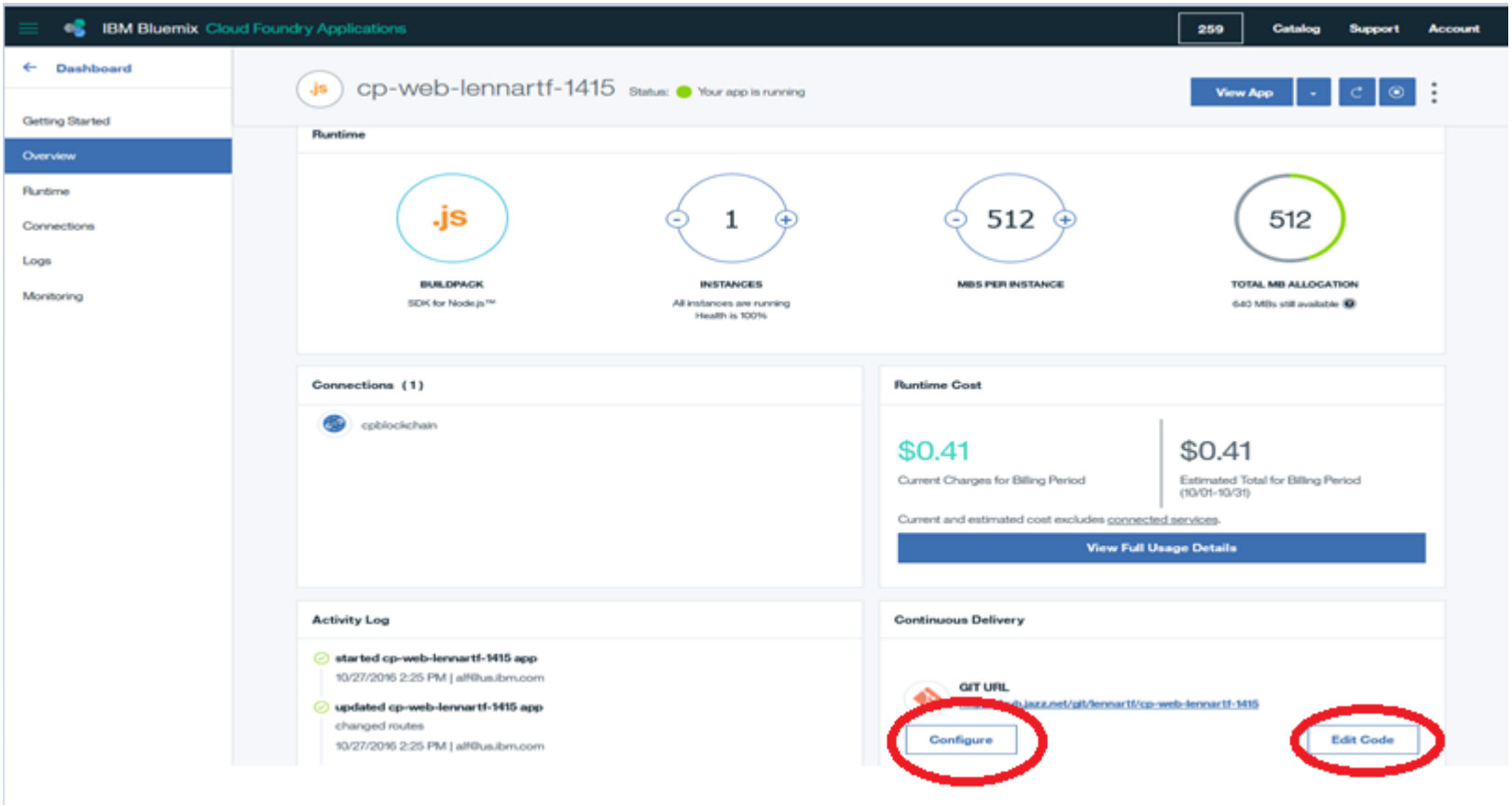
# Lab 2 Commercial Paper – View Your App – Trade Center



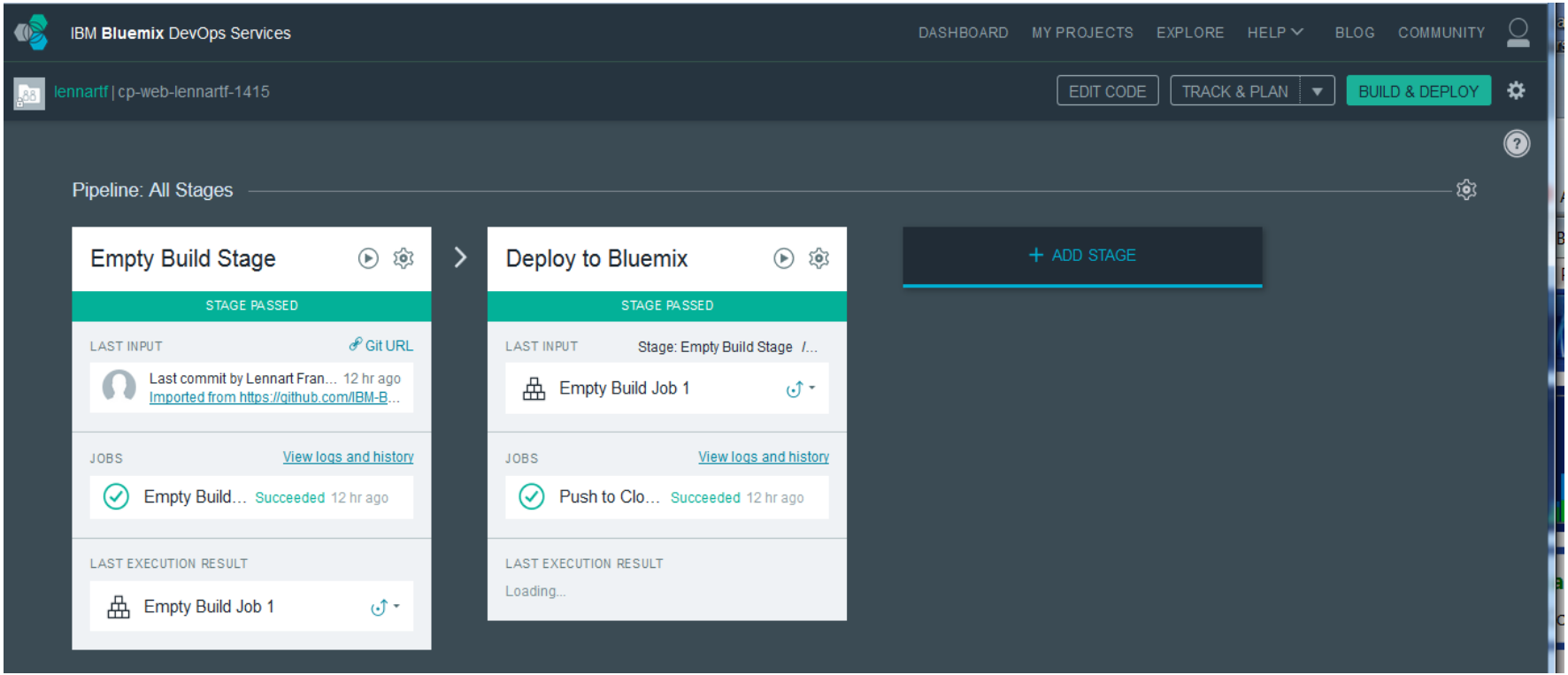
# Lab 2 Commercial Paper in Bluemix Dashboard

| | | | | | | |
|---|---|---|---|---|---|---|
| bluemixintro1 | bluemixintro1.mybluemix.net | 128 | 1 | 1 | Running | |
| bluemixintroiot | bluemixintroiot.mybluemix.net | 512 | 1 | 1 | Running | |
| cp-web-lennartf... | cp-web-lennartf-1415.mybluemix.... | 512 | 1 | 1 | Running | |
| falkenberg | falkenberg.mybluemix.net | 128 | 1 | 0 | Stopped | |
| IndiaHackathon... | IndiaHackathonMFSS.mybluemix... | 512 | 1 | 1 | Running | |
| manifesttest.yml | | 128 | 1 | 1 | Running | |

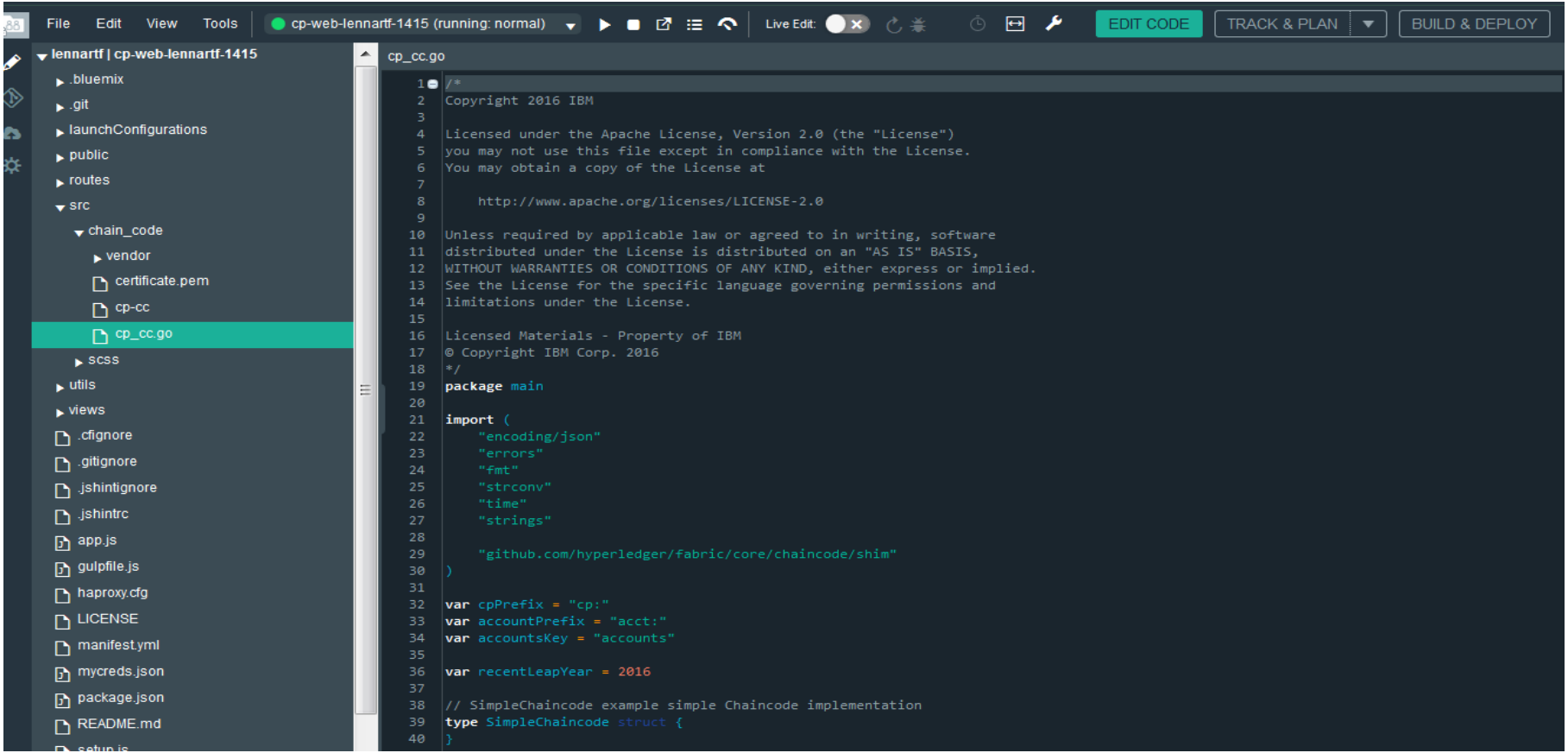
# Lab 2 Commercial Paper in Bluemix Dashboard



