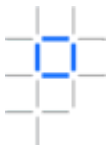**IBM Blockchain Platform**

← **A9: Publishing an event**

Tutorial A10: Claim your badge!

Estimated time: `60 minutes`

If you've completed all the tutorials up to this point, congratulations!

The purpose of this final tutorial is to:

- Summarize what you have learned so far
- Identify some additional resources that you might find interesting and helpful
- Invite you to test your learning and gain a badge

This tutorial will reference the concepts we learned in the previous tutorials in this set. If you were already familiar with Hyperledger Fabric and the IBM Blockchain Platform and jumped straight to this page to claim your badge, feel free to skip ahead to the badge section.

▢ `A10.1`:   Expand the first section below to get started.

▶ **Tutorial Set 'A' recap**

This set of tutorials introduced you to Hyperledger Fabric. You've learned all the the basic concepts by developing a smart contract and a set of applications that interact with a blockchain network.

Hyperledger Fabric allows you to create networks of *immutable, distributed blockchain ledgers* containing shared transactions. Hyperledger Fabric uses cryptographic mechanisms to create multi-party transactions that are provably valid by all organizations in the network. Such ledgers help to create trust amongst those who host them; trust which can be used to improve costly processes that require a verifiable view of shared transactions, such as dispute resolution or supply chain management.

Blockchains are used to share transaction data that is related to any business object of value. In these tutorials we created transactions for art objects, with simple key/value pairs for these objects' properties. However, the principles apply equally well to any business object of value to a network of organizations.

A solution involving a Hyperledger Fabric blockchain consists of three main components:

- A **network** of collaborating organizations who each host one of more copies of a distributed ledger of transactions on a set of peer nodes.
- A **smart contract** that defines, in code, the business rules used to create transactions relating to a business object. The transactions are cryptographically recorded on the ledger.
- A **client application** that can interact with the network and smart contracts:
  - Submit new multi-party transactions to the distributed ledger.

- Evaluate smart contracts to query the ledger
- Register for notification of custom events generated by smart contracts

The development process for smart contracts and client applications is similar to the development process for other software applications. Code is developed, built, tested, debugged and deployed. In these tutorials, we used the tools provided by the IBM Blockchain Platform VS Code extension to go through the main tasks in this process using a locally installed Hyperledger Fabric network.

🔲 `A10.2`: Expand the next section to continue.

▶ 🏆 **IBM Blockchain Foundation Developer badge**

By completing these tutorials you should have gained the knowledge required to pass the 🏆IBM Blockchain Foundation Developer course.

To gain this badge, you will need to pass a short test to demonstrate your understanding. Start the course and skip directly to the *Test your knowledge* section. Once you pass, you can add the badge to your Acclaim profile.

🔲 `A10.3`: Complete the course, then expand the next section to continue.

▶ **What's next?**

Using a local network is useful to understand concepts and start developing smart contracts and applications. However, our sample network was totally within our control; it allowed us to start our journey, but this only a small part of the blockchain story.

So far we have been submitting transactions that update ledgers hosted on our local machine. As this transaction data eventually maps to bytes on our disk, it is still fundamentally modifiable; it doesn't give us proof that data has not been tampered with.

What sets blockchain apart from a database - even a shared database - is the power of the network to enable tamper resistance. By distributing multiple copies of the data across a network and having different nodes agree on the contents through a process of *consensus*, it makes it unfeasible, even for administrators, to modify the data.

In the next set of tutorials ('B'), we will connect to an existing network and see the power of the network in action. We will see how another tool that is part of IBM Blockchain Platform - the web console - can be used to join a network. We'll then connect both VS Code and client applications to that network and create transactions that are agreed by multiple organizations in the network.

> **Tutorial set B: Joining a network**
>
> The next set of tutorials is coming soon. In the mean time, here are some links to additional resources you can use to continue your learning.
>
> - Getting Started with Enterprise Blockchain
> - Hyperledger Project
> - Hyperledger Fabric Documentation
> - IBM Blockchain Homepage

- IBM Blockchain Developer site
- Introduction to the IBM Blockchain Platform web console
- IBM Blockchain Blog

Congratulations on completing this tutorial set, and have fun!