✓ 100 XP

Exercise - Create and call a custom connector from a Logic Apps workflow

10 minutes

Sandbox activated! Time remaining: 3 hr 4 min

You have used 4 of 10 sandboxes for today. More sandboxes will be available tomorrow.

You have a Web API called Print Framer API that calculates a cost for a picture frame based on the dimensions you send to it. Developers throughout your company want to use this in their Logic Apps, but there is currently no way to make that connection.

In this exercise, you'll create a custom connector for the Print Framer API and use it to send and receive data from a Logic App to a Web API. First, we need a Logic App.

Create a Logic App

1. In the Azure portal menu or from the **Home** page, select **Create a resource**.
2. Select **Web** > **Logic App**.
3. In the **Logic App** window, enter the following settings, and then select **Create**:

Setting	Value
Name	Choose a unique name. Make a note of it, you'll need it later on.
Subscription	*Concierge Subscription*
Resource group	Select *Use existing* and choose *learn-0678b581-92a9-419c-8382-a464d7819f5a*
Location	Select a region near you, that is also in the sandbox region listed below.
Log Analytics	Off

Setting	Value

The free sandbox allows you to create resources in a subset of the Azure global regions. Select a region from the following list when you create resources:

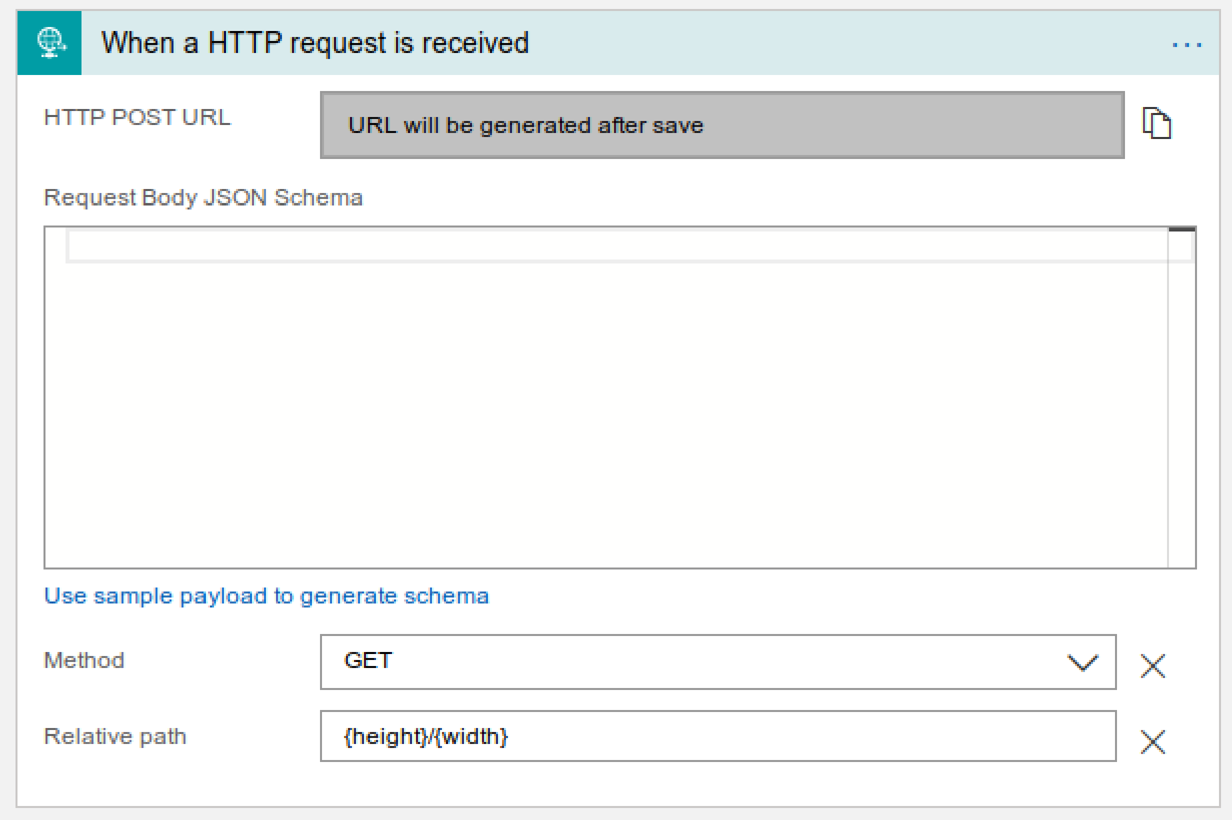
- West US 2
- South Central US
- Central US
- East US
- West Europe
- Southeast Asia
- Japan East
- Brazil South
- Australia Southeast
- Central India