# Lab 2 Commercial Paper – DevOps Services – Configure - Build and Deploy



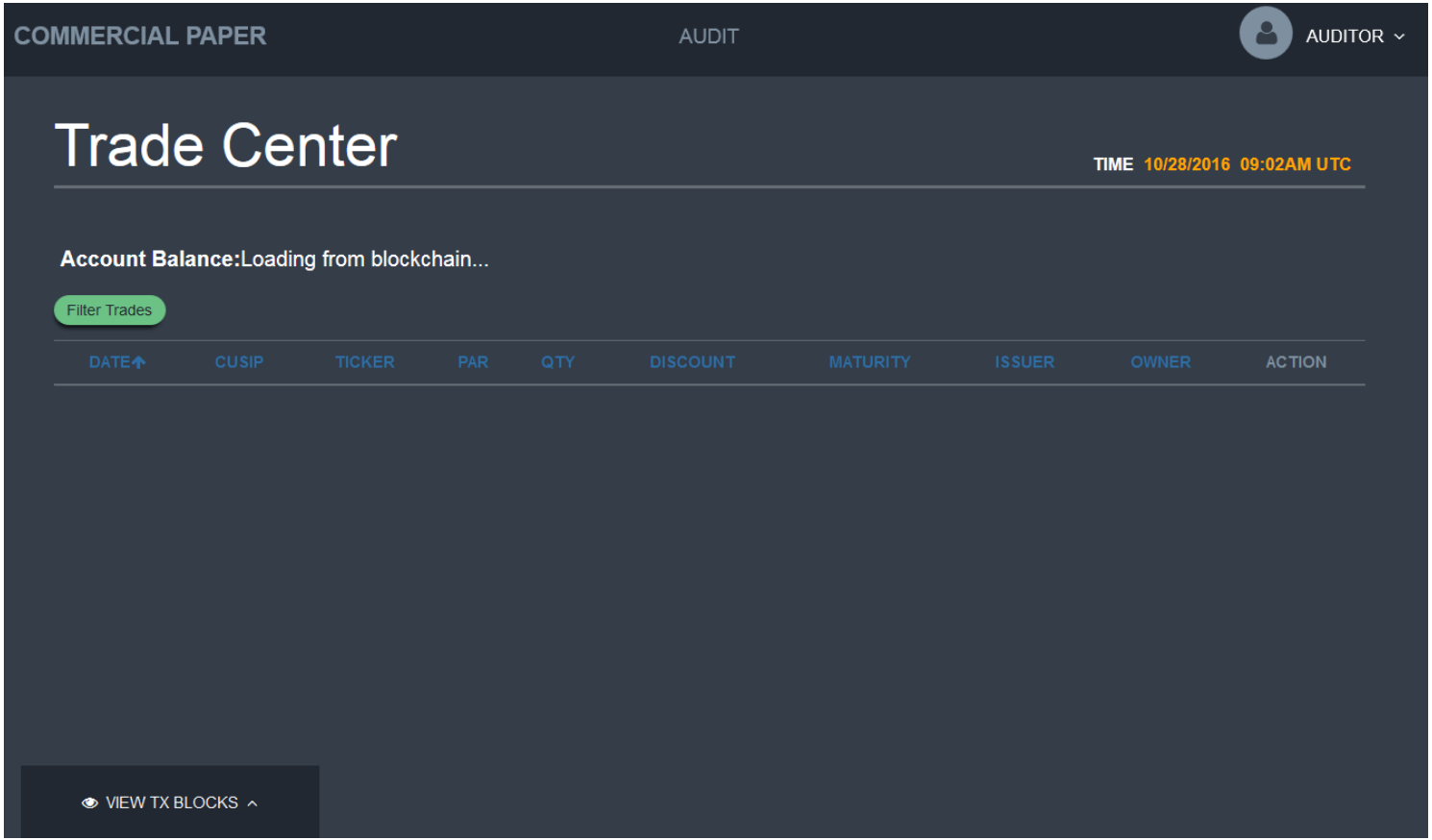
# Lab 2 Commercial Paper – Devops Services – Edit Code – Build and Deploy



# Lab 2 Commercial Paper in Bluemix Dashboard – View App

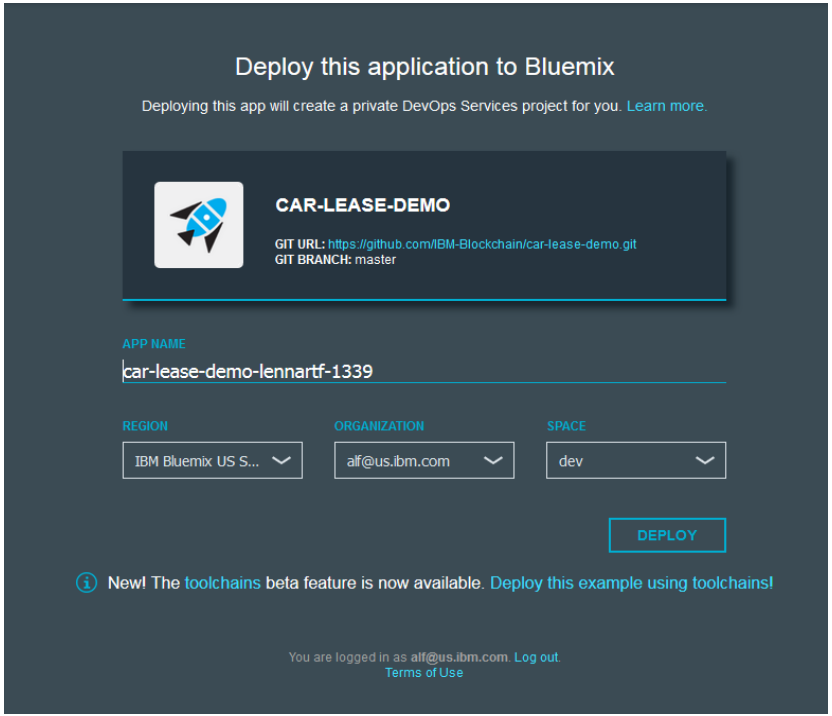
| | | | | | | |
|---|---|---|---|---|---|---|
| bluemixintro1 | bluemixintro1.mybluemix.net | 128 | 1 | 1 | Running | |
| bluemixintroiot | bluemixintroiot.mybluemix.net | 512 | 1 | 1 | Running | |
| cp-web-lennartf... | cp-web-lennartf-1415.mybluemix.... | 512 | 1 | 1 | Running | |
| falkenberg | falkenberg.mybluemix.net | 128 | 1 | 0 | Stopped | |
| IndiaHackathon... | IndiaHackathonMFSS.mybluemix... | 512 | 1 | 1 | Running | |
| manifesttest.yml | | 128 | 1 | 1 | Running | |

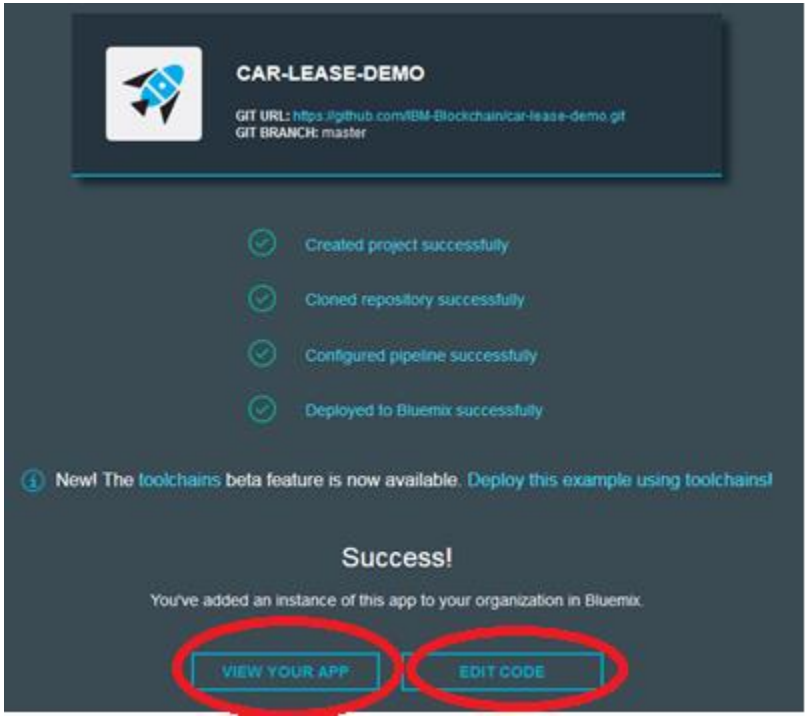
# Lab 2 Commercial Paper – View Your App – Trade Center



