

## Reading Sample

In this sample chapter, you'll learn about the features of the Open Connectors capability in SAP Integration Suite. The chapter describes how to work with Open Connectors and integrate them with non-SAP applications. First, you'll review the basics of Open Connectors, then you'll learn how to create and use Open Connectors instances. Finally, you'll see how Open Connectors can be used with API Management.



**"Open Connectors"**



**Contents**



**Index**



**The Authors**

Jaspreet Bagga

### SAP Integration Suite Certification Guide

Development Associate Exam

417 pages | 08/2023 | \$79.95 | ISBN 978-1-4932-2455-5



[www.sap-press.com/5735](https://www.sap-press.com/5735)

# Chapter 4

## Open Connectors

# 4

### Techniques You'll Master

- Understanding the basics of Open Connectors
- Creating and using Open Connectors instances
- Using Open Connectors with API Management

A connection to a particular application programming interface (API) provider endpoint is made possible through connectors, which are prebuilt API integrations, for example, for Salesforce, QuickBooks, or Marketo. Authentication, resources, pagination, failures, events, and search are among the standard set of functionalities that all connections have at the outset. Even though not all of the connections in a category share a resource, SAP enables a more extensive set of APIs that an application offers at the hub level.

In contrast to many other customer relationship management (CRM) applications, Salesforce Sales Cloud supports APIs. These APIs, which are only applicable to Salesforce, are listed in Salesforce's documentation:

- Connectors share a variety of common features, such as interactive documentation, bulk downloading and uploading, bulk uploading and searching, pagination, and bulk uploading and downloading.
- RESTful APIs provide access to normalized methods.
- Full data payloads are provided in JavaScript Object Notation (JSON) and can be normalized and transformed into a single resource.
- Each connector is kept current with modifications at the endpoint by SAP.
- Each connector supports an unlimited number of authorized accounts without the need for additional code because it is a multi-tenant connector.

With the help of Open Connectors, your organization can link and integrate its applications with a variety of external services, such as Salesforce and other CRM platforms, payment gateways, social media sites, and more. The platform offers a collection of prebuilt connectors that are simple to configure and use to connect with a range of applications and services.

In this chapter, we'll explore various topics related to Open Connectors. Our journey begins with a clear understanding of the objectives specific to this portion of the test. We'll then delve into the comprehensive features that Open Connectors offer, examining how they can enhance our connectivity and integration capabilities.

For a practical understanding, we'll dive deeply into the workings of Open Connectors. You'll learn how to log on to the Open Connectors platform and create a Twitter instance within it. Additionally, you'll discover how to establish an API provider using Open Connectors and create a fully functional API from the provider. Along the way, you'll gain insights into incorporating policies into the API and bundling a product to further enhance its functionality.

Expanding our knowledge, we'll explore the process of creating applications and testing the APIs we develop. Through hands-on activities, you can solidify your understanding and familiarize yourself with the practical implementation of Open Connectors.

Throughout this chapter, you'll encounter important terminology relevant to Open Connectors, empowering you to communicate effectively and comprehend the intricacies of the platform. To reinforce your learning, practice questions will challenge your understanding. Detailed answers and explanations will be provided to assist you in self-assessment and further learning.

To conclude this chapter, we'll summarize the key takeaways, highlighting the essential concepts and insights gained from our exploration of Open Connectors. This comprehensive overview will equip you with a solid foundation in utilizing Open Connectors effectively and efficiently in your integration projects.

### Real-World Scenario

Imagine your business offers an online food ordering and delivery service. To improve the efficiency of its delivery procedures, your company wishes to integrate its platform with a third-party logistics provider with the business's own Twitter feed. Your business could use Open Connectors, along with a cloud integration platform like Cloud Integration, to accomplish this integration.

Your organization would first sign into Open Connectors and choose a logistics provider connector from the library of prebuilt connectors. To configure the connector, you would next supply the required authentication information and map data fields from the food ordering platform to the relevant data fields in the logistics provider system.

You would then use Cloud Integration to develop a data flow that links your food ordering platform with the logistics service provider system using the Open Connectors connector. Any data transformations or business rules required to guarantee that the data is in the proper format for the logistics provider system would be included in the data flow.

Your business can manage its delivery operations with ease once the integration is online through its meal ordering platform and can post on the Twitter feed. Orders are automatically transmitted to the logistics provider system, and the food ordering platform could update a delivery's progress in real time.

In general, Open Connectors and cloud integration platforms like Cloud Integration offer a strong and adaptable approach to interface your applications with outside services, enabling you to improve company operations and customer experiences.

## 4.1 Objectives of This Portion of the Test

This chapter reviews the features of the Open Connectors capability in SAP Integration Suite. This chapter describes how to work with Open Connectors to integrate with non-SAP applications.

The goal of Open Connectors is to offer a cloud-based integration platform that enables companies to link and integrate their applications with third-party services quickly and effectively. The platform offers a number of prebuilt connectors you can quickly utilize to link with other applications and services, such as social media platforms, payment gateways, CRM systems, and others.



### Note

On the certification exam, this topic will involve a minimum of 9 to 10 questions. The weighting of this topic is more than 12%. You must provide at least 7 correct answers to clear the certification exam. The maximum time that you should spend on questions related to this chapter is 22 to 24 minutes.

## 4.2 Features

A consistent API, prebuilt connector libraries, a flexible integration platform, and a real-time data synchronization engine are just a few of the benefits offered by Open Connectors. Additionally, you'll use a simple visual interface to quickly develop, test, and deploy integrations.

Businesses can speed up time-to-market for their applications, decrease development costs, and simplify data integration procedures with Open Connectors. Additionally, the platform helps businesses understand their data better and derive insights that can enhance customer service and corporate processes.

Using standardized APIs from hubs for CRM solutions, Open Connectors can access data from non-SAP applications. The API Management feature, which allows for the application of all security best practices and governance guidelines, is responsible for managing these harmonized APIs. Ultimately, this synchronized data from any non-SAP CRM application of your choice is shown via an SAP Fiori application.

To combine API administration and integration with a data-centric approach, Open Connectors was designed from the ground up. The platform is designed to unify the developer experience across all types of applications and services, regardless of the application's backend, REST, SOAP, proprietary software development kit (SDK), database, and so on. The platform also includes more than 160 unique connectors. Through a standards-based implementation, Open Connectors unifies the API layer for use in all environments. In this way, you can ensure that integration users, developers, and their use cases are independent of the backend services they rely on.

Some other primary features of Open Connectors include the following:

- Open Connectors utilize prebuilt connectors for integration scenarios. Open Connectors provide a vast collection of prebuilt connectors for seamless integration between various applications and systems. These connectors are specifically designed to simplify and accelerate an integration process, allowing users to connect and interact with a wide range of cloud and on-premise applications without the need for extensive coding or development effort.
- Open Connectors create and apply complex integration workflow templates. With Open Connectors, you can create and apply complex integration workflow templates. These templates serve as blueprints for defining and automating integration processes, allowing for the efficient orchestration of data and activities across different systems. By leveraging these templates, your organization can streamline its integration workflows, reduce manual effort, and ensure consistency and reliability in its data exchange processes.
- Open Connectors enable users to create resource definitions and mappings for their cloud applications. This feature allows for the easy configuration and management of data mappings between different applications, ensuring that data is properly synchronized and aligned across integrated systems. By defining and mapping resources, an organization can establish a unified view of its data and maintain data integrity throughout its application landscape.
- Open Connectors offer the flexibility to add additional connectors or create custom connectors. Thus, users can extend the connectivity capabilities of the platform by integrating with specific applications or systems that are not included in the prebuilt connector library. By adding or creating connectors, your organization can achieve seamless integration with its unique set of applications and customize the integration experience according to its specific requirements.
- For queries, you can use the Open Connectors query language. This powerful query language allows users to perform inquiries and retrieve specific data from connected systems. This language offers a convenient and standardized way to query data across different applications, thus enabling users to retrieve the information they need efficiently. By leveraging the query language, your organization can gain valuable insights from its integrated systems and make data-driven decisions.
- To upload and download data from an endpoint in bulk in a normalized fashion, you can use Open Connectors bulk services. These services enable users to upload and download data from endpoints in a normalized fashion. As a result, large volumes of data can be efficiently transferred between systems in a structured and organized manner. The bulk services feature ensures that data transfers are optimized for performance and scalability, making it ideal for scenarios that involve handling significant volumes of data.
- With Open Connectors, you can set security parameters and create users and accounts. Open Connectors provides robust security capabilities, allowing users



to set security parameters and create user accounts. This feature ensures that integration processes are secure and protected from unauthorized access. Users can define access controls, permissions, and authentication mechanisms to safeguard their data and ensure compliance with security standards and regulations.

- For alerts on resource changes, use the Open Connectors events framework. This framework allows users to be notified whenever a resource changes in a connected system. With this feature, your organization can stay informed about updates, modifications, or events occurring in its integrated applications. By leveraging the events framework, users can proactively monitor changes and take appropriate actions in response, ensuring the timely synchronization of data and ensuring data consistency across systems.
- You can employ OAuth proxies to connect one endpoint application to many contexts, like development, quality assurance (QA), and others. Open Connectors provides this feature to enable user to connect one endpoint application to multiple contexts. This capability simplifies the management and configuration of connections, allowing users to switch between different environments or contexts seamlessly. By utilizing OAuth proxy, an organization can efficiently manage its integration setups across various stages of the application lifecycle and streamline its development and testing processes.

4.3 How Open Connectors Fit into SAP Integration Suite

Through the Open Connectors cloud-based integration platform, you can quickly connect to and integrate with a variety of external systems and services, including CRM programs, marketing automation programs, social media sites, and more. Without the need for custom coding, Open Connectors is a set of prebuilt connectors, referred to as *adapters*, which can be used to connect to systems and services.

To integrate numerous systems and applications throughout your organization, use Open Connectors, which is a component of SAP Integration Suite, a complete integration platform. SAP Integration Suite offers a number of functionalities, including process integration, data integration, and application integration.

To connect to numerous other systems and services for use as part of larger integration projects, Open Connectors, as a supplement to SAP Integration Suite, offer a quick and simple solution. An integration process can be sped up, less custom coding is required, and your integration connections can be simply managed from a single location by using Open Connectors.

To build end-to-end connection scenarios and processes, Open Connectors can be used in conjunction with other integration services from SAP Integration Suite, such as Cloud Integration and API Management. In general, Open Connectors is an essential part of SAP Integration Suite to enable your business to expand its connectivity and integration capabilities outside of the SAP ecosystem.

With Open Connectors, your business can automate and accelerate its critical business processes. By integrating SAP solutions with external applications, data can be seamlessly synchronized, eliminating manual data entry and reducing the risk of errors. For instance, when a new customer is created in Salesforce, Open Connectors can automatically update the customer’s information in a corresponding SAP system. This automation not only saves time but also ensures data consistency across systems, leading to enhanced operational efficiency.

When integrating systems, security and compliance are of course paramount. Open Connectors provides robust security measures to ensure the confidentiality, integrity, and availability of data during integration processes. This feature supports secure authentication protocols, encrypted data transmission, and data masking techniques to safeguard sensitive information. Furthermore, Open Connectors adheres to industry-standard compliance regulations, such as General Data Protection Regulation (GDPR) and Health Insurance Portability and Accountability Act (HIPAA), enabling businesses to meet regulatory requirements and maintain data privacy.

By incorporating Open Connectors into SAP Integration Suite, your organization can leverage a comprehensive set of tools and technologies to achieve seamless and efficient integration across their entire application landscape. Whether you’re connecting SAP systems with external applications, integrating cloud applications, or enabling data synchronization between various systems, Open Connectors provide the necessary capabilities to simplify and accelerate the integration process, ultimately helping your business achieve its digital transformation goals.

Open Connectors should be considered a capability of SAP Integration Suite, as shown in Figure 4.1.

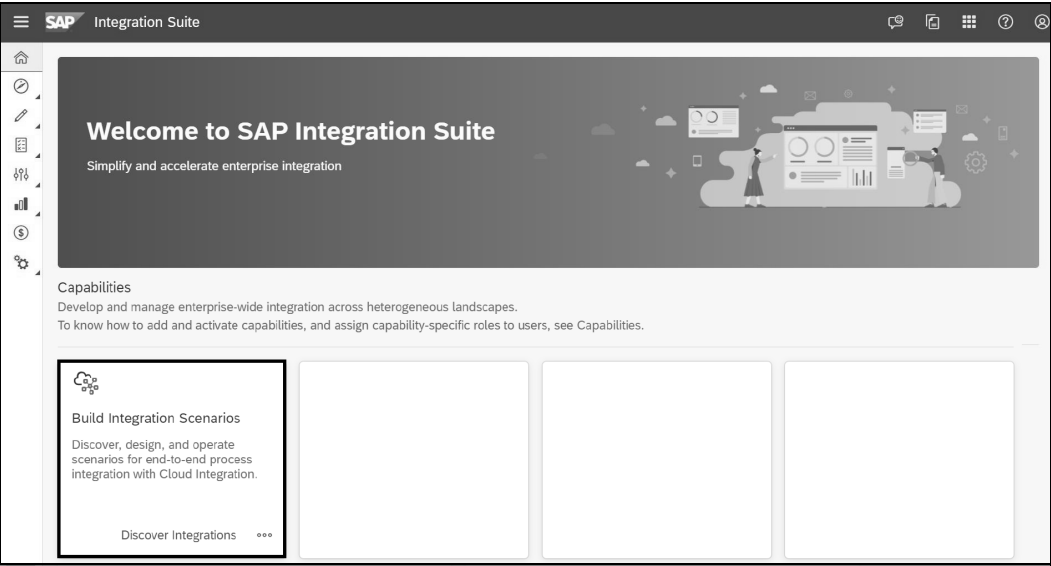


Figure 4.1 Open Connectors as a Capability in SAP Integration Suite

4.4 Working with Open Connectors

In this section, you’ll learn about creating instances and connectors, and we’ll show you how to use instances in API Management.