4. When the Logic App has been created, from the Azure portal menu, select **All resources** and then select the Logic App. The Azure portal opens the designer for the Logic App.

5. Under **Start with a common trigger**, select **When an HTTP request is received**.

6. Select **Add a new parameter**, then **Method**, and then in the **Method** drop-down list, select **GET**.

7. Once again, select **Add a new parameter**, then **Relative path**, and then in the **Relative path** field, type `{height}/{width}`.


When a HTTP request is received
...
HTTP POST URL
URL will be generated after save
Request Body JSON Schema
Use sample payload to generate schema

Method: GET

Relative path: {height}/{width}

8. In the **Logic Apps Designer**, select **Save**.

We now have a basic Logic App. Let's add a custom connector so that we can call our custom PrintFramer API from our workflow.

Create a new custom Logic Apps connector in the Azure portal

1. In the Azure portal menu or from the **Home** page, select **Create a resource**.
2. In the **Search the marketplace** textbox, type **Logic Apps Custom Connector**, select **Logic Apps Custom Connector**, and then **Create**.
3. In the **Logic App Custom Connector** window, enter the following settings, and then select **Create**:

Setting	Value
Name	PrintFramerConnector
Subscription	*Concierge Subscription*
Resource group	Select *Use existing* and choose *learn-0678b581-92a9-419c-8382-a464d7819f5a*

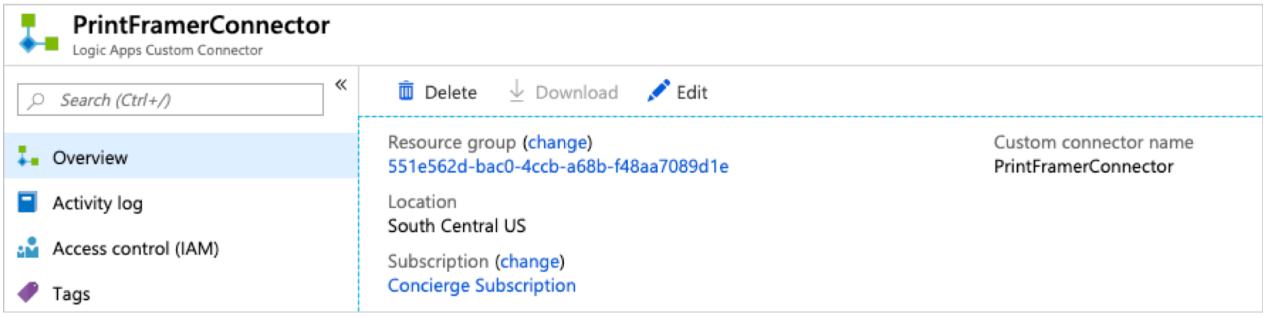
Setting	Value
Location	Select a region near you, that is also in the list of regions supported by the sandbox.

Import the OpenAPI definition

Now let's use the OpenAPI file we saved early to define custom connector:

1. In the Azure portal or from the **Home** page, select **All resources** and then select **PrintFramerConnector**.

2. On the **Overview** page, select **Edit**.



Resource group (change)
551e562d-bac0-4ccb-a68b-f48aa7089d1e

Location
South Central US

Subscription (change)
Concierge Subscription

Custom connector name
PrintFramerConnector

3. In the **Custom connectors** section, select **OpenAPI file** and then **Import**. Open the JSON file you saved in the last exercise.