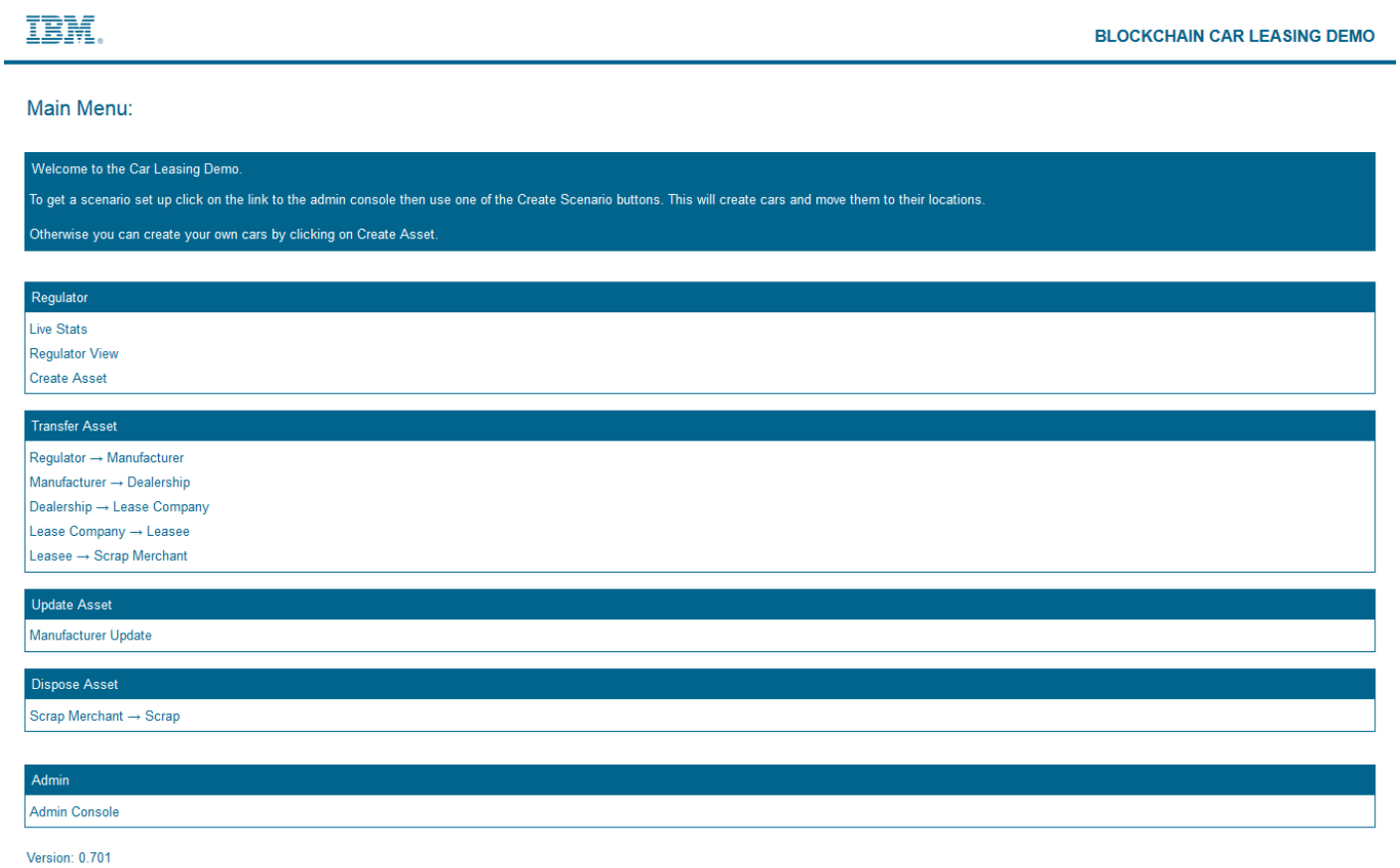
# Lab 3 Car Lease

https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_tutorials.html

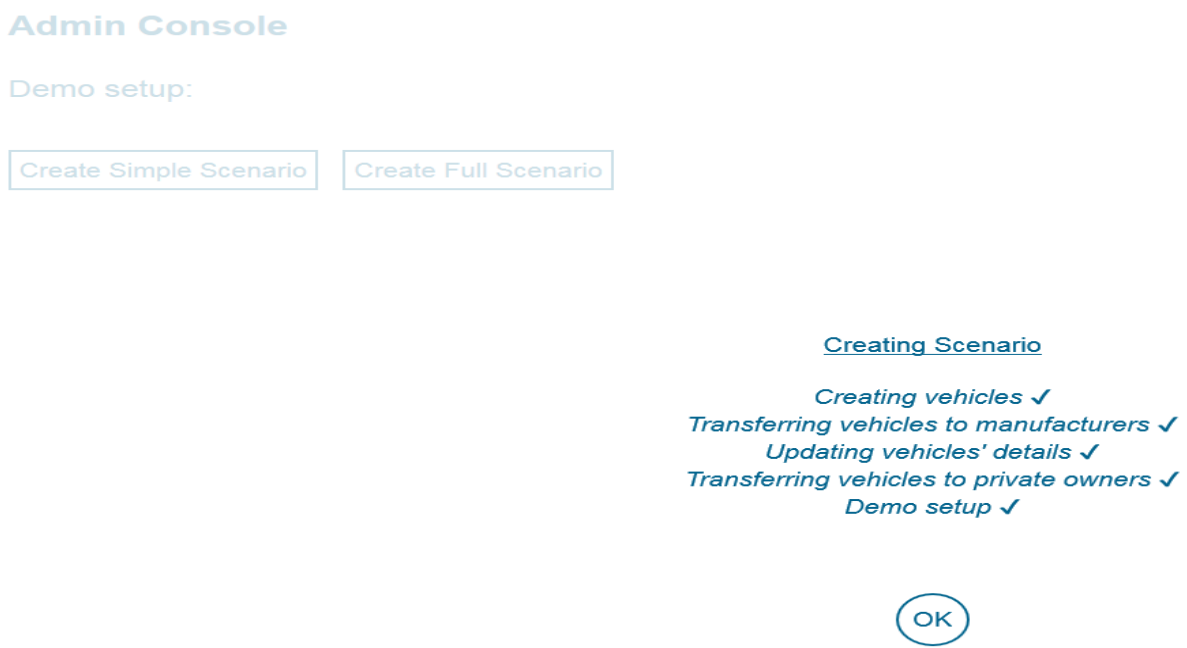


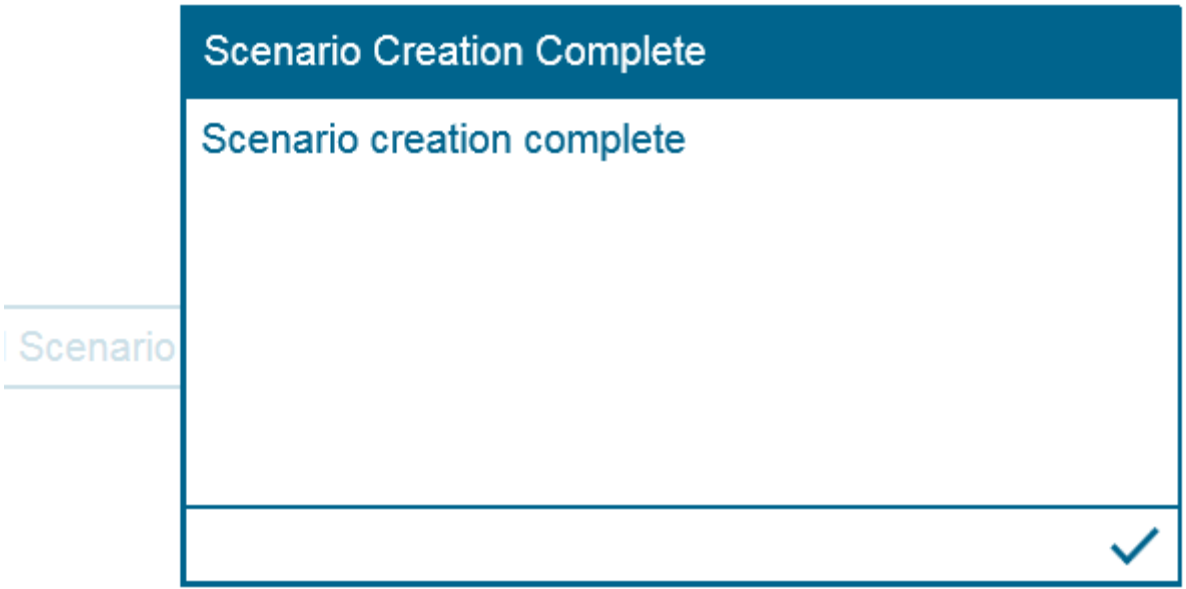