To use the cloud connector with API Management, you must first perform the following steps:

- 1. Log on to Open Connectors.
- 2. Create a Twitter instance.
- 3. Create an API provider with Open Connectors.
- 4. Create an API from the API provider.
- 5. Add policies to the API.
- 6. Create a product to bundle the API.
- 7. Create an application and get the API key.
- 8. Test the APIs.

We’ll walk through these steps in more detail in the following sections.

4.4.1 Logging On to Open Connectors

To log on to Open Connectors, you must first receive authorization from the Open Connector tenant’s administrator. This person would be responsible for adding subaccounts and other Open Connector capabilities. If you don’t have the authorization, you’ll be notified when you log on.

To add more members to this tenant, if you’re the administrator of the tenant, you must carry out these actions:

- 1. In Open Connectors, select the **Security** menu item.
- 2. Navigate to the **Identify** tab and add the following details, as shown in Figure 4.2:
  - **Member:** Enter your SID or PID, which can be the SAP Global account ID or your logon e-mail ID, depending on how your account was set up.
  - **Roles:** Enter your account admin user name.

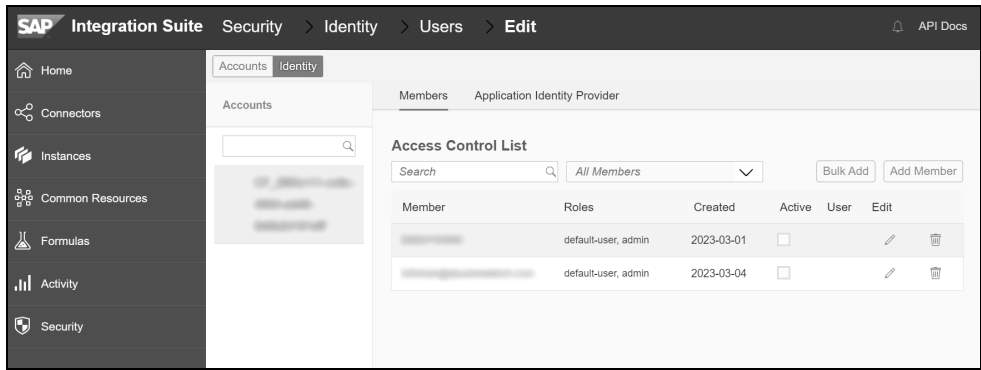


Figure 4.2 Access Control List

4.4.2 Creating a Twitter Instance

Now, let’s focus on creating a Twitter instance in Open Connectors through an example. However, note that the process of creating an instance is not limited to Twitter alone—you can follow these general steps to add any kind of instance.

In our Twitter example, we hope to provide a practical demonstration of how to establish a connection with a popular social media platform. This example allows you to understand the specific configuration requirements and steps involved in creating an instance within Open Connectors.

To create a Twitter instance, follow these steps:

- 1. Log on to Open Connectors from SAP Integration Suite.
- 2. Select the **Instances** menu item and, in the top-right corner, click **Create Instance**, as shown in Figure 4.3.

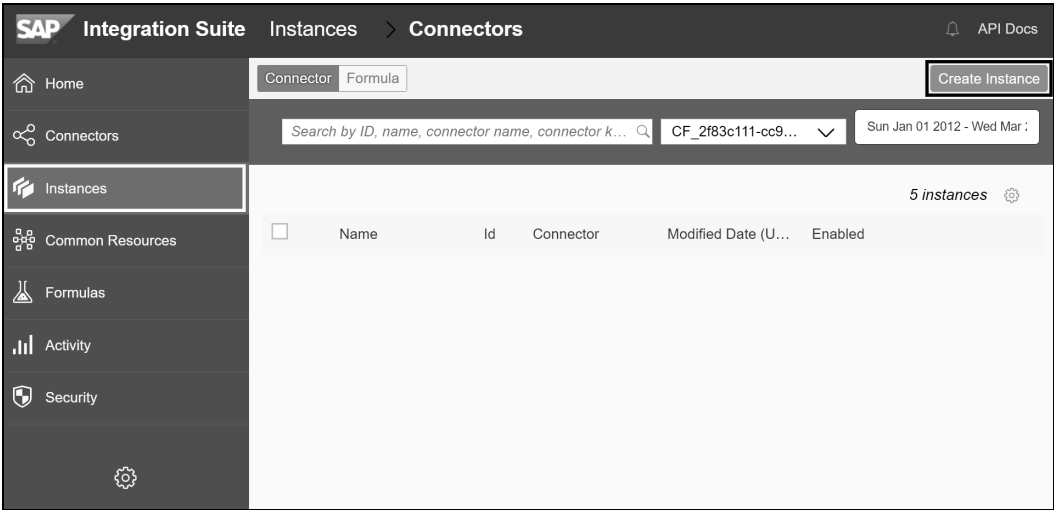


Figure 4.3 Creating an Instance

- 3. From the list of connectors, search for “Twitter” and click on the add button (+), as shown in Figure 4.4.

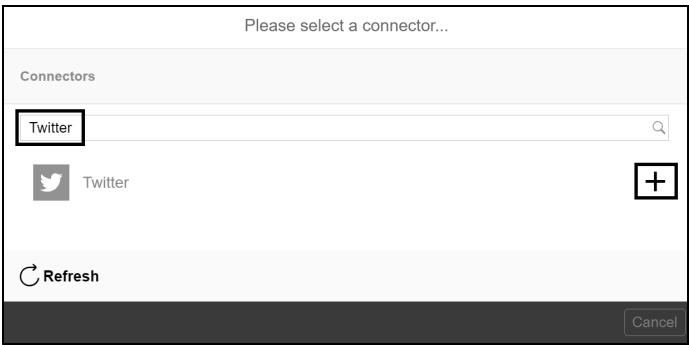


Figure 4.4 Searching for Twitter

4. Provide a name for the instance and click **Create Instance**, as shown in Figure 4.5.

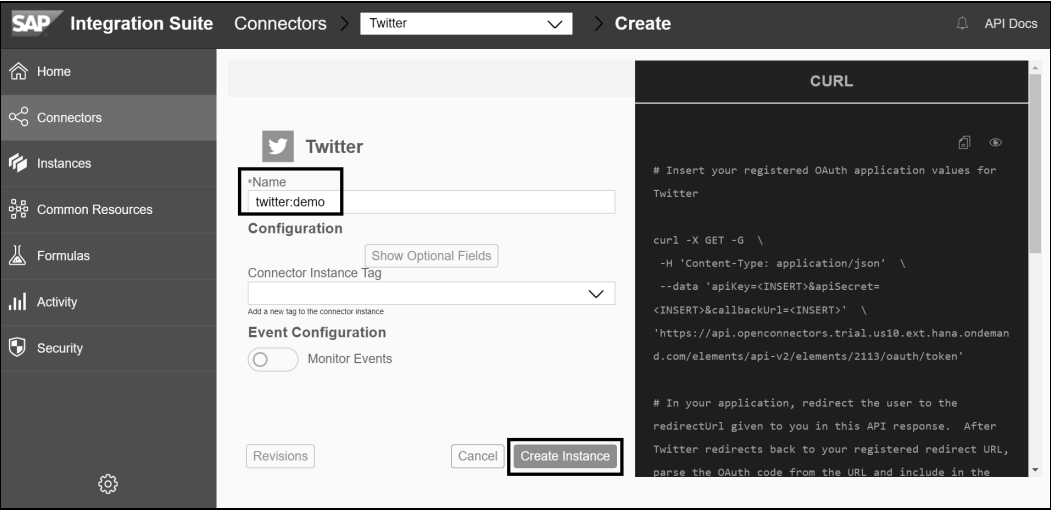


Figure 4.5 Creating an Instance

5. Open Connectors will ask permission for authorizing your Twitter account. Provide this access by clicking **Authorize app**, as shown in Figure 4.6.

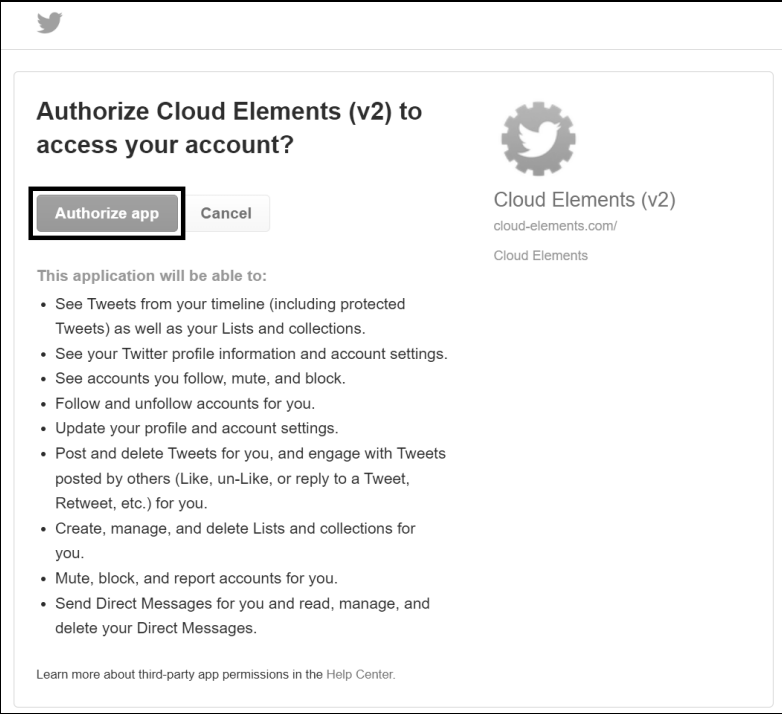


Figure 4.6 Authorizing Twitter

6. You should now find the instance created in Open Connectors. Test out some of the APIs of the Twitter instance by clicking on **API Docs**, as shown in Figure 4.7.

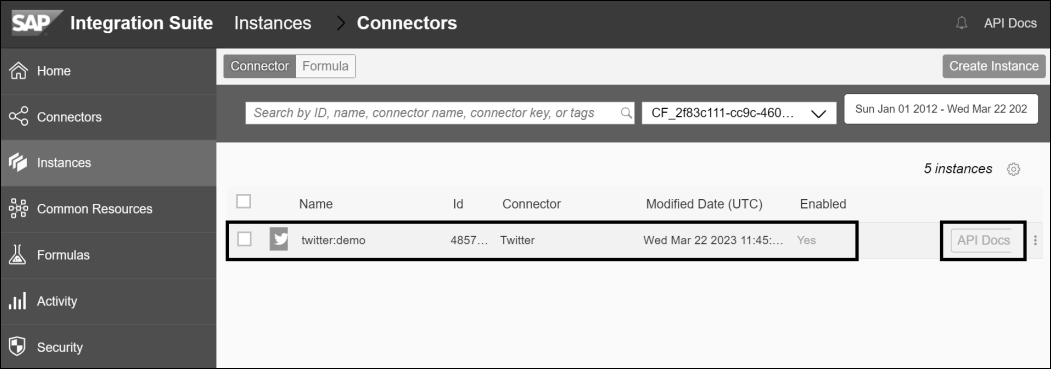


Figure 4.7 Instance Created

7. To test an API, select the API and click **Try Out**. You'll see the authorization string and the **Execute** button, as shown in Figure 4.8.

