Exercise 1: Setup Azure Active Directory Tenant
In this exercise, the student will configure a new Azure AD Tenant that the Azure Blockchain Workbench will use to authenticate users.

SOLUTION:
Is Azure really needed? We already have another directory service solution in place. We can't change it simply because we want to leverage Blockchain!
The PoC (Proof of Concept) level of using Azure Blockchain Workbench integrates with Azure AD as the preferred choice of handling identity and authentication. However, if you do need another solution to be implemented, the architecture can be integrated with other existing ID management solutions when necessary.

Exercise 2: Deploy Azure Blockchain Workbench
In this exercise, the student will deploy and setup Azure Blockchain App Builder.

SOLUTION:
With Azure Blockchain Workbench, Microsoft has already done all the heavy lifting. Not only are all the relevant Azure services deployed to your Azure subscription with an ARM Template, but the are automatically stitched together and offer a simple to use presentation layer. This allows customers to model their workflows and quickly demonstrate value to both business and technical decision makers. And, the Blockchain Workbench only takes about 1 hour to initially deploy.

Exercise 3: Check Blockchain Workbench Web Client Deployment
In this exercise, the student will access the Azure Blockchain Workbench Web Client to make sure the deployment is working as expected. The student will also check the Ethereum network status.

Exercise 4: Create Smart Contract
In this exercise, you will create a new Smart Contract that targets the Ethereum blockchain that is written in the Solidity programming language.

SOLUTION:
When adopting Blockchain Workbench, the Smart Contracts are written using a language called Solidity. The Solidity language syntax was developed to be similar to C++ and JavaScript language syntax. This helps keep the Solidity language syntax familiar to most existing developers who are already familiar with these other C-style syntax languages.

Exercise 5: Assign Users to Contract Personas
In this exercise, the student will create some additional Users with Azure AD that correspond to the different user roles with the Blockchain solution. These users will also be assigned to the different personas within the Azure Blockchain Workbench their roles within the workflow coincide with.

SOLUTION:
How are user accounts or Blockchain wallets managed within Azure Blockchain Workbench? We really need to make sure we have complete control over the security of the entire system.
One of the components that are setup with Azure Blockchain Workbench at the time of provisioning is the integration of Azure Active Directory (Azure AD). This enables the system to utilize and rely on Azure AD for the management of all the users and their passwords for authenticating and accessing

the Azure Blockchain Workbench presentation layers.

Exercise 6: Create and Process an Instance of the Smart Contract
In this exercise, the student will create a new instance of the Telemetry Compliance Smart Contract, and work through the workflow of the Supply Chain as programmed in the code of the Smart Contract. The student will also test the Compliance piece by simulating IoT Device Telemetry by logging in as the “Simulated Device” user.