# Lab 3 Car Lease - View App



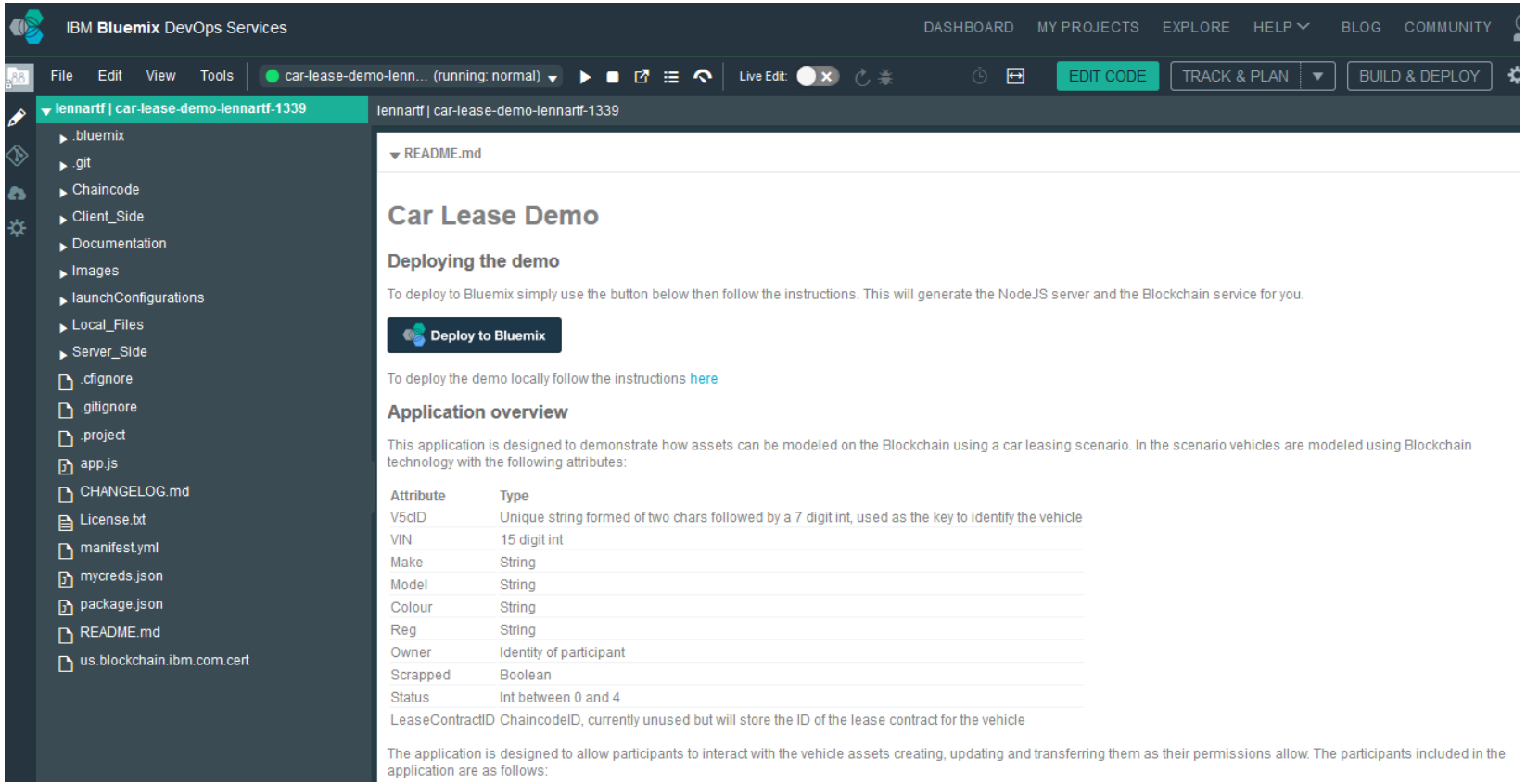
# Lab 3 Car Lease - View App



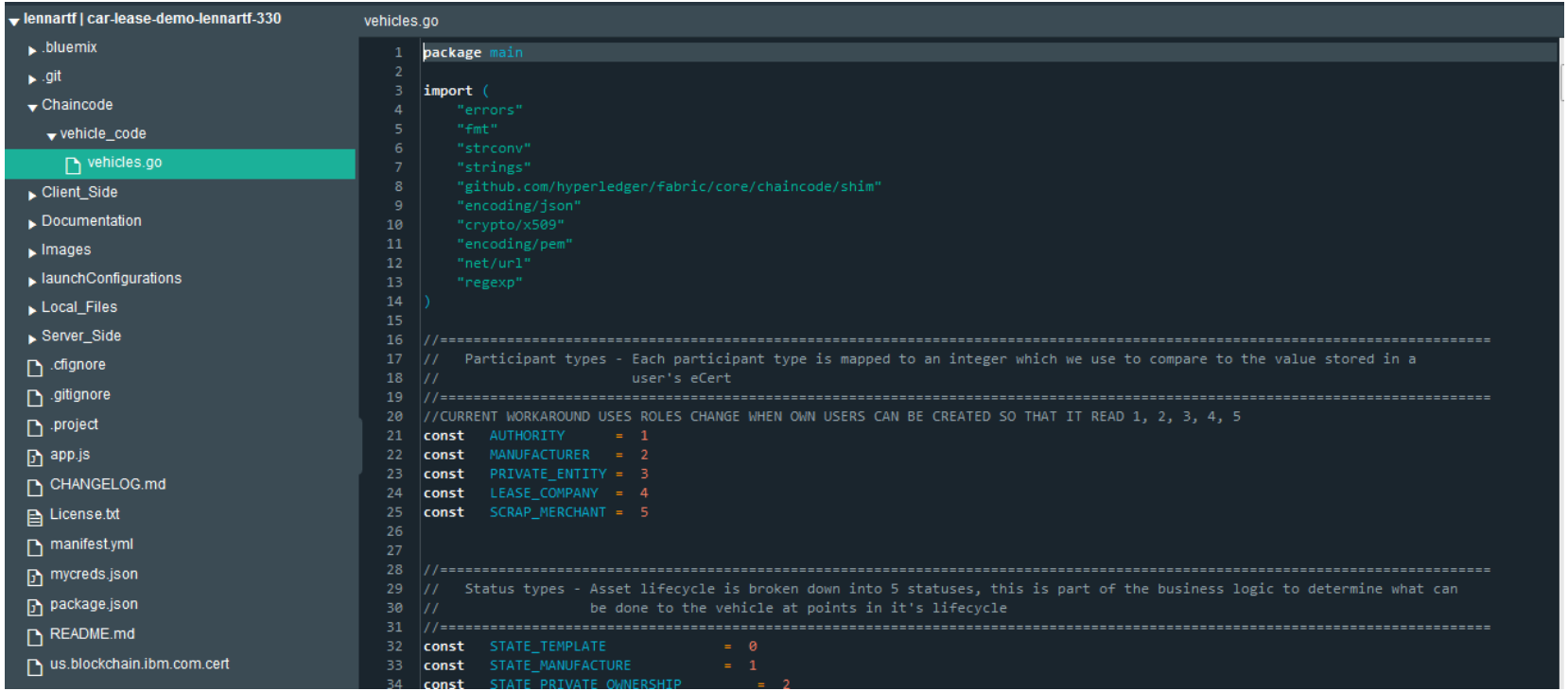
# Lab 3 Car Lease - View App



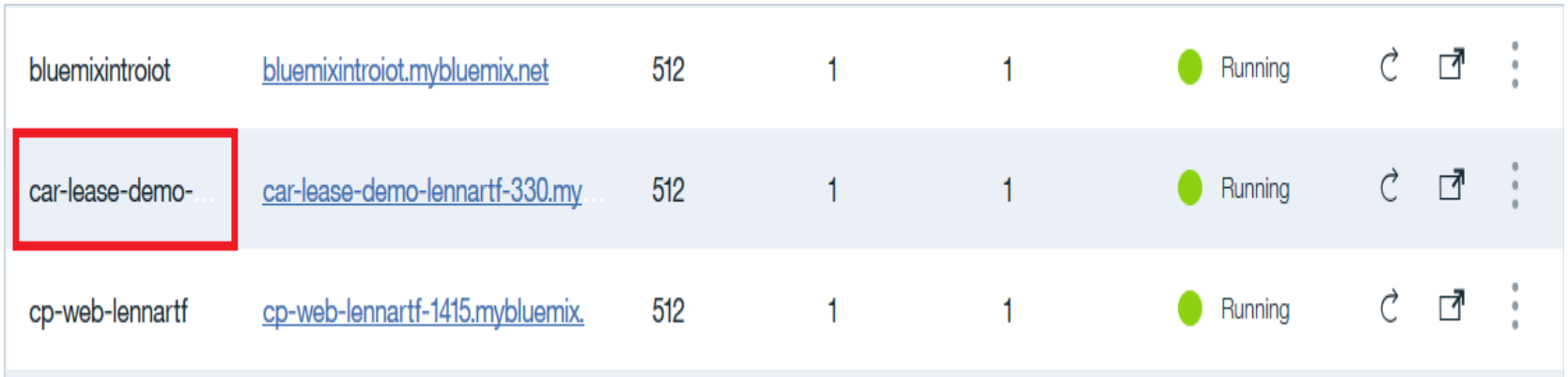
# Lab 3 Car Lease - Edit Code – Build and Deploy