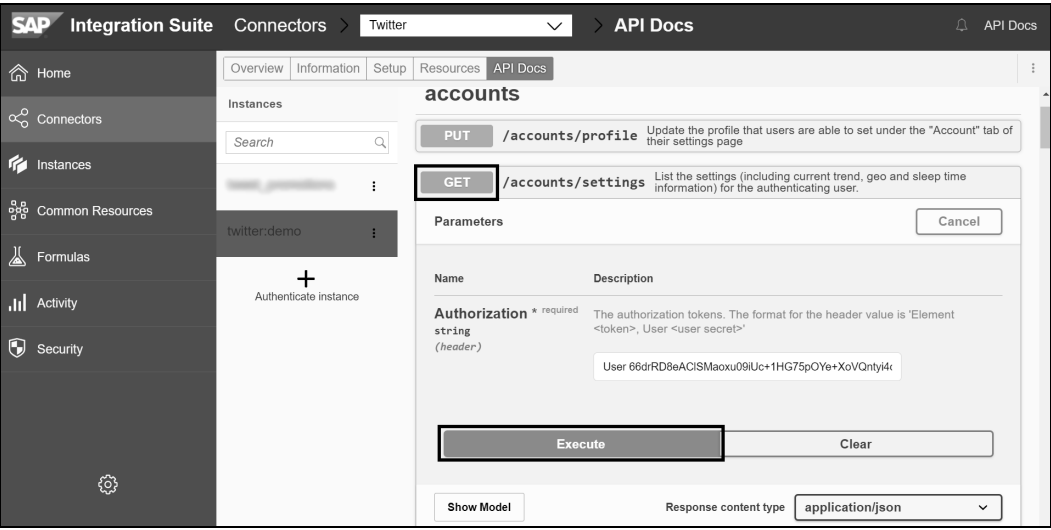


Figure 4.8 Executing the API

8. Click **Execute**, and you should see the HTTP response in the output box, as shown in Figure 4.9.

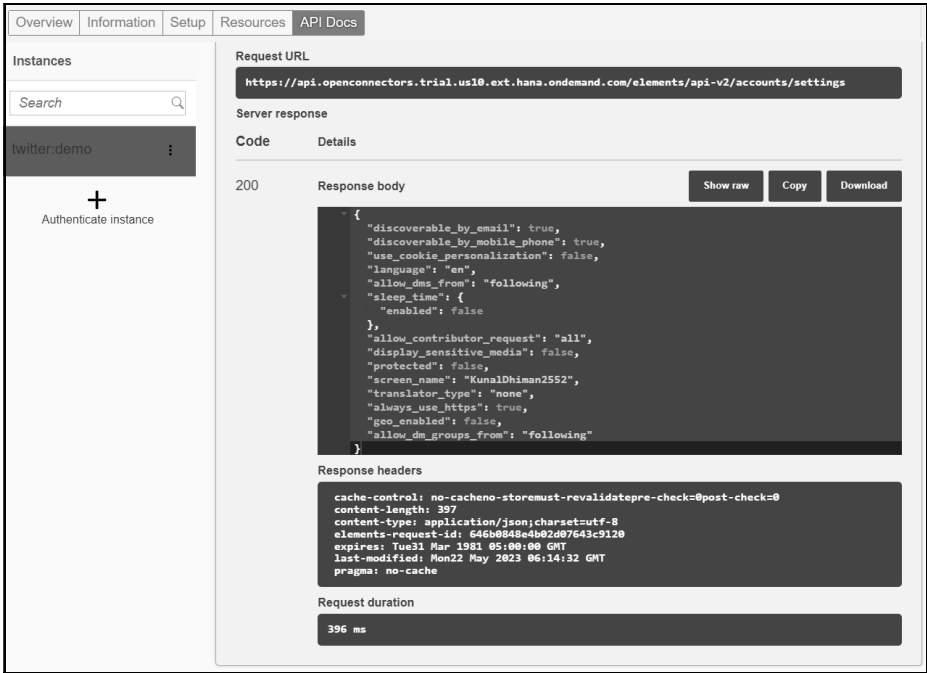


Figure 4.9 Response from the API

4.4.3 Creating an API Provider with Open Connectors

To create an API provider with Open Connectors, follow these steps:

- 1. In Open Connectors, click on the **Settings** menu item and copy the information in the **Organization Secret** and **User Secret** fields, as shown in Figure 4.10.

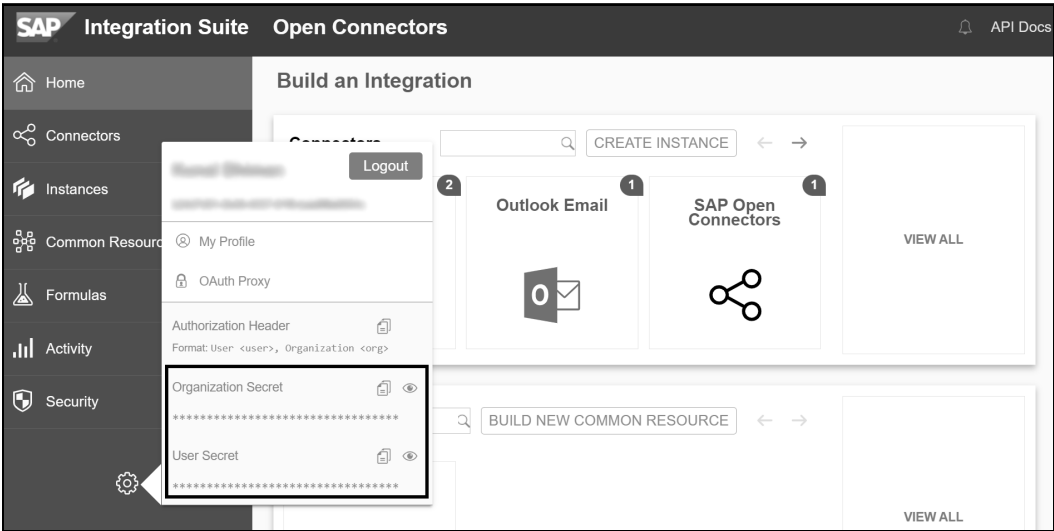


Figure 4.10 Organization and User Secret

- 2. Open API Management and select the **Configure** menu item. Create an API provider by clicking the **Create** button, as shown in Figure 4.11.

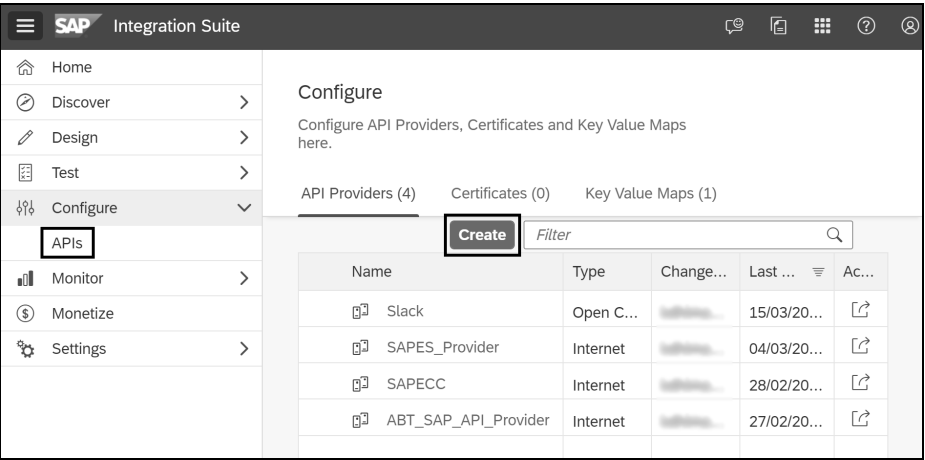


Figure 4.11 Creating an API Provider

- 3. Provide a name for the API provider under the **Overview** tab. Then, under the **Connection** tab, maintain the following details, as shown in Figure 4.12:

  - **Type:** Choose the **Open Connectors** option from this dropdown list.
  - **Region:** Select the region of your account tenant.

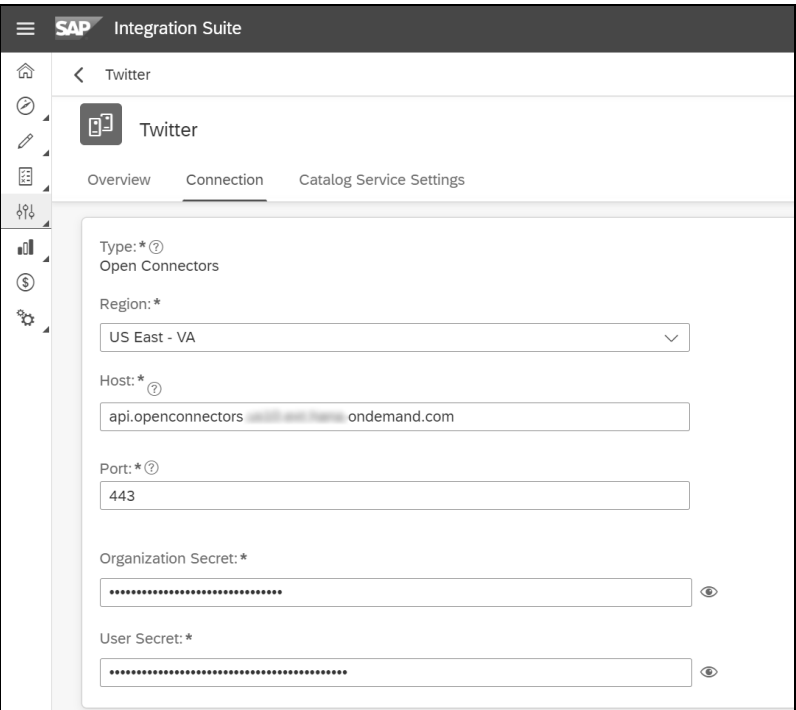


Figure 4.12 Connection Tab of the API Provider



- **Host:** The host name will be autopopulated.
- **Port:** The port number will be autopopulated.
- **Organization Secret:** Enter the information you copied earlier from the Open Connectors **Organization Secret** field.
- **User Secret:** Enter the information you copied earlier from the Open Connectors **User Secret** field.

4.4.4 Creating an Application Programming Interface from the API Provider

To create an API from the API provider, follow these steps:

1. In SAP Integration Suite, select the **Design** menu item and choose the **APIs** tab. Click **Create**, as shown in Figure 4.13.

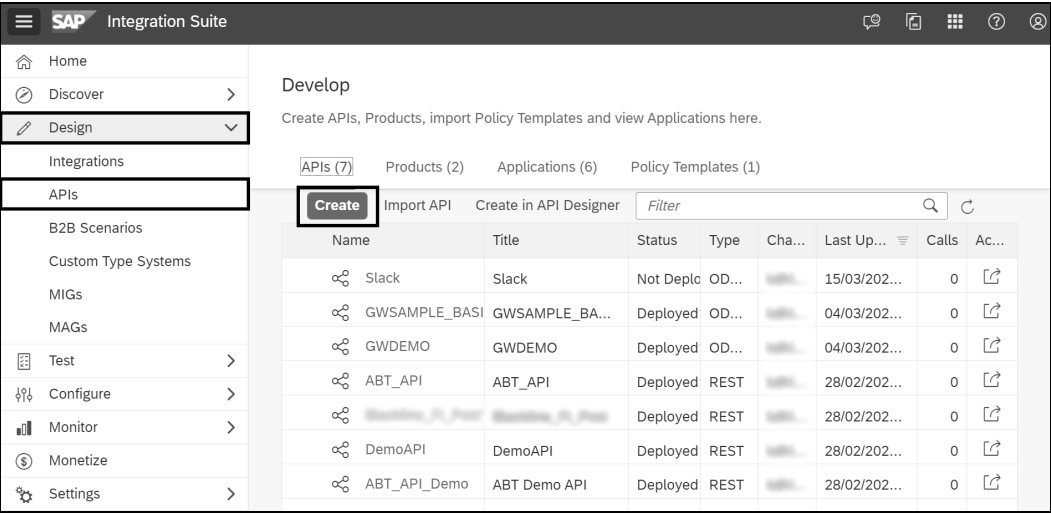


Figure 4.13 Creating an API

2. Select the **API Provider** and select the **Twitter** instance from the dropdown menu. Provide all the mandatory details for creating the API proxy, such as the following:
  - **API Provider:** Provide the name of the API provider we created earlier.
  - **URL:** Provide the end part URL from the HTTP request that we have to send. For example, for the URL `https://api.openconnectors.trial.us10.ext.hana.ondemand.com/elements/api-v2/accounts/settings`, enter `/elements/api-v2/accounts/settings`. You'll get the rest of the URL get from the API Provider.
  - **Name:** Enter a name for the API.
  - **Title:** Enter a short description for the API, which can be same the or different from its name.

- **API State:** Select **Active**.
- **Host Alias:** Provide the host alias from SAP Integration Suite.
- **API Base Path:** Provide the base path of the API, which must always start with a forward slash (/).

After providing all the mandatory details, click **Create**, as shown in Figure 4.14.

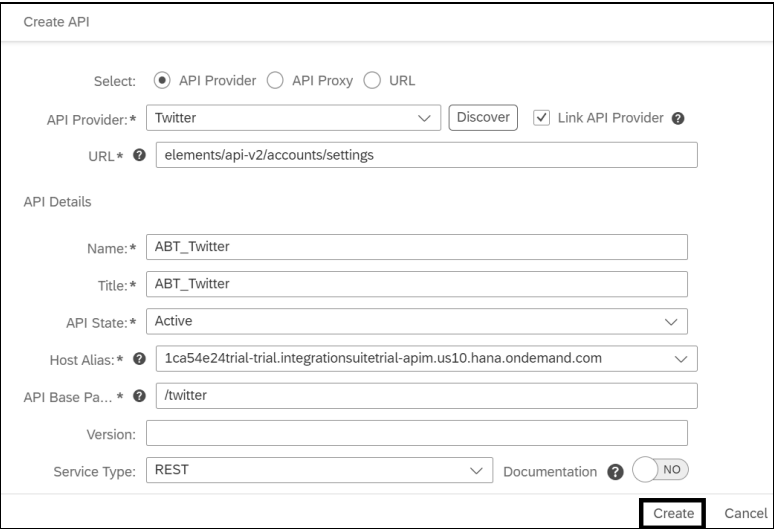


Figure 4.14 API

3. You should now see the API proxy has been created, as shown in Figure 4.15. In Open Connectors, under the **Resources** tab, under **direct-messages**, you can also try out the various messages.

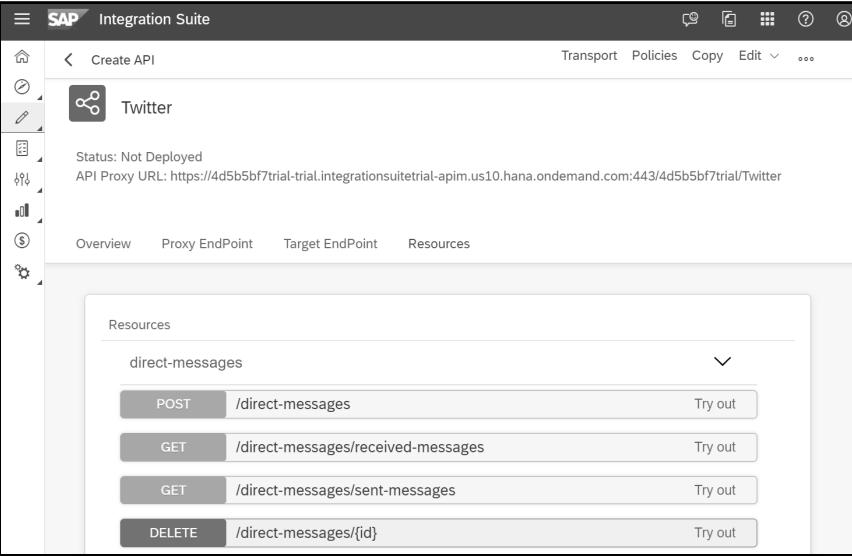


Figure 4.15 API Created

4. Save and deploy the API by clicking **Save** and then **Deploy**, as shown in Figure 4.16.

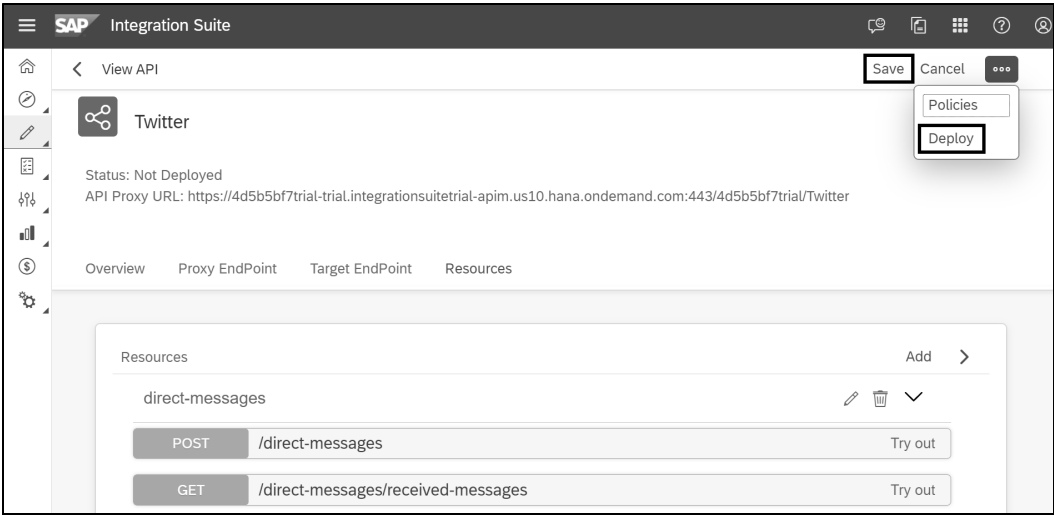


Figure 4.16 Saving and Deploying the API Proxy

4.4.5 Adding Policies to an Application Programming Interface

Once deployed, you can add policies to your API by following these steps:

1. Open the API we created and deployed earlier and select the **Policies** menu item, as shown in Figure 4.17.

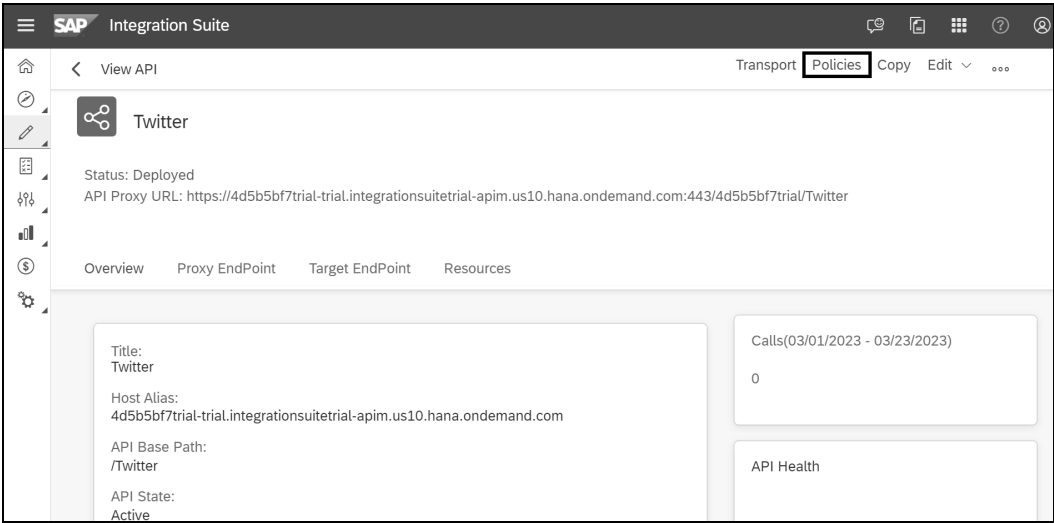


Figure 4.17 Adding Policies

2. Now, we'll add the `Verify_API_Key` in the `ProxyEndpoint` preflow of the API. Open the `ProxyEndpoint` preflow, select **Security Policies**, and click **Verify API Key (+)**. The policy will be added to the preflow. Click on the recently added policy (**Verify API Key**) in the diagram with the **(+)** sign. Enter the code shown in Listing 4.1 into the policy, as shown in Figure 4.18.

```
<!--Specify in the APIKey element where to look for the variable
containing the api key-->
<VerifyAPIKey async='true' continueOnError='false' enabled='true'
xmlns='http://www.sap.com/apimgmt'>
  <APIKey ref='request.header.APIKey' />
</VerifyAPIKey>
```

Listing 4.1 Verify\_API\_Key

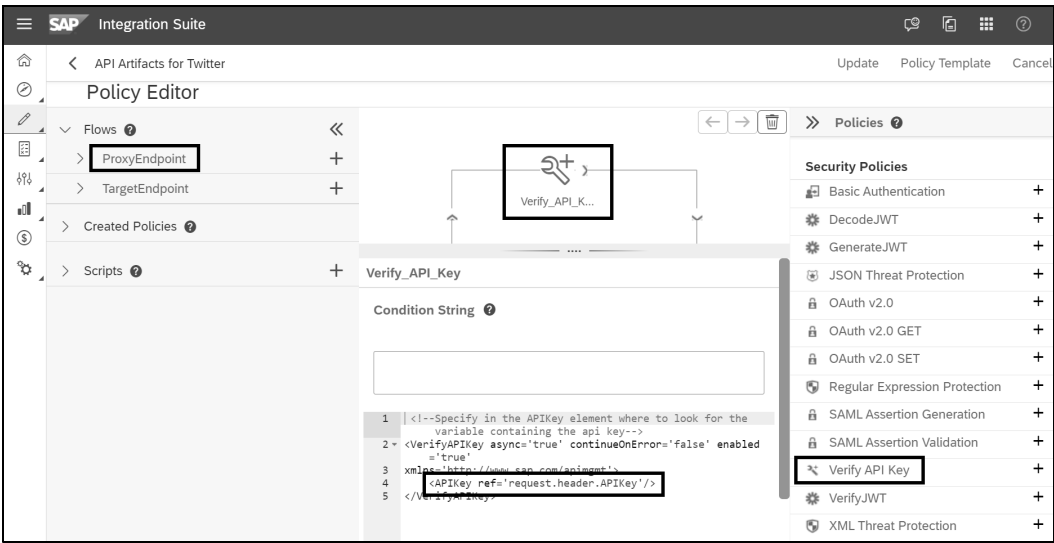


Figure 4.18 Adding an API Key

3. Save and deploy the API by clicking **Save** and then **Deploy**.

4.4.6 Creating a Product to Bundle with the Application Programming Interface

To create a product and bundle it to the API, follow these steps:

1. In the SAP Integration Suite, select the **Design** menu item. Select the **Products** tab and click **Create**, as shown in Figure 4.19.
2. Navigate to the **Overview** menu item and maintain the **Name** and the **Title** fields for the product, as shown in Figure 4.20. You can use the same name and title or use different names. You can use any name you want so long as within the boundaries of your business requirements.

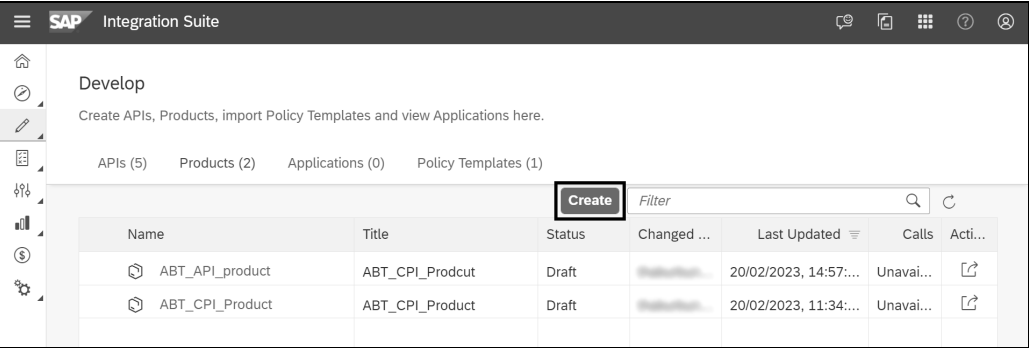


Figure 4.19 Creating Products

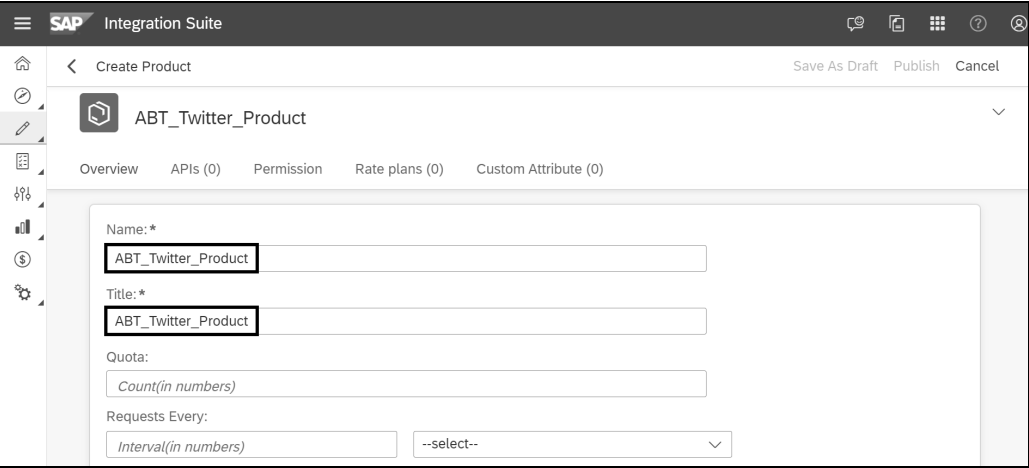


Figure 4.20 Overview

3. Navigate to the **APIs** tab and add APIs to the product, as shown in Figure 4.21. Click the **Add** button and select the API we created earlier.

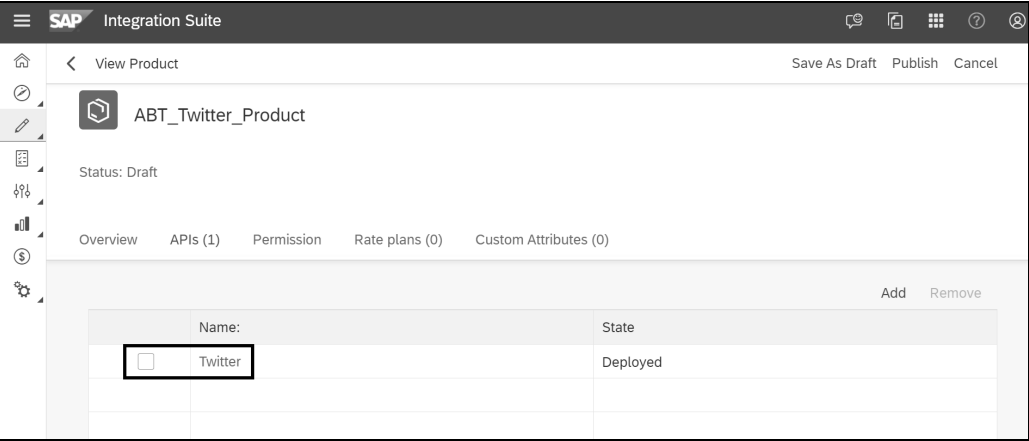


Figure 4.21 Adding APIs

4. Save and publish the product by clicking **Save as Draft** and then **Publish**, as shown in Figure 4.22.

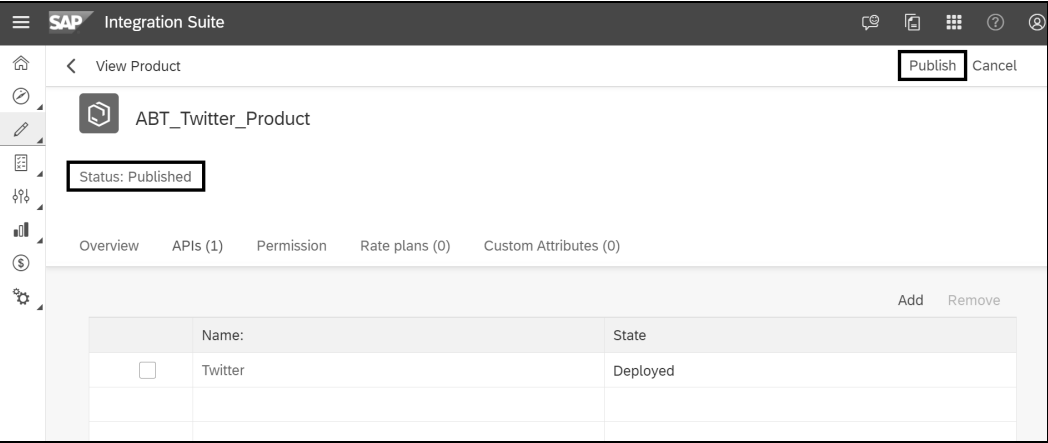


Figure 4.22 Publishing a Product

### 4.4.7 Creating an Application

Now, let's create an application by following these steps:

1. In SAP Integration Suite, in the top-right corner, select **API Business Hub Enterprise**, as shown in Figure 4.23.

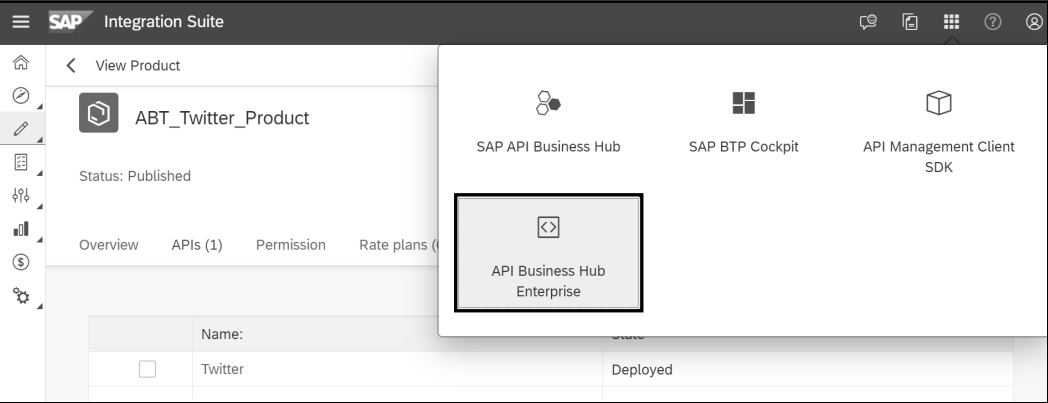


Figure 4.23 SAP Business Accelerator Hub Enterprise

2. Click on the product you're working with on the **API Business Hub Enterprise** homepage, as shown in Figure 4.24.

3. Select **Subscribe • Create Application**, as shown in Figure 4.25.

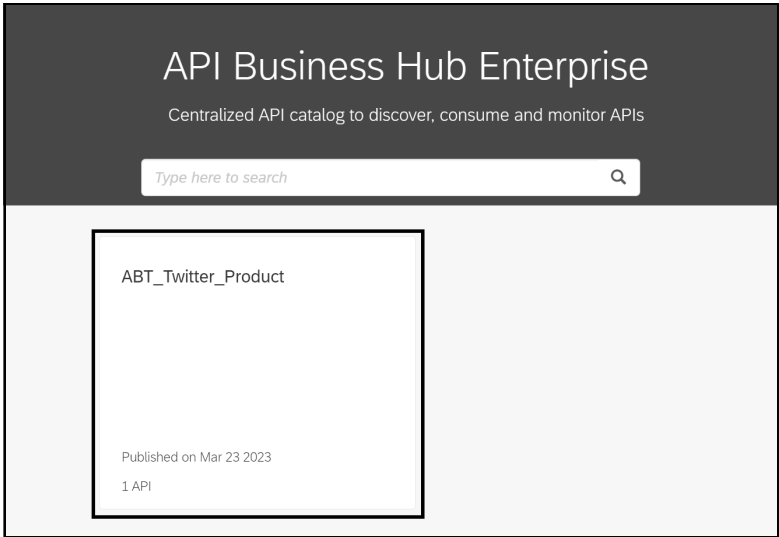


Figure 4.24 Product

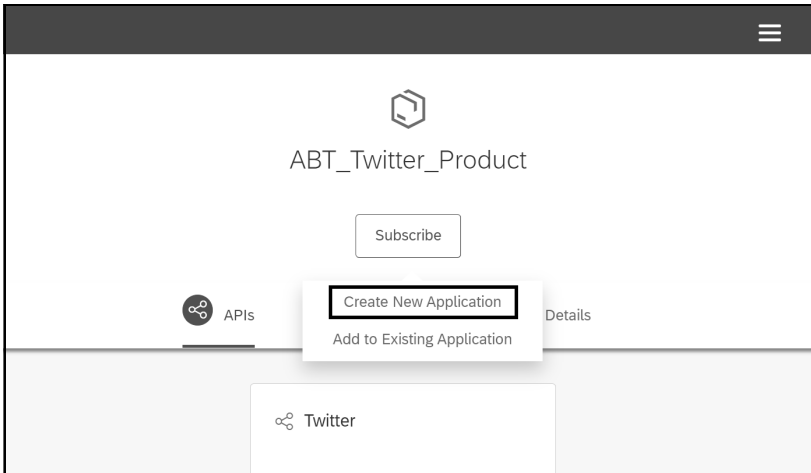


Figure 4.25 Creating an Application

4. Create the application with the mandatory details like name of the application and provide a product to the application. Then, click **Save**, as shown in Figure 4.26.
5. To get the API key, access the **Application Info** page by clicking on our newly created application. Then, copy the value in the **Application Key** field, as shown in Figure 4.27. You'll need this information handy for the next step.

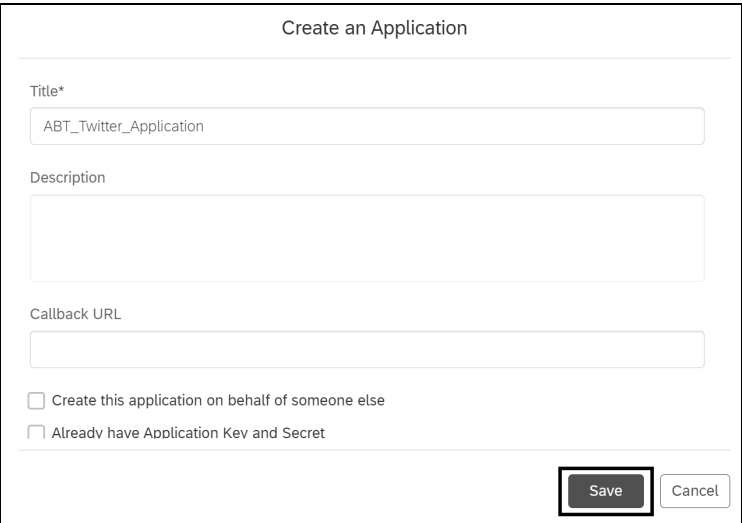


Figure 4.26 Saving an Application

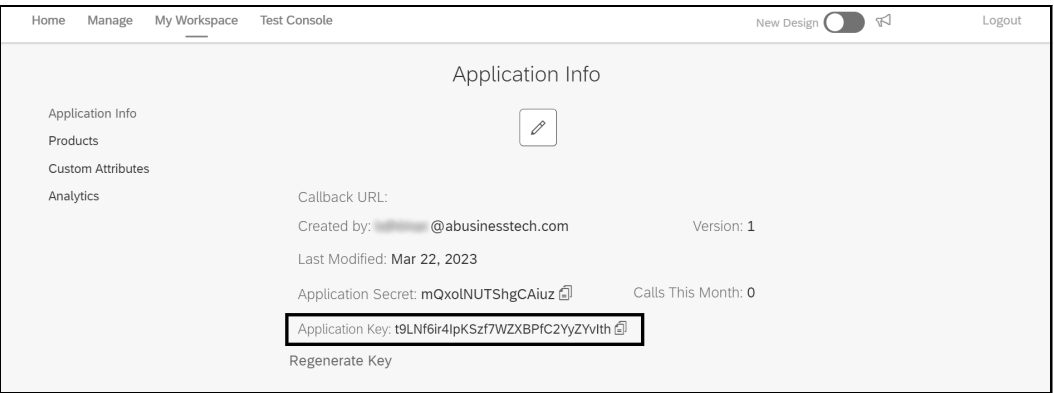


Figure 4.27 Application Key

### 4.4.8 Testing an Application Programming Interface

To test your API, follow these steps:

1. In SAP Integration Suite, select the **Test** menu item from the left.
2. Choose the API that you want to test from the dropdown list.
3. Set the HTTP method type by selecting the **GET** radio button.
4. In the **Headers** section, paste the API key you copied earlier from SAP Business Accelerator Hub Enterprise.
5. Click the **Send** button, as shown in Figure 4.28.

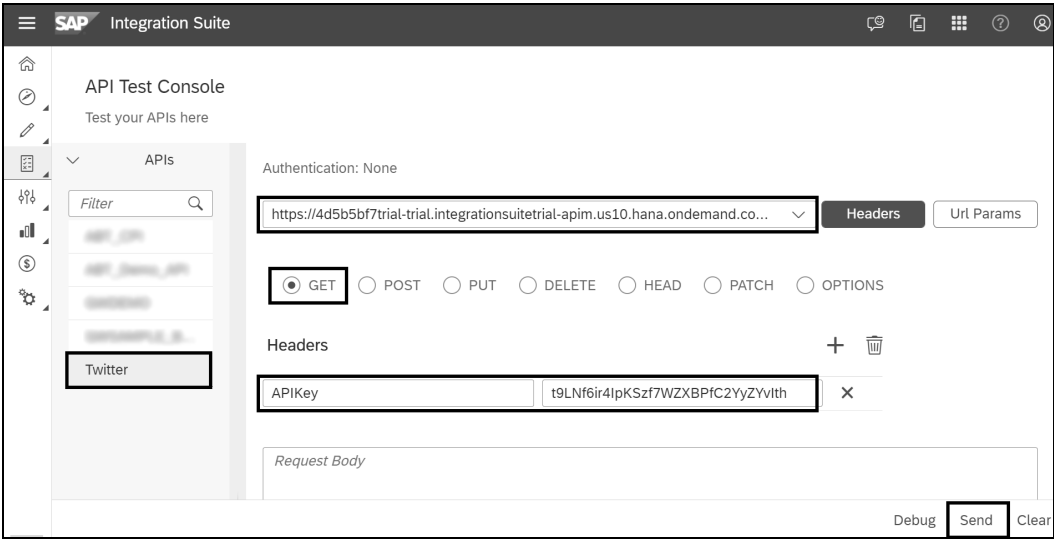


Figure 4.28 API Test Console

6. You'll see the Twitter API, which we covered in Section 4.4.2, sending back the same **200 OK** response, as shown in Figure 4.29.

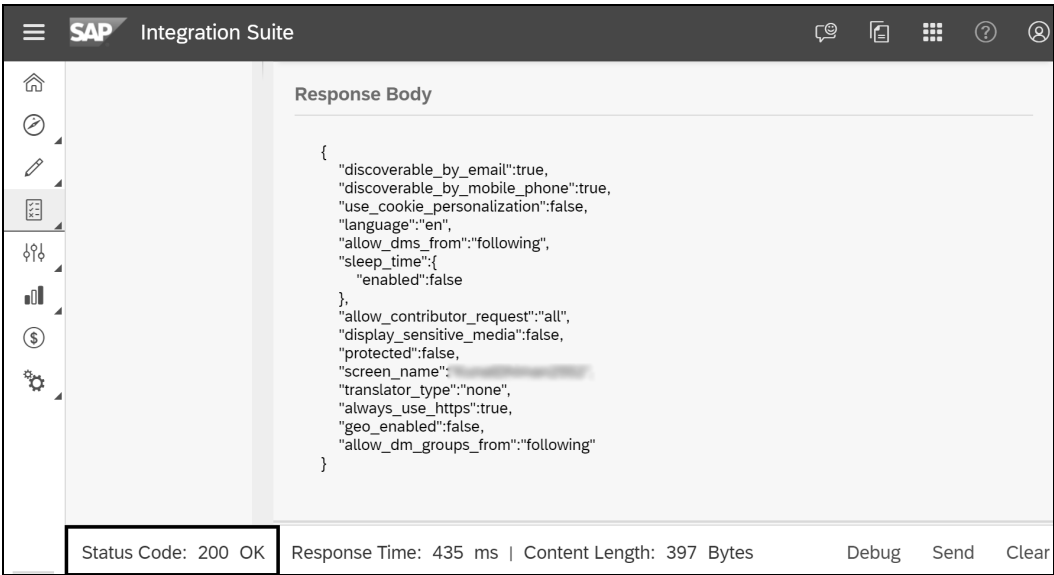


Figure 4.29 Successful Test

## 4.5 Important Terminology

In this chapter, we used the following terminology:

- **Application programming interface (API)**  
An API is a group of protocols, procedures, and tools used to create software

programs and enable system communication. Developers can interface with connectors and access the features and capabilities of third-party apps or services through the API, which offers a standardized method.

- **Connector**  
A ready-made integration, a connector offers a standardized, prebuilt connection to a particular application or service from a third party. Developers and users can simply connect to a variety of apps and services by using connectors, which include all the configuration and authentication information required for a certain API and eliminate the need to build bespoke integration code.
- **Instance**  
An instance is a specific implementation of a connection for a given user account. Each instance is started by a user, and it is linked to a set of authentication credentials that provide the connector access to data in the third-party program or service it is integrated with. Instances can be set up to carry out tasks or react to particular occasions, such as for creating a new record in a customer relationship management (CRM) database whenever a new order is placed on an e-commerce platform. Users can manage connector integrations and automate workflows with more control and flexibility by using instances.
- **Endpoint**  
An endpoint is a URL used to access a particular API resource. Endpoints are essential to API integration because they specify the location and features of the API with which a connector will communicate. Endpoints can be used to update records, retrieve data from a third-party application, and carry out other operations that the API specifies. Users can define the precise endpoints they require interaction with a certain API by configuring endpoints for their connector instances within Open Connectors.
- **Authentication**  
With Open Connectors, authentication is the process of confirming a user's or system's identity in order to provide secure access to APIs and resources. Open Connectors supports many authentication methods, including OAuth 2.0, API keys, and basic authentication. Users can allow access to their data without disclosing their logon information thanks to the widely used standard authentication and authorization mechanism known as OAuth 2.0. Whereas basic authentication uses a user name and password to authenticate users, API keys offer a special identifier that is used to authenticate API requests.
- **OAuth**  
OAuth is a widely accepted authentication and authorization mechanism that is employed by Open Connectors. Users can safely offer outside applications access to their data without disclosing their logon information. By giving a third-party application access to their Open Connectors account, users can authenticate themselves with that application via OAuth, and they can then allow that application permission to access particular data or take particular actions on their behalf.