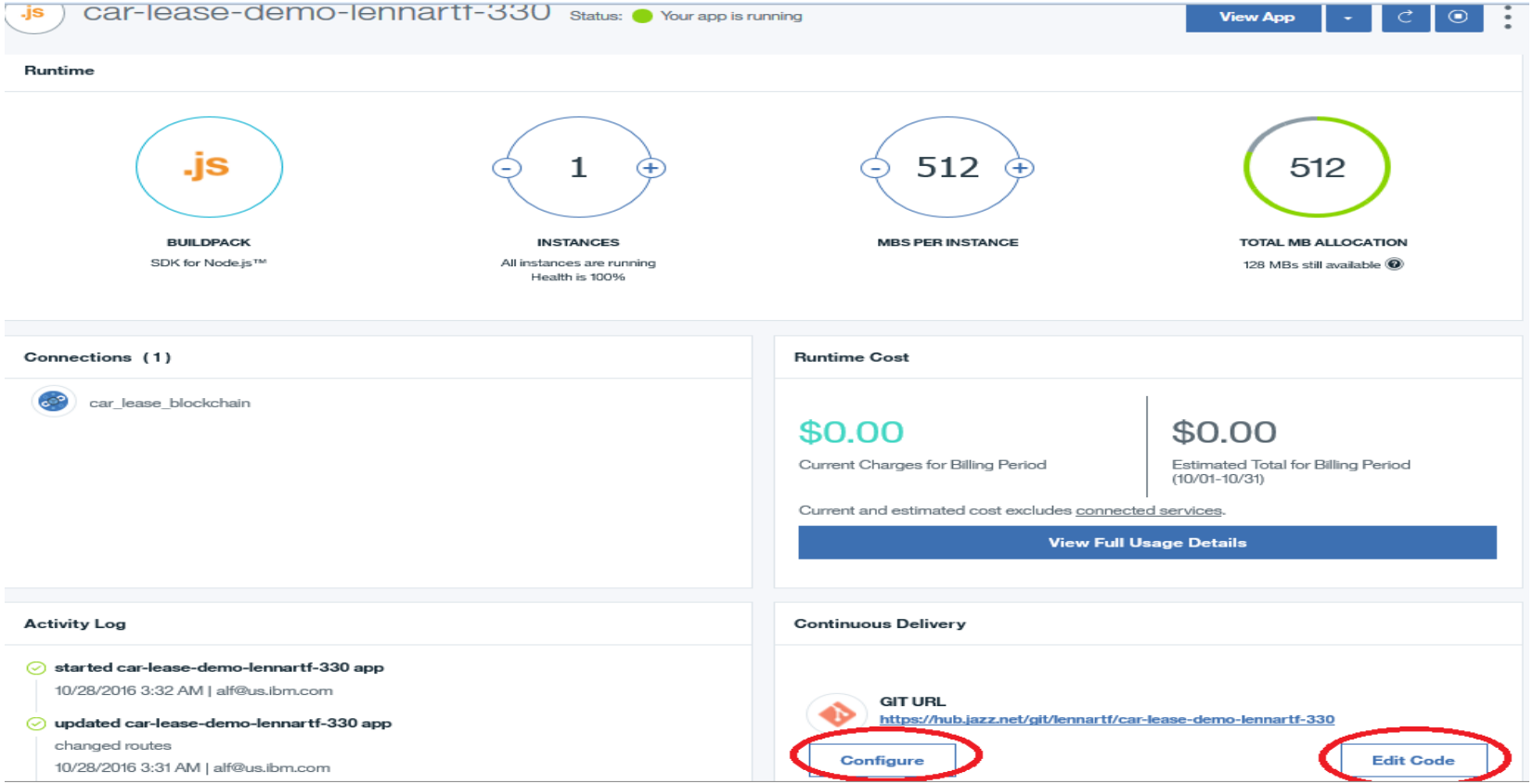
# Lab 3 Car Lease - Edit Code – Build and Deploy – Chain Code



# Bluemix Console



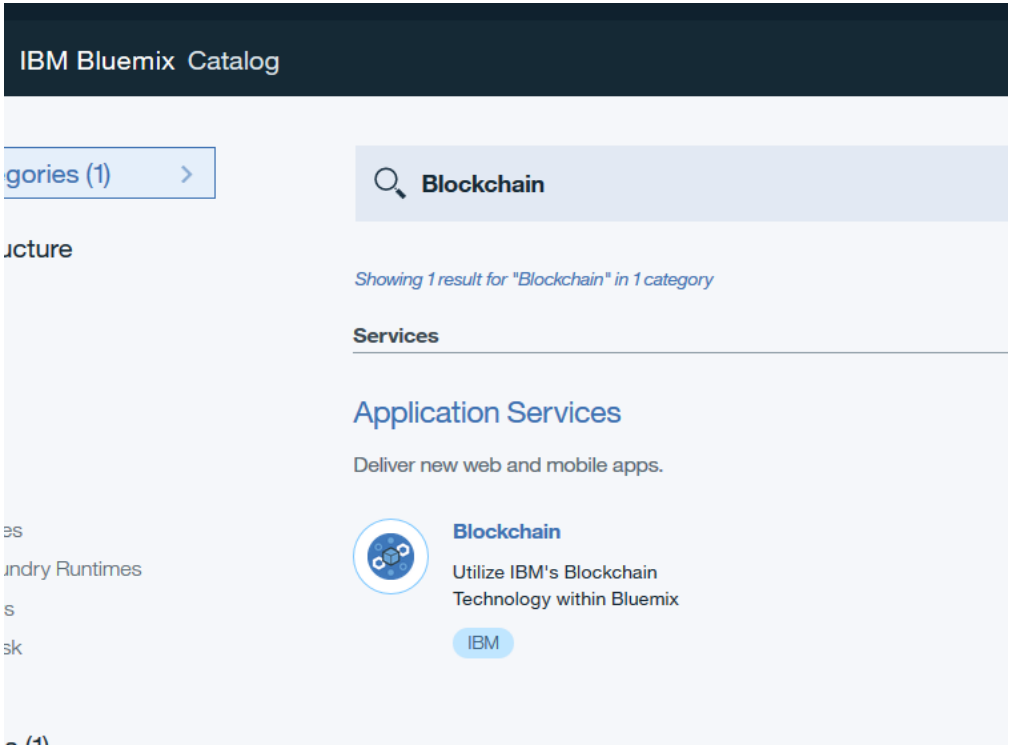
# Bluemix Catalog



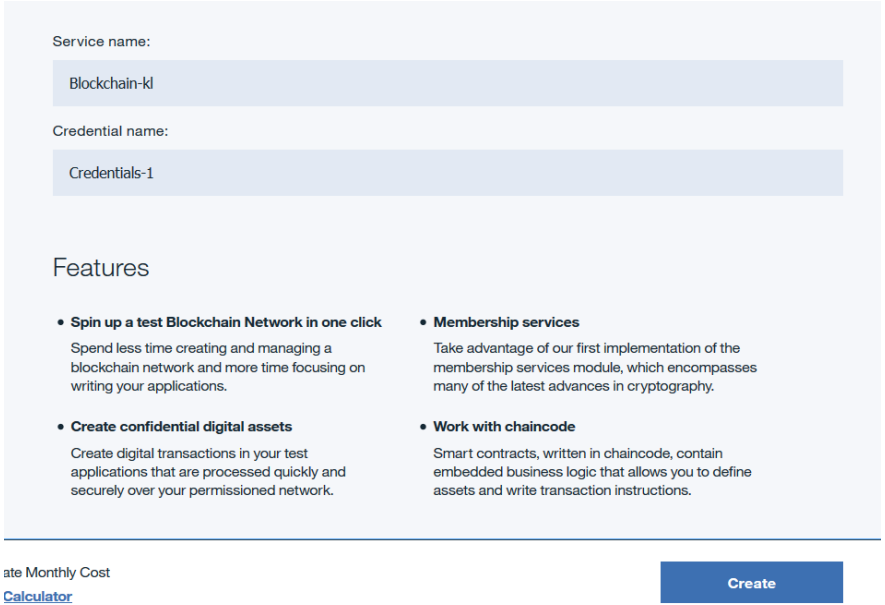
# Bluemix Catalog

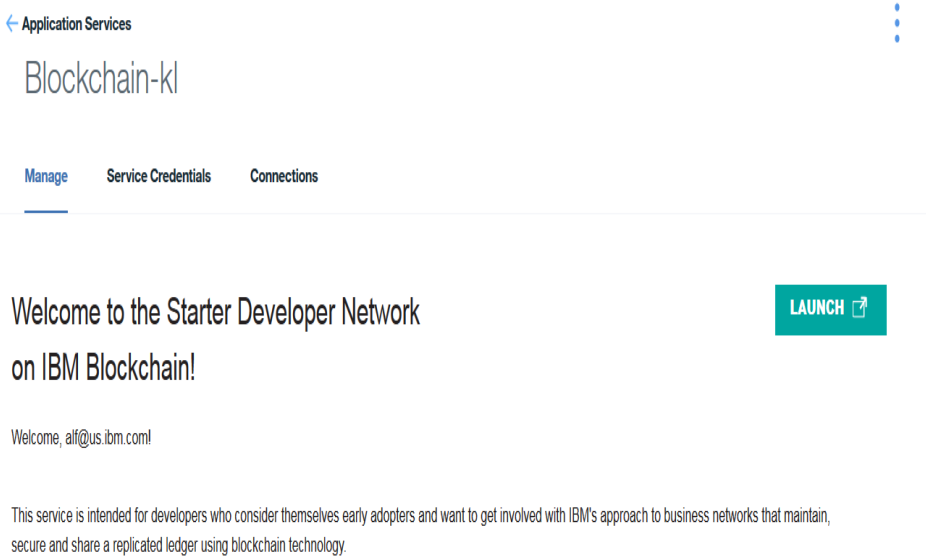
| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| bluemixintroiot | bluemixintroiot.mybluemix.net | 512 | 1 | 1 | Running | | | |
| car-lease-demo-... | car-lease-demo-lennartf-330.my... | 512 | 1 | 1 | Running | | | |
| cp-web-lennartf | cp-web-lennartf-1415.mybluemix. | 512 | 1 | 1 | Running | | | |