■ **API key**

An API key is a special identification number that is used to verify API requests and queries. When users create an instance of a connection, the system normally generates API keys and provides them to users. These keys are employed to confirm the user's identity and guarantee that only valid requests are handled by the connection.

■ **Event**

An event is a trigger in a connector or flow that starts a certain action. The connector can be set up to automatically carry out a connected action in response to a certain occurrence, such as the addition of a new record to a third-party program, or to send a notification to a particular user.

■ **Payload**

The information supplied or received in an API request or response is referred to as the payload. Its contents may include details about a user's logon information, information about a particular record in a third-party application, or directions for a connector or flow to carry out a certain operation.

**4.6 Practice Questions**

1. Which one of the authentication techniques is most frequently applied to Open Connectors?  
*Select the correct answer:*
  - ☐ A. OAuth
  - ☐ B. Basic authentication
  - ☐ C. Token-based authentication
  - ☐ D. All of the above
2. What does an Open Connectors connector instance mean?  
*Select the correct answer:*
  - ☐ A. A prebuilt integration for a particular application or service from a third party
  - ☐ B. A particular connector implementation for a given user account
  - ☐ C. A particular API endpoint used to access a resource
  - ☐ D. A special identification number used to verify API requests

3. Which of the following parameters must be used when creating an instance in Open Connectors?  
*Select the correct answer:*
  - ☐ A. Connector type
  - ☐ B. Instance name
  - ☐ C. Authentication type
  - ☐ D. API endpoint URL
4. How can you find the APIs in the instance created?  
*Select the correct answer:*
  - ☐ A. By clicking on the instance
  - ☐ B. By clicking on the API docs
  - ☐ C. Under the **Resources** tab
  - ☐ D. Under the **Security** tab
5. While testing out an API in the instance, what are the three values in the Authorization header?  
*Select one or more answers:*
  - ☐ A. User
  - ☐ B. Element
  - ☐ C. Organization
  - ☐ D. Instance token
6. While trying out the APIs in the API docs, what is the format of the Authorization header?  
*Select the correct answer:*
  - ☐ A. User <user secret>, token <value>
  - ☐ B. User <user>, organization <org>
  - ☐ C. Element <token>, organization <value>
  - ☐ D. Organization <secret>, user <secret>
7. What is the first step in the Open Connectors instance creation process?  
*Select the correct answer:*
  - ☐ A. Decide which connector to use
  - ☐ B. Establish an Open Connectors account
  - ☐ C. Provide the necessary authentication information
  - ☐ D. Develop the API endpoints

8. What is an Open Connectors API key?

*Select the correct answer:*

- ☐ A. A special identification number used to verify API requests
- ☐ B. A configuration setting identifying the intended API endpoint
- ☐ C. An API call authorization authentication token
- ☐ D. A data format for information transmission between the client and the API

9. With Open Connectors, how can you test a connection to an API?

*Select one or more answers:*

- ☐ A. On the instance configuration screen, click the **Test Connection** button
- ☐ B. Make use of the tool Open Connectors API test console
- ☐ C. Send a test API request and check the outcome
- ☐ D. All the above

10. Which of the following protocols does API Management support while creating an API proxy?

*Select the correct answer:*

- ☐ A. OData
- ☐ B. REST
- ☐ C. SOAP
- ☐ D. All of the above

11. What is an advantage of using Open Connectors?

*Select the correct answer:*

- ☐ A. Increased API security
- ☐ B. Integration of third-party apps is made simpler
- ☐ C. Increased API performance
- ☐ D. Improved API scalability

12. What are the connection types available in API Management while creating an API provider?

*Select the correct answer:*

- ☐ A. Internet, on-premise, Open Connectors, and Cloud Integration
- ☐ B. Internet, open connector, process integration, and internet
- ☐ C. Cloud Integration, process integration, on-premise, and internet
- ☐ D. Cloud-to-on-premise, internet, Cloud Integration, and Open Connectors

13. What are the two mandatory values required from cloud connectors while creating the Open Connector type API provider?

*Select one or more answers:*

- ☐ A. Organization secret
- ☐ B. Element
- ☐ C. Instance token
- ☐ D. User secret

14. What is the purpose of the connector instance ID in Open Connectors?

*Select the correct answer:*

- ☐ A. With Open Connectors, this ID is used to identify a specific connector.
- ☐ B. This ID is applied to the target system's connector authentication.
- ☐ C. This ID is used to specify which connection version to use.
- ☐ D. This ID is optional while making a connector.

15. Where can you find the user and organization secret in Open Connectors?

*Select the correct answer:*

- ☐ A. Under **Settings**
- ☐ B. Under **Instances**
- ☐ C. Under **Common Resources**
- ☐ D. None of the above

16. How is API versioning handled by Open Connectors?

*Select the correct answer:*

- ☐ A. It automatically updates to the latest API version.
- ☐ B. It allows users to select the API version they want to use.
- ☐ C. It uses a fixed API version that cannot be changed.
- ☐ D. It does not support API versioning.

17. Which of the following pertains to Open Connectors' features?

*Select the correct answer:*

- ☐ A. Data backups that are automated
- ☐ B. Predictive modeling
- ☐ C. Real-time event notifications
- ☐ D. Geolocation tracking

18. Where can you add a member in Open Connectors?

*Select the correct answer:*

- ☐ A. Under **Settings**
- ☐ B. Under **Formulas**
- ☐ C. Under **Security**
- ☐ D. Under **Instances**

19. What are the two mandatory details needed while adding the member in Open Connectors?

*Select one or more answers:*

- ☐ A. Your SID/PID
- ☐ B. Roles
- ☐ C. Name
- ☐ D. Organization name

20. Which of the following statements most accurately sums up how instance tokens are used by Open Connectors?

*Select the correct answer:*

- ☐ A. Open Connectors uses them to authenticate user credentials.
- ☐ B. They are utilized to monitor the condition of an SAP system that's connected using Open Connectors.
- ☐ C. They are used to specify the precise SAP system instance that will be utilized for data transfer.
- ☐ D. Data transport over Open Connectors is encrypted using them.

21. In Open Connectors, what is the purpose of defining a connection?

*Select the correct answer:*

- ☐ A. To establish a secure network connection between SAP systems
- ☐ B. To define the data structure for incoming and outgoing messages
- ☐ C. To specify the authentication details and communication parameters for connecting to an external system
- ☐ D. To configure the mapping between source and target fields in data integration

22. Can multiple instances of the same project be created simultaneously in Open Connectors?

*Select the correct answer:*

- ☐ A. Yes, multiple instances can be created concurrently.
- ☐ B. No, only one instance can be created at a time.
- ☐ C. It depends on the available system resources.
- ☐ D. Multiple instances can only be created for high-priority projects.

23. How does the creation of an instance in Open Connectors benefit the API provider in API Management?

*Select the correct answer:*

- ☐ A. It allows the API provider to expose APIs to external consumers.
- ☐ B. It enables the API provider to define the business logic for API operations.
- ☐ C. It provides a secure and scalable connection to external systems.
- ☐ D. It facilitates the management and monitoring of API usage and performance.

24. Which authentication method is required to create a Twitter instance in Open Connectors?

*Select the correct answer:*

- ☐ A. Basic authentication
- ☐ B. OAuth 2.0
- ☐ C. API key authentication
- ☐ D. JSON Web Token (JWT) authentication

25. What is the purpose of creating a Twitter instance in Open Connectors?

*Select the correct answer:*

- ☐ A. To establish a connection between SAP systems and Twitter
- ☐ B. To generate automated tweets using Open Connectors
- ☐ C. To enable real-time data synchronization between SAP and Twitter
- ☐ D. To import Twitter user data into SAP databases

## 4.7 Practice Answers and Explanations

1. Correct Answer: A

OAuth authentication is the most common technique used for authentication in Open Connectors.

2. Correct Answer: **B**

In Open Connectors, the connector instance is a specific implementation of a connector for a particular user account.

3. Correct Answer: **A**

Specifying the connector type is the most important step to create an instance. Without the connector type, creating an instance is not relevant and not possible.

4. Correct Answer: **B**

Once an instance has been created, the APIs are easily accessible if we find them in the API docs.

5. Correct Answers: **A, B, and C**

The values that we receive from the header of an API are user, element, and organization.

6. Correct Answer: **B**

The format of the Authorization header is User <user>, Organization <org>.

7. Correct Answer: **A**

The initial step when creating an instance in Open Connectors is to specify which connector we must use.

8. Correct Answer: **A**

An API key in Open Connectors is a unique identifier for authenticating requests to an API.

9. Correct Answer: **D**

To test a connection to an API using Open Connectors, you can click the **Test Connection** button or send a test request. You can also use the API test console.

10. Correct Answer: **D**

API Management enables the creation of an API proxy utilizing the SOAP, REST, and OData protocols, regardless of whether the API provider uses Open Connectors or not.

11. Correct Answer: **B**

A main advantage of Open Connectors is the ease with which third-party applications can be integrated. Interoperability is made possible by prebuilt connectors that offer preconfigured connectivity with well-known third-party programs like Salesforce, NetSuite, and Marketo.

12. Correct Answer: **A**

In API Management, you can create an API provider, which enables you to create an API proxy. Options while creating API providers include internet, on-premise, Open Connectors, and Cloud Integration.

13. Correct Answers: **A and D**

The organization secret and the user secret are the two values required while creating an API provider.

14. Correct Answer: **A**

The purpose of connector instance ID in Open Connectors is to uniquely identify the connector within Open Connectors.

15. Correct Answer: **A**

The organization secret and the user secret are available by clicking the **Settings** button in the Open Connectors UI.

16. Correct Answer: **B**

Open Connectors always allows a user to select the API versions they want to use.

17. Correct Answer: **C**

Open Connectors provides real-time event notifications, one of its greatest features.

18. Correct Answer: **C**

Open Connectors allow you to add a member by navigating to the **Security** tab; under the **Identity** section, you can add members.

19. Correct Answers: **A and B**

When you add a new member in Open Connectors, you must provide an SID or PID along with the role that you want to assign them.

20. Correct Answer: **C**

An instance token in Open Connectors is used to identify the specific instance of an SAP system to be used for data transfer.

21. Correct Answer: **C**

The purpose of defining a connection in Open Connectors is to specify the authentication details and communication parameters required for connecting to an external system. This information allows the system to establish a connection with the external system and enable data transfer and integration between them.

22. Correct Answer: **A**

Multiple instances can be created concurrently in Open Connectors. In other words, you can create and run multiple instances of the same project at the same time, thus allowing for parallel processing and increased efficiency in handling data integration tasks.

23. Correct Answer: **C**

The creation of an instance in Open Connectors benefits the API provider in API Management by providing a secure and scalable connection to external systems. This feature enables an API provider to establish reliable and efficient communication with external systems, thus ensuring data integrity and facilitating seamless integration between different platforms or applications.

24. Correct Answer: **B**

OAuth 2.0 is the authentication method required to create a Twitter instance in Open Connectors. OAuth 2.0 is a widely used authentication protocol that

allows secure access to resources without disclosing the user's credentials. It enables users to authorize third-party applications to access their Twitter account on their behalf.

25. Correct Answer: A

The purpose of creating a Twitter instance in Open Connectors is to establish a connection between SAP systems and Twitter. This connection allows SAP systems to interact with Twitter, such as sending and receiving data, posting tweets, retrieving tweets, and performing various operations using the Twitter API.

## 4.8 Key Takeaways

As an integration platform, Open Connectors enables companies to connect to a range of programs and platforms without having to invest heavily in custom development. The prebuilt connectors offered by Open Connectors can be used to quickly connect to well-known apps like Salesforce, Microsoft Dynamics 365, and Oracle as well as craft generic connectors like HTTP and SOAP. Open Connectors facilitate the process of connecting to various systems, lessen the requirement for custom development, and help streamline a company's integration strategy by offering a uniform platform for connection.

Prebuilt connectors are available for a variety of applications and systems through Open Connectors. These connectors function right out of the box, so you can use them right away without the need for customization. Open Connectors offers ready-made connectors for popular apps like Salesforce or Microsoft Dynamics 365, which is especially helpful for companies that utilize those solutions. The platform also provides generic connectors, such as REST and SOAP, for connecting to a variety of applications and systems that are not supported by the prebuilt connectors.

Although prebuilt connectors are available, Open Connectors is a customizable platform. You can alter connectors to suit your particular integration requirements. For instance, you can alter the prebuilt connectors to accommodate particular data formats or requirements for data mapping. Moreover, the platform offers a connection builder tool that enables programmers to create original bespoke connectors. With this ability to customize integrations to meet specific needs, your business can optimize its integration strategies and improve outcomes.

Developers can include several members to their tenant using Open Connectors. Prebuilt connectors can be used by developers to construct their own instances and access the APIs. Using the various technologies of SAP Integration Suite, such as API Management and Cloud Integration, these APIs can be used with ease. Open Connectors are designed to use APIs to integrate different systems and applications. The APIs of the systems to be integrated must therefore be thoroughly

understood. The authentication procedures, endpoint URLs, data formats, and request and response payloads are all covered by this comprehension. OAuth, basic authentication, and API keys are just a few of the authentication types that Open Connectors allow. To verify each connector, a crucial step is choosing the right authentication technique based on the systems being merged and to apply the proper credentials.

## 4.9 Summary

With the help of APIs, you can link different applications and systems with Open Connectors, a cloud-based integration platform. This solution offers both prebuilt connectors for connecting to well-known apps like Salesforce, NetSuite, and ServiceNow, as well as bespoke connectors for integrating with any system that supports APIs. By the provision of a consistent API for data transmission, mapping, and transformation between systems, Open Connectors can streamline the integration procedure. To ensure efficient data transfer, it supports a number of authentication techniques, including OAuth, basic authentication, and API key, and it offers monitoring and debugging tools. Powerful data transformation and mapping capabilities are provided by Open Connectors to guarantee correct data transformation and mapping across your source and target systems.

These connectors are simple to configure for different authentication techniques, such as OAuth and API keys, to transport data across systems and applications. To guarantee that data is accurately mapped and transformed between systems, the platform provides powerful data transformation and mapping capabilities.



# Contents

Preface .....	13
Introduction .....	19
<b>1    Fundamentals</b> .....	<b>37</b>
<hr/>	
1.1    Objectives of This Portion of the Test .....	39
1.2    SAP Business Technology Platform .....	39
1.2.1    Capabilities .....	40
1.2.2    Technical Service Planes .....	41
1.2.3    Programming Models and Environments .....	43
1.2.4    Commercial Models .....	45
1.2.5    SAP Discovery Center .....	47
1.2.6    SAP Business Technology Platform Free Trial Account Offerings .....	50
1.3    SAP Integration Suite .....	51
1.3.1    Four Core Capabilities .....	52
1.3.2    Three-Layer Architecture .....	52
1.3.3    Setting Up Your Environment: Practice Exercise .....	54
1.3.4    Role Collections .....	59
1.3.5    Migration Assessment .....	62
1.4    Important Terminology .....	63
1.5    Practice Questions .....	64
1.6    Practice Answers and Explanations .....	70
1.7    Key Takeaways .....	73
1.8    Summary .....	74
<b>2    API Provisioning</b> .....	<b>75</b>
<hr/>	
2.1    Objectives of This Portion of the Test .....	76
2.2    Features and Capabilities .....	77
2.3    How API Management Fits into SAP Integration Suite .....	81
2.4    SAP Business Accelerator Hub Enterprise .....	82
2.4.1    Overview .....	82
2.4.2    Creating an SAP Business Accelerator Hub Enterprise Page .....	84

<b>2.5</b>	<b>API Proxies</b>	89
2.5.1	Overview and Features	89
2.5.2	Creating an API Proxy	90
2.5.3	Creating a Product	92
<b>2.6</b>	<b>Policies</b>	94
<b>2.7</b>	<b>Consumption Analytics</b>	98
<b>2.8</b>	<b>Testing APIs</b>	99
<b>2.9</b>	<b>Important Terminology</b>	102
<b>2.10</b>	<b>Practice Questions</b>	104
<b>2.11</b>	<b>Practice Answers and Explanations</b>	109
<b>2.12</b>	<b>Key Takeaways</b>	112
<b>2.13</b>	<b>Summary</b>	113
<b>3</b>	<b>Consuming and Processing APIs Using Cloud Integration</b>	115
<b>3.1</b>	<b>Objectives of This Portion of the Test</b>	117
<b>3.2</b>	<b>Cloud Integration Features</b>	117
<b>3.3</b>	<b>How Cloud Integration Fits into SAP Integration Suite</b>	119
<b>3.4</b>	<b>Navigating the Web User Interface</b>	121
3.4.1	Discover	121
3.4.2	Design	122
3.4.3	Monitor	123
3.4.4	Settings	124
<b>3.5</b>	<b>Integration Packages</b>	126
<b>3.6</b>	<b>Integration Artifacts</b>	130
3.6.1	Integration Flows	130
3.6.2	Value Mappings	131
3.6.3	SOAP Application Programming Interfaces	132
3.6.4	REST Application Programming Interfaces	134
3.6.5	OData Application Programming Interfaces	135
3.6.6	Integration Adapter	136
3.6.7	Script Collection	137
3.6.8	Message Mapping	138
<b>3.7</b>	<b>Integration Flows</b>	139
3.7.1	Integration Flow Components	139
3.7.2	Creating an Integration Flow	151

3.7.3	Using Adapters	152
3.7.4	Message Context	153
3.7.5	Message Conversion	155
3.7.6	Using Data Stores	157
3.7.7	Handling Exceptions	158
3.7.8	Deploying the Integration Flow	158
3.7.9	Managing Deployed Integration Content	159
3.7.10	Enabling Trace Logs for an Integration Flow	161
3.7.11	Testing Connectivity	162
<b>3.8</b>	<b>Creating an Integration Flow Purchase Order</b>	165
<b>3.9</b>	<b>Finding and Exploring the Purchase Order API within the SAP Business Accelerator Hub</b>	167
<b>3.10</b>	<b>Consuming the Purchase Order API from SAP Business Accelerator Hub</b>	169
<b>3.11</b>	<b>Building a Second Integration Flow for Purchase Order Notification</b>	173
<b>3.12</b>	<b>Building a Script Collection Artifact</b>	177
<b>3.13</b>	<b>Important Terminology</b>	180
<b>3.14</b>	<b>Practice Questions</b>	181
<b>3.15</b>	<b>Practice Answers and Explanations</b>	186
<b>3.16</b>	<b>Key Takeaways</b>	189
<b>3.17</b>	<b>Summary</b>	190

<b>4</b>	<b>Open Connectors</b>	191
<b>4.1</b>	<b>Objectives of This Portion of the Test</b>	194
<b>4.2</b>	<b>Features</b>	194
<b>4.3</b>	<b>How Open Connectors Fit into SAP Integration Suite</b>	196
<b>4.4</b>	<b>Working with Open Connectors</b>	198
4.4.1	Logging On to Open Connectors	198
4.4.2	Creating a Twitter Instance	199
4.4.3	Creating an API Provider with Open Connectors	202
4.4.4	Creating an Application Programming Interface from the API Provider	204
4.4.5	Adding Policies to an Application Programming Interface	206
4.4.6	Creating a Product to Bundle with the Application Programming Interface	207
4.4.7	Creating an Application	209
4.4.8	Testing an Application Programming Interface	211

4.5 Important Terminology ..... 212

4.6 Practice Questions ..... 214

4.7 Practice Answers and Explanations ..... 219

4.8 Key Takeaways ..... 222

4.9 Summary ..... 223

**5 Integration Advisor** ..... 225

5.1 Objectives of This Portion of the Test ..... 227

5.2 Features ..... 227

5.3 How the Integration Advisor Fits into SAP Integration Suite ..... 229

5.3.1 Browsing the Library of Type Systems ..... 231

5.3.2 Creating Source and Target Message Implementation Guides ..... 233

5.3.3 Using Mapping Guidelines ..... 234

5.3.4 Getting the Proposal Service ..... 236

5.4 Message Implementation Guides ..... 238

5.4.1 Filtering Message Implementation Guides ..... 238

5.4.2 Creating Message Implementation Guides ..... 240

5.5 Mapping Guidelines ..... 261

5.5.1 Filtering Mapping Guidelines ..... 261

5.5.2 Creating and Editing Mapping Guidelines ..... 263

5.6 Mapping Proposals ..... 266

5.7 Generating and Consuming Integration Advisor Artifacts ..... 268

5.7.1 Creating Artifacts ..... 268

5.7.2 Consuming Mappings ..... 269

5.8 Important Terminology ..... 274

5.9 Practice Questions ..... 275

5.10 Practice Answers and Explanations ..... 280

5.11 Key Takeaways ..... 283

5.12 Summary ..... 284

**6 Trading Partner Management** ..... 285

6.1 Objectives of This Portion of the Test ..... 287

6.2 Features ..... 287

6.3 How Trading Partner Management Fits into SAP Integration Suite ..... 289

6.4 Navigating Business-to-Business Scenarios in the Web User Interface ..... 291

6.5 Designing Scenarios ..... 292

6.5.1 Company Profiles ..... 292

6.5.2 Trading Partners ..... 298

6.5.3 Agreement Templates ..... 307

6.5.4 Agreements ..... 310

6.6 Monitoring Scenarios ..... 316

6.7 Important Terminology ..... 317

6.8 Practice Questions ..... 319

6.9 Practice Answers and Explanations ..... 324

6.10 Key Takeaways ..... 326

6.11 Summary ..... 327

**7 Practice Exams** ..... 329

7.1 Exam 1 ..... 330

7.1.1 Exam 1 Practice Questions ..... 330

7.1.2 Exam 1 Practice Answers and Explanations ..... 347

7.2 Exam 2 ..... 357

7.2.1 Exam 2 Practice Questions ..... 357

7.2.2 Exam 2 Practice Answers and Explanations ..... 373

7.3 Exam 3 ..... 383

7.3.1 Exam 3 Practice Questions ..... 384

7.3.2 Exam 3 Practice Answers and Explanations ..... 401

The Author ..... 411

Index ..... 413

# Index

## A

ABAP RESTful application	
programming model	351
Access control	94
Access entity	95
Adapter	140, 181
types	141, 404
use	152
Advanced Encryption Standard (AES)	375
Agreement	310, 325
create	312
renewal and termination	312
structure	311
Agreement template	307, 318
create	308
AI-driven integration	77
AMQP connectivity test	164
Analytics and reporting	40, 288, 299
Apache Camel	352, 355, 380, 408
API	76, 192, 212
add policies	206
collaboration	83
create and bundle product	207
create from API provider	204
discovery	82
documentation	83
link external	354
monetization	83
policies	94
test	211
test console	99, 101, 113
testing	83
API key	109, 207, 211, 214, 220
API Management	52, 78, 80, 99, 111, 194, 196, 203, 220, 348, 380, 382
role collection	59
API POST request	407
API provider	103, 110, 202, 204, 349, 356, 377, 379, 402
API provisioning	76
API proxy	89, 102, 109–111
create	90
API resource	103
App router	374
Application	
create	209
Application development	41
Application key	88
Application management services	405

Application plane	41, 63
Application-to-application (A2A)	117
Approval workflow	311
Artifact	268
Artifacts tab	127
Assign message	95
Asynchronous notification	357
Authentication	196, 197, 213, 350, 401
Authorization	220

## B

Basic authentication	95, 355, 407
Bulk message	316
Business context	236, 274, 318
Business logic	377
Business partner	318
Business process integration	311
Business services	48
Business-to-business (B2B)	117
integrations	82
Business-to-business (B2B) scenarios	
design	292
monitor	316
navigate	291

## C

Cardinality	275
Centralized maintenance	307
Choose Best Proposal option	237
Cloud computing	
environment	378
principles	380
Cloud connector	378, 409
Cloud connector connectivity test	164, 188
Cloud Integration	52, 78, 80, 112, 116, 120, 187, 196, 325, 349, 351, 353, 375, 378, 380, 405, 407, 410
features	117
integration with Trading Partner	
Management	289
role collection	60
web UI	121
Cloud Platform Enterprise Agreement (CPEA)	46, 64
Code values	275
Codelist	275, 349
Company	292

Company profile ..... 292, 324, 326  
    *create* ..... 293  
Complex types ..... 274  
Compliance and risk management ..... 288  
Connectivity test ..... 162, 181, 188  
Connector ..... 192, 213, 220, 402  
Constraints ..... 356  
Consume artifact mapping ..... 269  
Consumer layer ..... 53  
Consumption analytics ..... 98  
Consumption-based model ..... 46  
Content modifier ..... 180, 187, 373  
Core data services (CDS) ..... 350  
CSV to XML converter ..... 156  
Custom tags ..... 125  
Customer relationship management  
    (CRM) applications ..... 192  
Customized partner agreements ..... 289

**D**

Data extraction ..... 62  
Data management ..... 40  
Data plane ..... 42  
Data privacy ..... 288  
Data store operations ..... 157  
Data types ..... 181  
Data/backend layer ..... 53  
DataStoreOperations ..... 403  
Design section ..... 122  
Developer perspective ..... 42  
DevOps ..... 382  
Discover section ..... 121  
Divide message ..... 349  
Documents tab ..... 128

**E**

EDI to EXML converter ..... 156  
Endpoint ..... 181  
Enterprise messaging ..... 79  
Enterprise Services Repository ..... 125  
Event ..... 142, 214  
Exam  
    *objective* ..... 21  
    *scoring* ..... 23  
    *structure* ..... 22  
Extension policies ..... 97  
External calls ..... 150  
Extract variables ..... 95

**F**

Flexibility ..... 288, 307  
Flexible and full-control  
    development ..... 42, 63  
Foundational plane ..... 42  
FTP connectivity test ..... 163

**G**

Gather messages ..... 149  
General Data Protection Regulation  
    (GDPR) ..... 197  
Groovy Script ..... 155, 177, 187, 347, 381

**H**

Handling exceptions ..... 158  
Header tab ..... 126  
Health Insurance Portability and  
    Accountability Act (HIPAA) ..... 197  
High-level coding model ..... 42  
Holistic integration ..... 77  
HTTP adapter ..... 349, 382  
HTTPS GET request ..... 354  
Hyperscalers ..... 353