# Bluemix Catalog

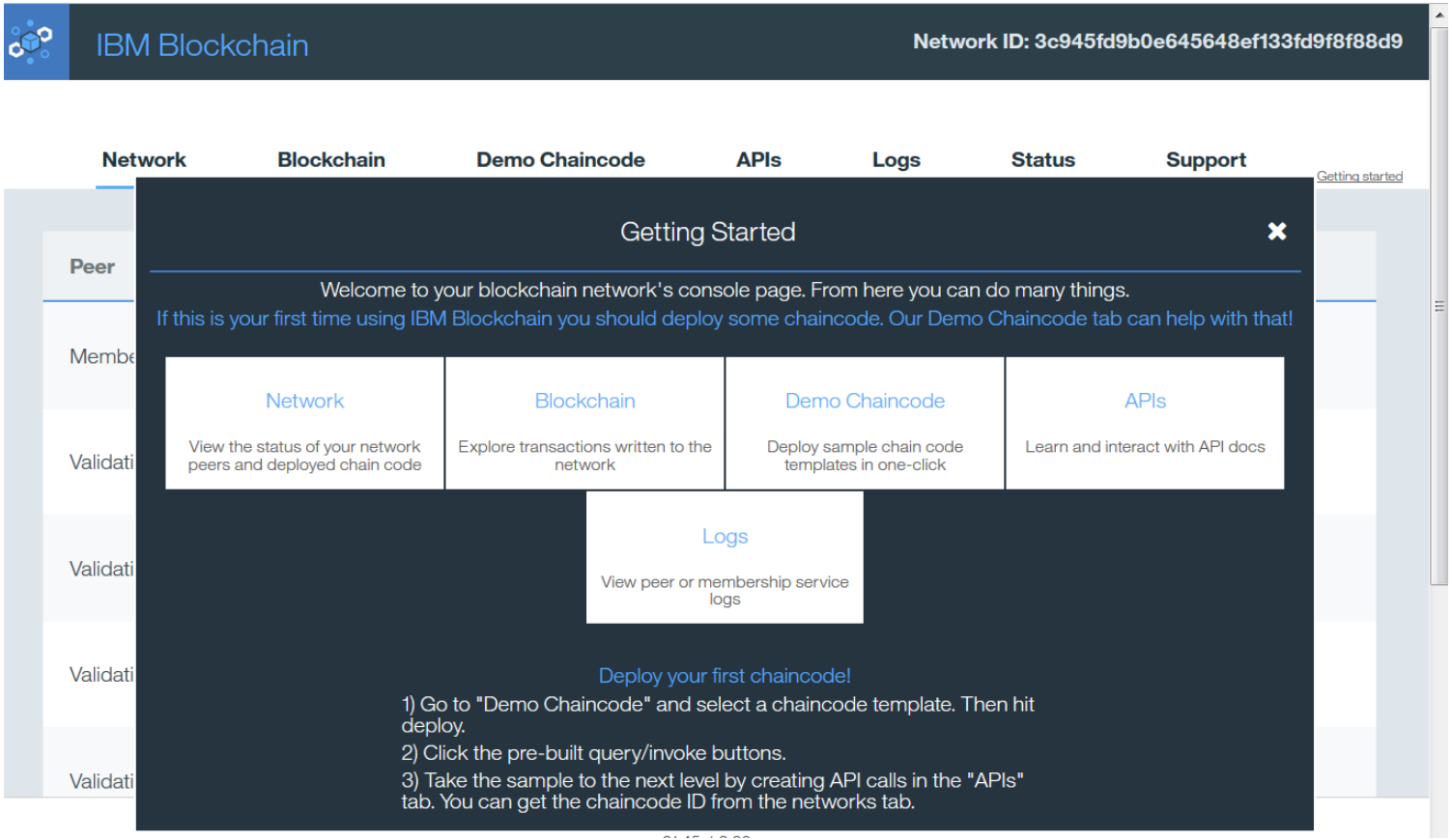


# Bluemix Catalog



# Bluemix Catalog



# Blockchain, Demo Chaincode

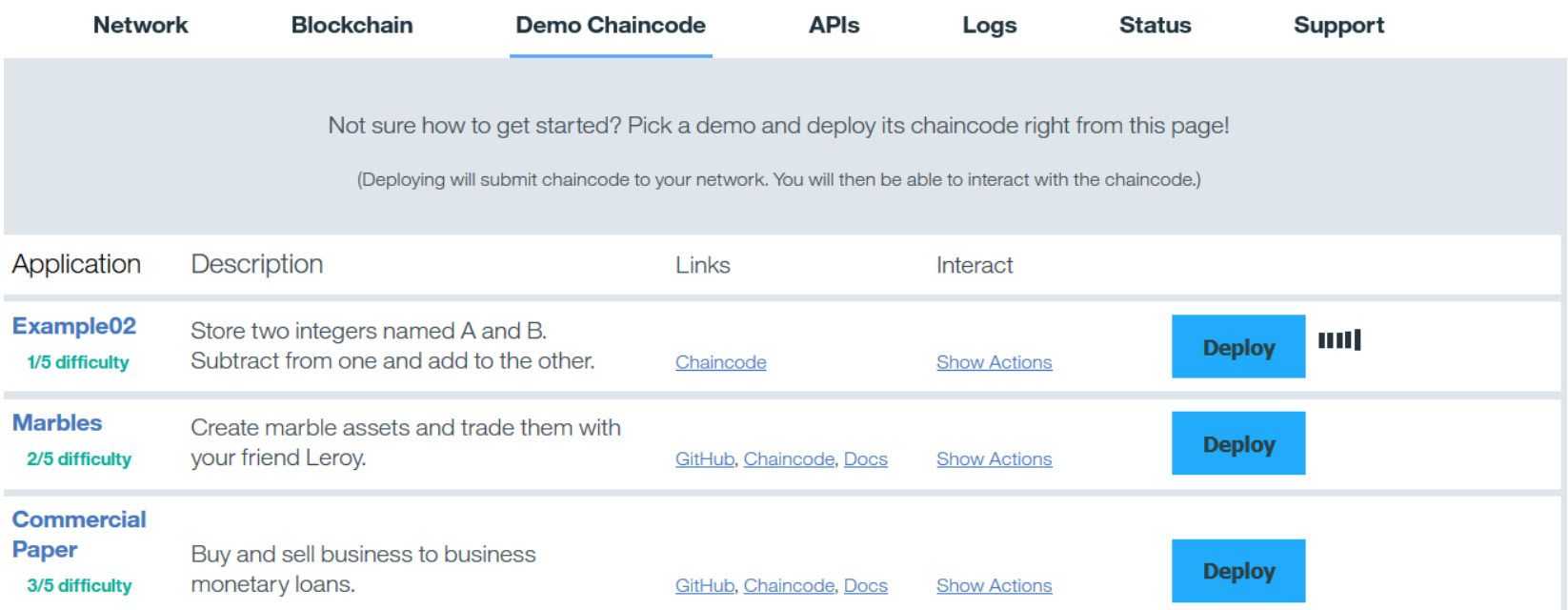
Network | Blockchain | Demo Chaincode | APIs | Logs | Status | Support

Not sure how to get started? Pick a demo and deploy its chaincode right from this page!

(Deploying will submit chaincode to your network. You will then be able to interact with the chaincode.)

| Application | Description | Links | Interact | |
|---|---|---|---|---|
| **Example02** 1/5 difficulty | Store two integers named A and B. Subtract from one and add to the other. | Chaincode | Show Actions | Deploy |
| **Marbles** 2/5 difficulty | Create marble assets and trade them with your friend Leroy. | GitHub, Chaincode, Docs | Show Actions | Deploy |
| **Commercial Paper** 3/5 difficulty | Buy and sell business to business monetary loans. | GitHub, Chaincode, Docs | Show Actions | Deploy |

# Bluemix chaincode samples



# Bluemix Example02 sample

| Application | Description | Links | Interact | |
|---|---|---|---|---|
| **Example02**<br>1/5 difficulty | Store two integers named A and B.<br>Subtract from one and add to the other. | Chaincode | Show Actions | Deploy |