**I**

Identifier ..... 318  
IDoc ..... 348, 377, 404  
iFlow ..... 130, 139, 180, 347, 349–354, 356,  
    374, 401  
    *components* ..... 139  
    *create* ..... 131, 151  
    *create purchase order* ..... 165  
    *customizable* ..... 121  
    *deploy* ..... 158  
    *editable* ..... 122  
    *enable trace logs* ..... 161  
    *purchase order notification* ..... 173  
    *upload* ..... 131  
IMAP connectivity test ..... 163  
Inbound direction ..... 317, 328  
Infrastructure as a service (IaaS) ..... 404  
Instance ..... 213, 220, 221  
    *create* ..... 199  
    *token* ..... 221  
Integration adapter ..... 136  
Integration Advisor ..... 52, 78, 80, 226, 374,  
    375, 383, 402, 408, 410  
    *features* ..... 227  
    *role collection* ..... 60  
    *type systems* ..... 231

Integration artifact ..... 130, 180, 186  
Integration assessment ..... 60  
Integration package ..... 126, 180  
    *create* ..... 126  
    *delete* ..... 129  
    *export* ..... 129  
Integration platform as a service (iPaaS) ..... 78  
Integration runtime ..... 188  
Intelligent technologies ..... 41  
Interchange ..... 318  
Internet of Things (IoT) ..... 352  
Invalidate cache ..... 95

**J**

Java ..... 375  
JavaScript ..... 97, 187, 381  
    *mappings* ..... 155  
JavaScript Object Notation (JSON) ..... 192, 407  
JMS ..... 377  
Join messages ..... 148  
JSON threat protection ..... 96  
JSON to XML ..... 95  
JSON to XML converter ..... 156

**K**

KAFKA connectivity test ..... 164  
Key value maps ..... 111  
Key value operations ..... 95

**L**

Local integration ..... 376  
Log level ..... 402  
Lookup cache ..... 94  
Low-code/no-code model ..... 42, 63  
Low-code/no-code programming ..... 401

**M**

Malware scanner ..... 125  
Manage integration content ..... 159  
Manage user request access ..... 84  
Mapping ..... 145, 318  
    *artifacts* ..... 354  
Mapping guideline (MAG) ..... 231, 261,  
    274, 383  
    *create* ..... 263  
    *filter* ..... 261  
    *use* ..... 234  
Mapping proposal ..... 235, 266, 282  
Marketo ..... 192, 220  
Mediation policies ..... 95

Message ..... 274  
Message context ..... 153  
Message conversion ..... 155, 403  
Message implementation guideline  
    (MIG) ..... 231, 274, 280, 354, 381, 401, 406  
    *create* ..... 240  
    *create source* ..... 240  
    *enhance source* ..... 247  
    *enhance target* ..... 255  
    *filter* ..... 238  
    *organizational structure* ..... 233  
    *target* ..... 233  
Message logging ..... 97  
Message mapping ..... 138, 188, 355  
Message mapping transformer ..... 382  
Message processing flaws ..... 378  
Message processing monitor ..... 404  
Message routing ..... 147  
Message transformer ..... 144  
Message validation ..... 97  
Microservices ..... 348  
Migration assessment ..... 62, 72  
    *role collection* ..... 61  
Monitor section ..... 123  
Monitoring and compliance ..... 311  
Multicast ..... 147

**N**

NetSuite ..... 220  
Node.js ..... 375

**O**

OAuth ..... 96, 110–112, 196, 213, 219, 221  
OData ..... 378, 379, 383  
    *API* ..... 118, 135  
Omnichannel experiences ..... 81  
Open Connectors ..... 52, 79, 80, 192, 350, 351,  
    355, 381, 406  
    *create API provider* ..... 202  
    *features* ..... 194  
    *log on* ..... 198  
    *query language* ..... 195  
    *role collection* ..... 60  
    *usage* ..... 198  
Open integration ..... 77  
Open PGP ..... 356  
openSAP courses ..... 32  
Operations ..... 103  
Organization secret ..... 202, 221  
Outbound direction ..... 317  
Out-of-the-box integration ..... 77, 406  
Overview tab ..... 126



P

Partner data management ..... 287

Partner determination ..... 298

Partner function ..... 298

Partner profile ..... 299

*integration with Trading Partner Management* ..... 307

Partner role ..... 299

Pay-As-You-Go for SAP BTP model ..... 46, 63, 71

Payload ..... 192, 214, 401

Platform as a service (Paas) ..... 382, 409

Policy ..... 103

POP3 connectivity test ..... 164

Populate cache ..... 94

Postflow ..... 103

Preflow ..... 103, 110

Pricing ..... 49

Private link service ..... 409

Product ..... 109, 379

*create* ..... 92

*profile* ..... 124

Proposal ..... 275

Proposal service ..... 236

Provisioning capabilities ..... 55

Proxy endpoint ..... 89, 103

Purchase order API

*consume* ..... 169

*explore* ..... 167

Q

Qualifier ..... 275

Questionmark Secure browser ..... 24

QuickBooks ..... 192

Quota ..... 94

R

Raise fault ..... 95

Read-only access ..... 374

Real world scenario

*Cloud Integration* ..... 116

*Integration Advisor* ..... 226

*Open Connectors* ..... 193

*SAP Integration Suite* ..... 38

*Trading Partner Management* ..... 286

Real-time collaboration ..... 288

Real-world scenario

*API Management* ..... 76

Receiver ..... 140, 326

Receiver server certificate ..... 356

Regular expression protection ..... 96

Related missions ..... 50

Request quota ..... 94

Request reply ..... 380

Response cache ..... 94

REST API ..... 134, 350

RESTful API ..... 192

Role collection ..... 59

Role-based authorization ..... 407

Router ..... 378, 407

Runtime artifact ..... 231, 281

S

Salesforce ..... 192, 220

SAML assertion generation ..... 96

SAML assertion validation ..... 97

SAP Analytics Cloud ..... 408

SAP Application Interface Framework ..... 186

SAP Ariba extensions ..... 376

SAP BTP, ABAP environment ..... 44, 374

SAP BTP, Cloud Foundry environment ..... 44, 64, 374, 382, 410

SAP BTP, Kyma runtime ..... 44, 64, 374

SAP BTP, Neo environment ..... 45

SAP BTP, serverless runtime ..... 79

SAP Build Process Automation ..... 409

SAP Business Accelerator Hub ..... 53, 70, 78, 82, 104, 187, 350, 353, 354, 375, 377, 405

*create application* ..... 87

*purchase order API* ..... 167

*roles* ..... 98

SAP Business Technology Platform

    (SAP BTP) ..... 38, 63, 351–353, 355, 377, 379, 381, 406

*architecture* ..... 41

*capabilities* ..... 40

*commercial models* ..... 45

*create trial account* ..... 54

*programming environments* ..... 43

*technical service planes* ..... 41

*trial offerings* ..... 50

SAP Cloud ..... 356, 409

SAP Cloud Application Programming

    Model ..... 349, 353

SAP Data Intelligence ..... 79

SAP Data Quality Management ..... 403

SAP Discovery Center ..... 47, 72, 378

*business services* ..... 48

SAP Integration Suite .... 38, 196, 347, 355, 376, 378, 381

*API management* ..... 81

*Cloud Integration* ..... 119

*core capabilities* ..... 52

*features* ..... 77

SAP Integration Suite (Cont.)

*Integration Advisor* ..... 229

*Open Connectors* ..... 197

*resources* ..... 29

*subscribe* ..... 54

*Trading Partner Management* ..... 289

SAP Intelligent Enterprise

    Framework ..... 350, 351

SAP Learning Hub ..... 31

SAP Learning Journeys ..... 31

SAP NetWeaver Developer Studio ..... 374

SAP OData Provisioning ..... 377

SAP Process Orchestration ..... 62, 118

SAP Remote Access and Connectivity ..... 378

SAP S/4HANA ..... 348, 351

SAP S/4HANA Cloud ..... 352, 353, 408

*OData system* ..... 231

SAP Smart Business ..... 354

SAP training course CLD900 ..... 30

SAP Web IDE ..... 112, 353

Scenario evaluation ..... 62

Script collection ..... 137

*build* ..... 177

Security ..... 95, 288, 347

Sender ..... 140, 326

Service callout ..... 97

Settings ..... 124

Simple types ..... 274

SMTP connectivity test ..... 163

SOAP ..... 354, 402, 408

SOAP adapter ..... 355

SOAP API ..... 132

Spike arrest ..... 94

Splitter ..... 149

SSH connectivity test ..... 163

SSL ..... 348

Standard API layer ..... 53

Standardization ..... 307

Statistics collector ..... 97

String manipulation ..... 381

Subscriptions-based model ..... 45, 64

T

Tags tab ..... 128

Target endpoint ..... 89, 103

Template hierarchy ..... 307

Tenant administration ..... 401

Three-layer architecture ..... 52

TLS connectivity test ..... 163

Trading partner ..... 298, 317, 324

*create* ..... 299

*profile* ..... 318

Trading partner agreement ..... 318

*template* ..... 288

Trading Partner Management ..... 286, 324

*B2B scenarios* ..... 291

*features* ..... 287

*role collection* ..... 61

Transport ..... 125

Twelve factors ..... 376

Twitter instance ..... 199, 222, 404

Type system ..... 274

Type system library ..... 231

U

User secret ..... 202, 221

V

Value mappings ..... 131

Verify API key ..... 95

Verify JWT ..... 96

Version control ..... 308

Versioning ..... 311

W

Web Services Security (WS Security) ..... 356

Workflow management ..... 288

X

X.509 certificate ..... 378

XML ..... 154, 378, 407

XML threat protection ..... 96

XML to CSV converter ..... 156

XML to EDI converter ..... 156

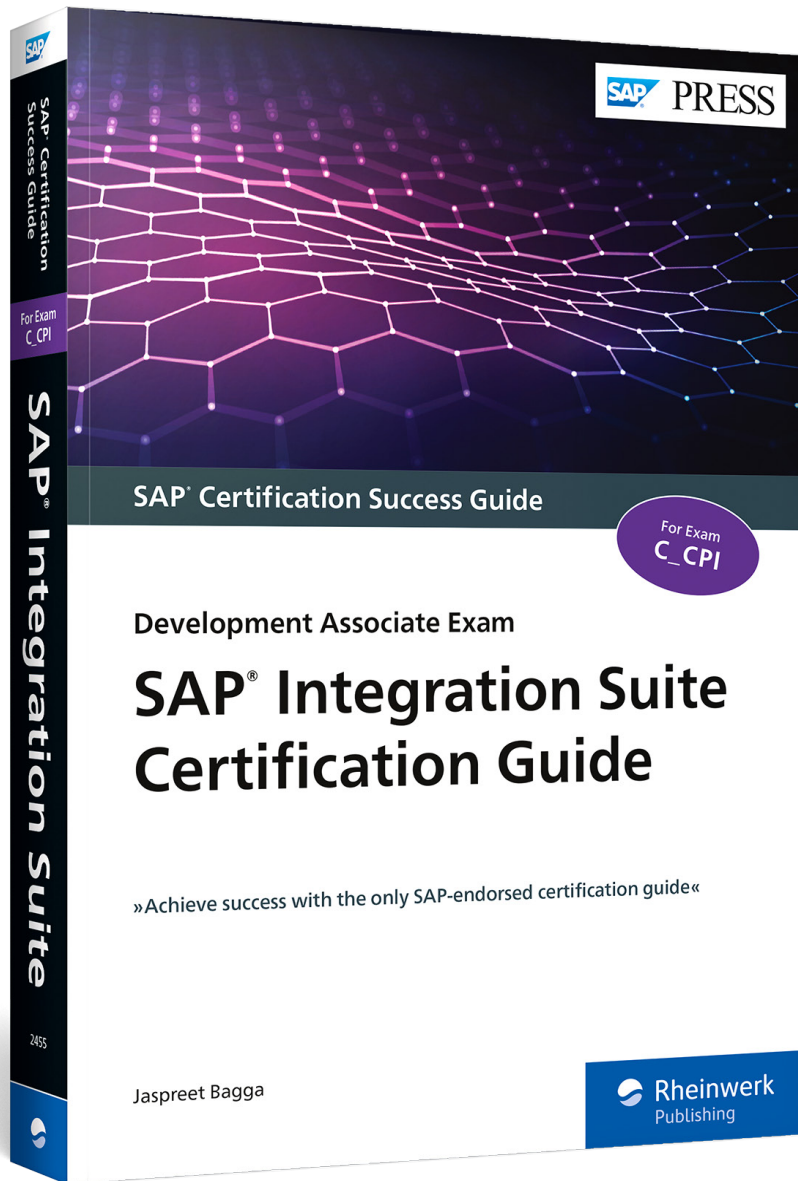
XML to JSON ..... 95

XML to JSON converter ..... 156

XSL transform ..... 95

XSLT ..... 154

XSLT mapping ..... 270, 374



**Jaspreet Bagga** is an executive consultant and SAP integration subject matter expert with more than 200 successful projects delivered to Fortune 500 companies. Jaspreet is experienced in SAP integration project delivery, team management, IT transformation strategy, go-to-market strategy, SAP package selection, program budget and estimates, change control, business migrations, and managing stakeholder expectations. He has hands-on SAP skills in solution architecture, development, leading the delivery of complex integration programs, managing dispersed teams, and ensuring successful project go-live/goals.

Jaspreet Bagga

## SAP® Integration Suite Certification Guide

### Development Associate Exam

417 pages | 08/2023 | \$79.95 | ISBN 978-1-4932-2455-5

 [www.sap-press.com/5735](https://www.sap-press.com/5735)

*We hope you have enjoyed this reading sample. You may recommend or pass it on to others, but only in its entirety, including all pages. This reading sample and all its parts are protected by copyright law. All usage and exploitation rights are reserved by the author and the publisher.*