Select the correct chaincode `example02: 72c3c395...`

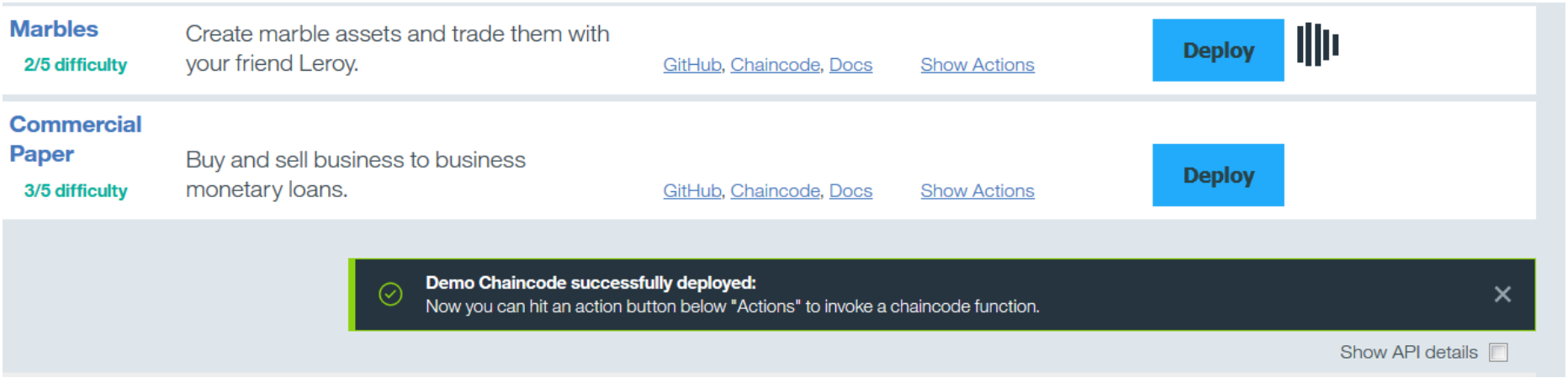
Transfer from A to B

Transfer from B to A

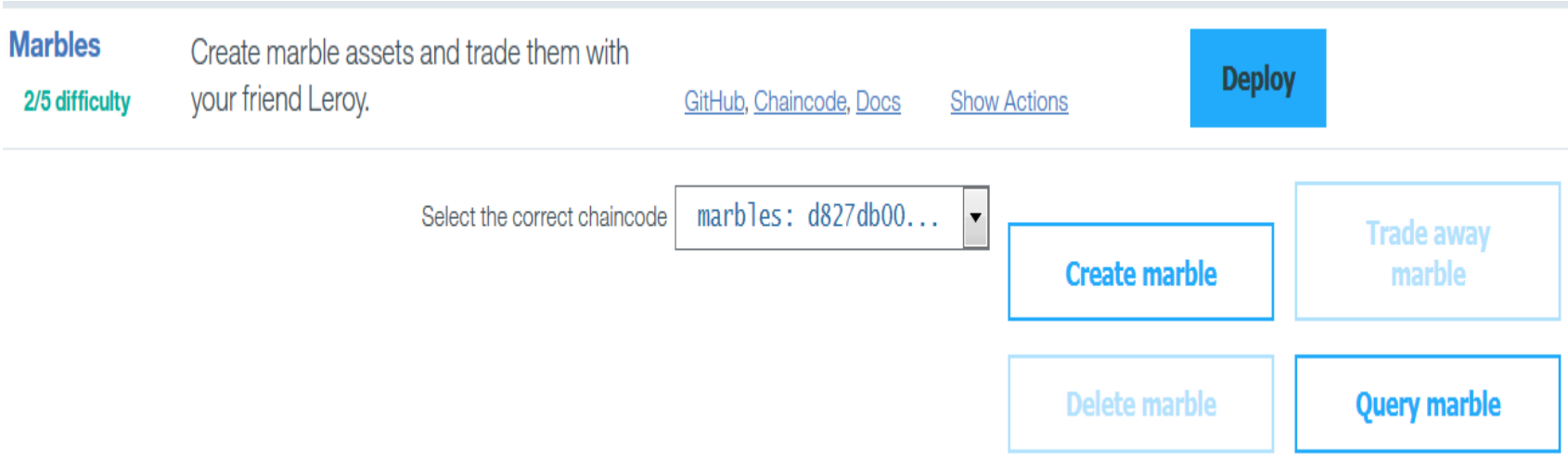
Query A

Query B

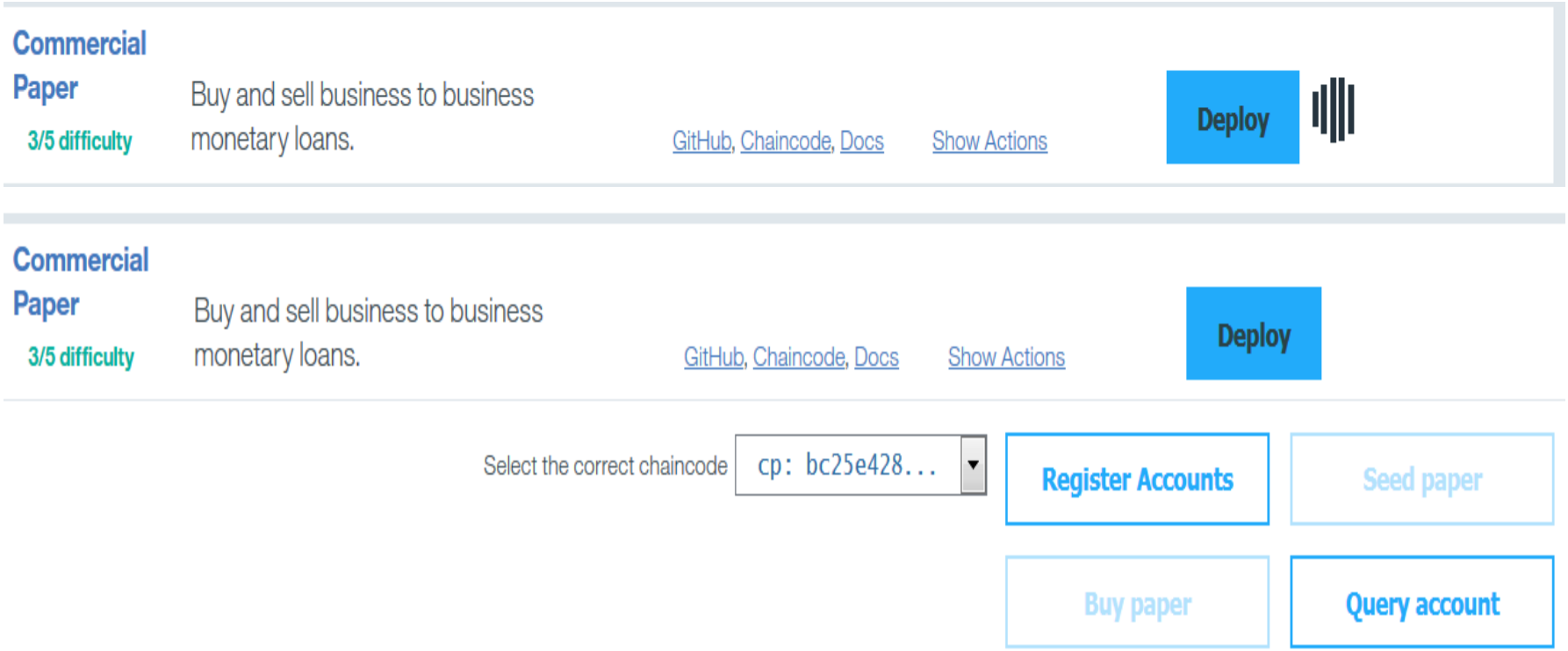
# Chaincode: Marbles



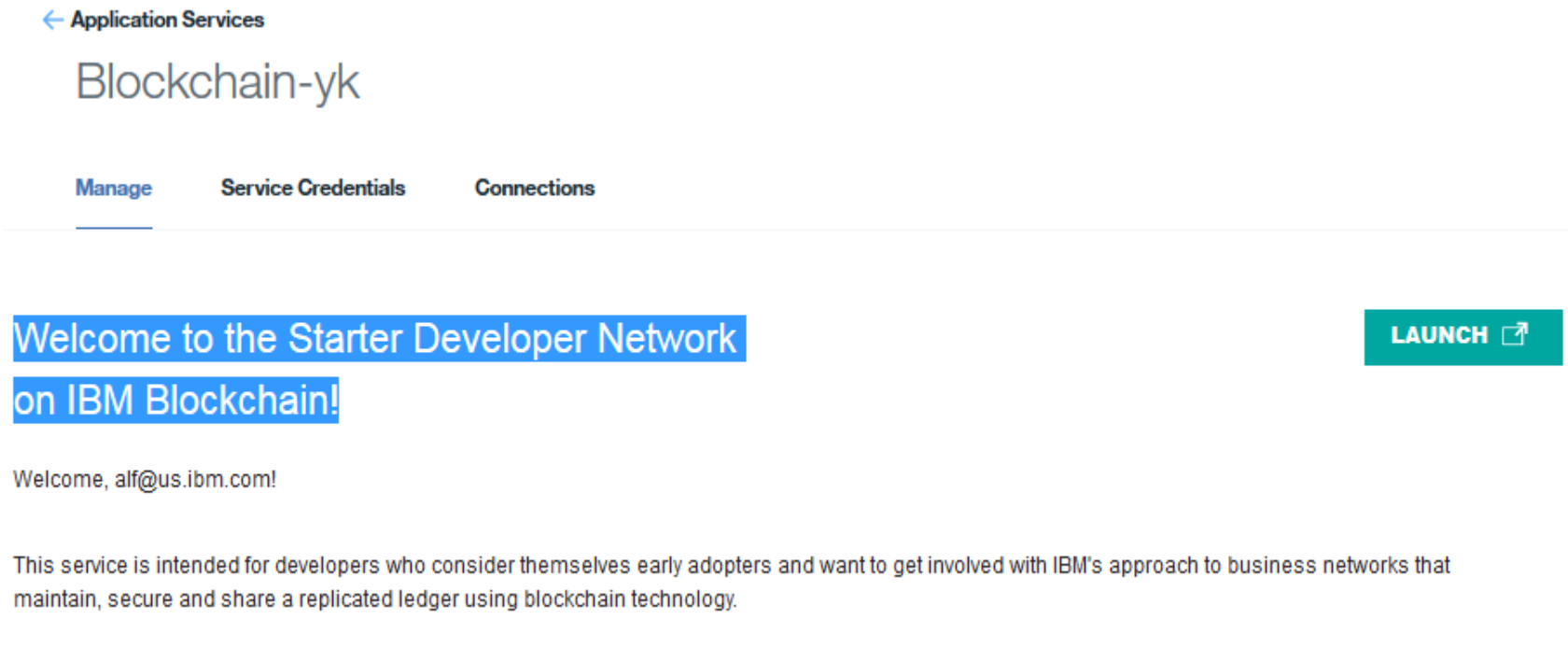
# Blockchain Marbles



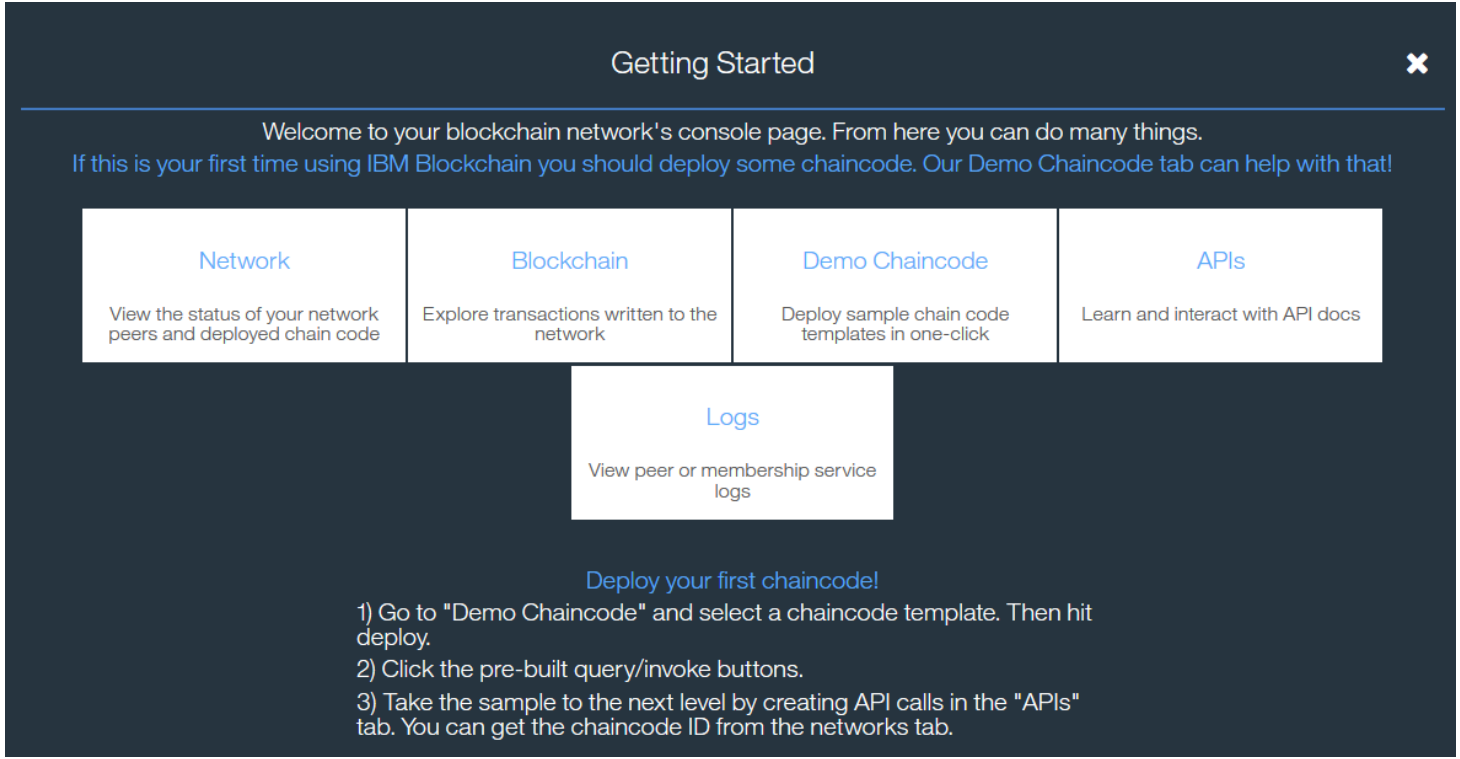
# Bluemix Commercial Paper



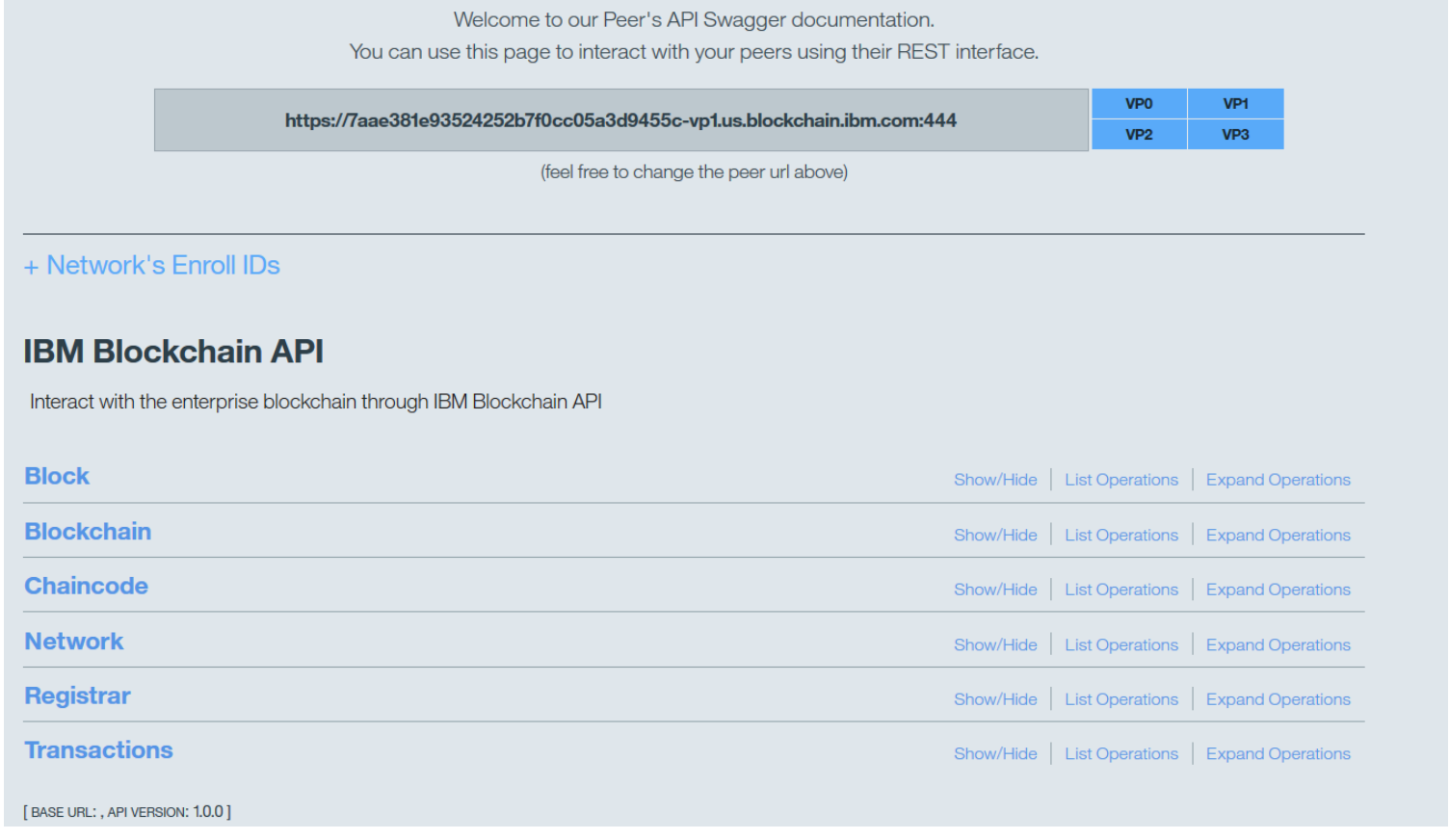
# Welcome to the Starter Developer Network on IBM Blockchain!



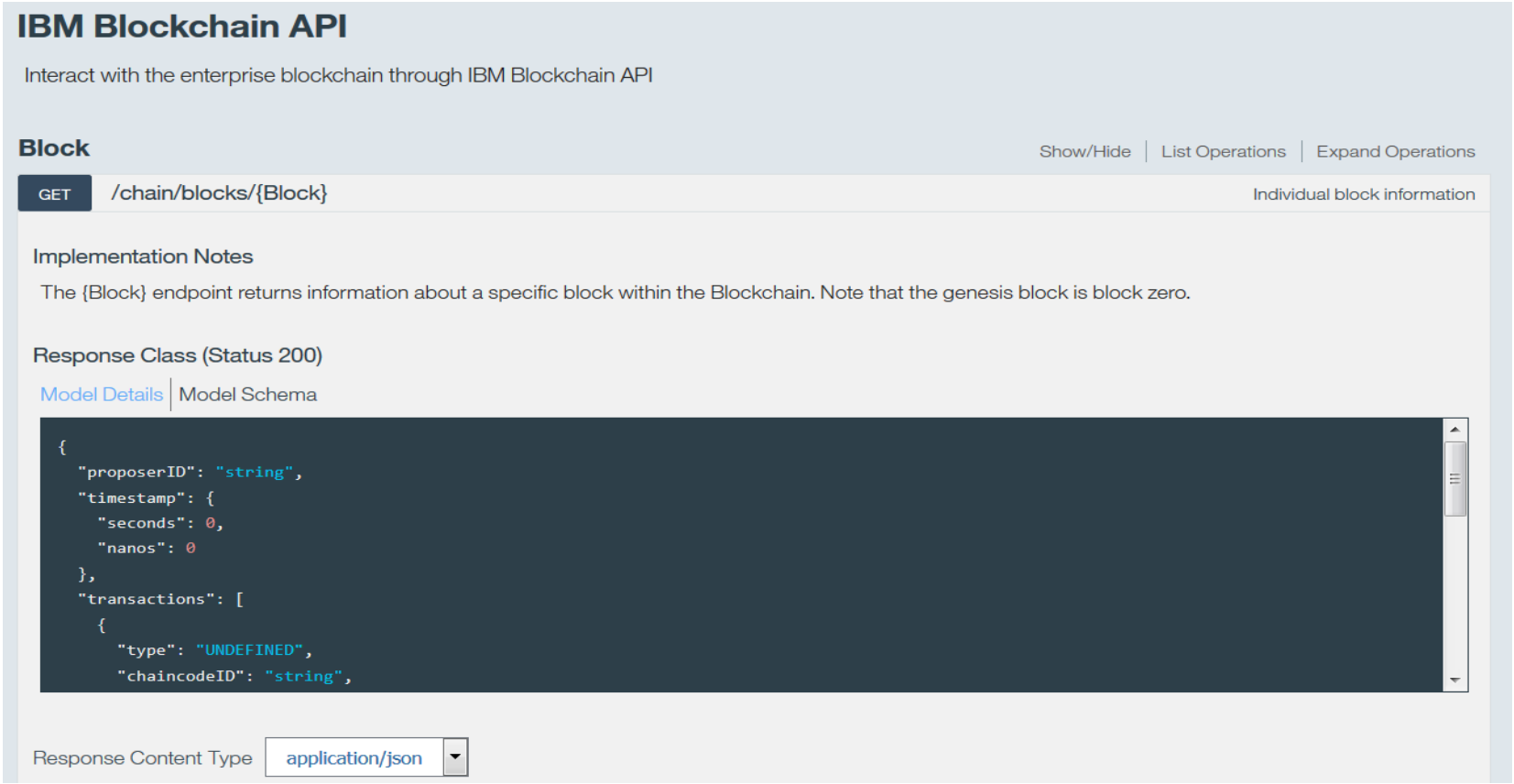
# Welcome to the Starter Developer Network on IBM Blockchain!



# Welcome to the Starter Developer Network on IBM Blockchain!



# Welcome to the Starter Developer Network on IBM Blockchain!



# Section 4
# Where do we go from here

# Using the Hello Chaincode tutorial

This tutorial guides you through using basic building blocks to code an elementary cha

- 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?

https://console.ng.bluemix.net/docs/services/blockchain/ibmblockchain_tutorials.html#hellocc

# Learn Chaincode

A tutorial to get you started with writing smart contracts for Hyperledger.

## Deployment

In order to support multiple versions of the Hyperledger fabric, this repository uses branches in combination with gopkg.in URLs. What does this mean for beginners? Just pick the branch below and use the instructions for that branch to complete the tutorial

https://github.com/IBM-Blockchain/learn-chaincode/blob/master/README.md

# Educate Yourself!

- Everything is available through the web:
  - http://hyperledger.org
  - https://github.com/hyperledger
  - http://ibm.com/blockchain