Configure the custom connector

The OpenAPI file tells the custom connector about the operations that are available in the Web API. There are a few other properties we must set for the custom connector:

1. In the **General Information** section, in the **Description** box, type **Obtain an estimate for a picture frame**. You might have to scroll down to see this part of the configuration page.

2. In the **Host** textbox, type the URL you noted earlier.

3. On the **Security** page, ensure the **No Authentication** is selected.

> ⓘ Important
>
> For this demonstration, it's OK to allow access to the API without any authentication. For real-world APIs, which contain sensitive data or require a paid subscription,

authentication is essential.

4. On the **Definition** page, in the **Summary** textbox, type **Estimate the cost of a picture frame.**.

5. Under **Visibility**, select **Important**.

6. At the top of the page, select **Update connector**.

Add the custom connector to our Logic App

Now, you can add the custom connector to our Logic App.

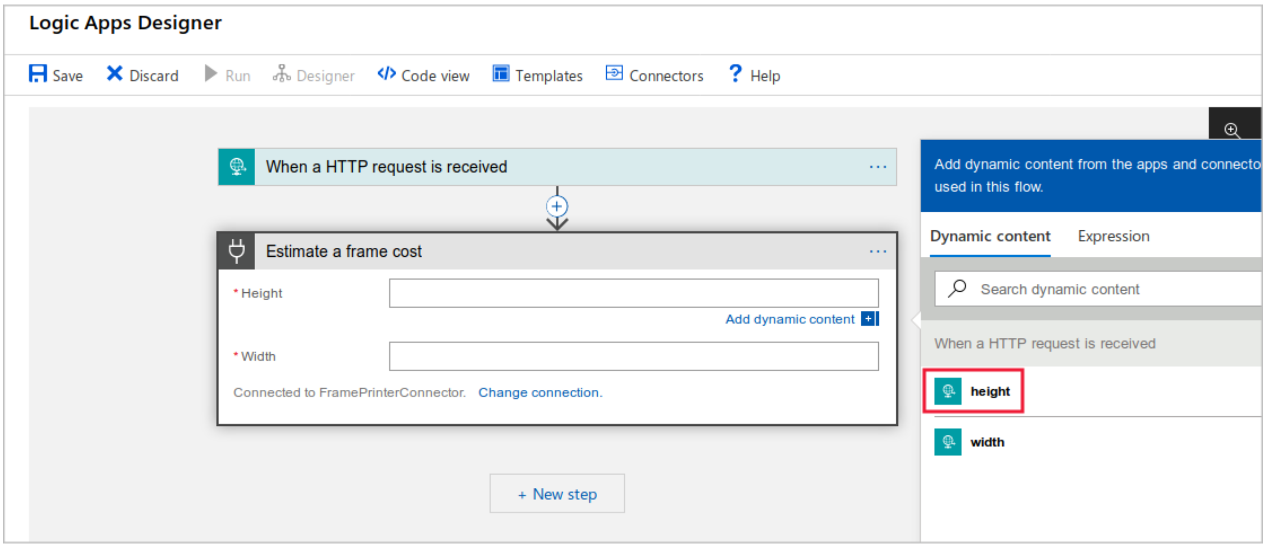
1. In the Azure portal or from the **Home** page, select **All resources** and then click the Logic App that you created in unit 3.

2. In the Logic App resource page, select **Edit** and then select **+ New Step**.

3. In the **Choose an action** box, select the **Custom** tab, and then select **PrintFramerConnector**.

4. Under **Actions** select **Estimate the cost of a picture frame**.

5. In the **Estimate the cost of a picture frame** box, select the **Height** field, and then, select **height** from the **Dynamic content** panel on the right. The following screenshot illustrates this selection.



Logic Apps Designer

* Height

Add dynamic content

* Width

Connected to FramePrinterConnector. Change connection.

Add dynamic content from the apps and connecto used in this flow.

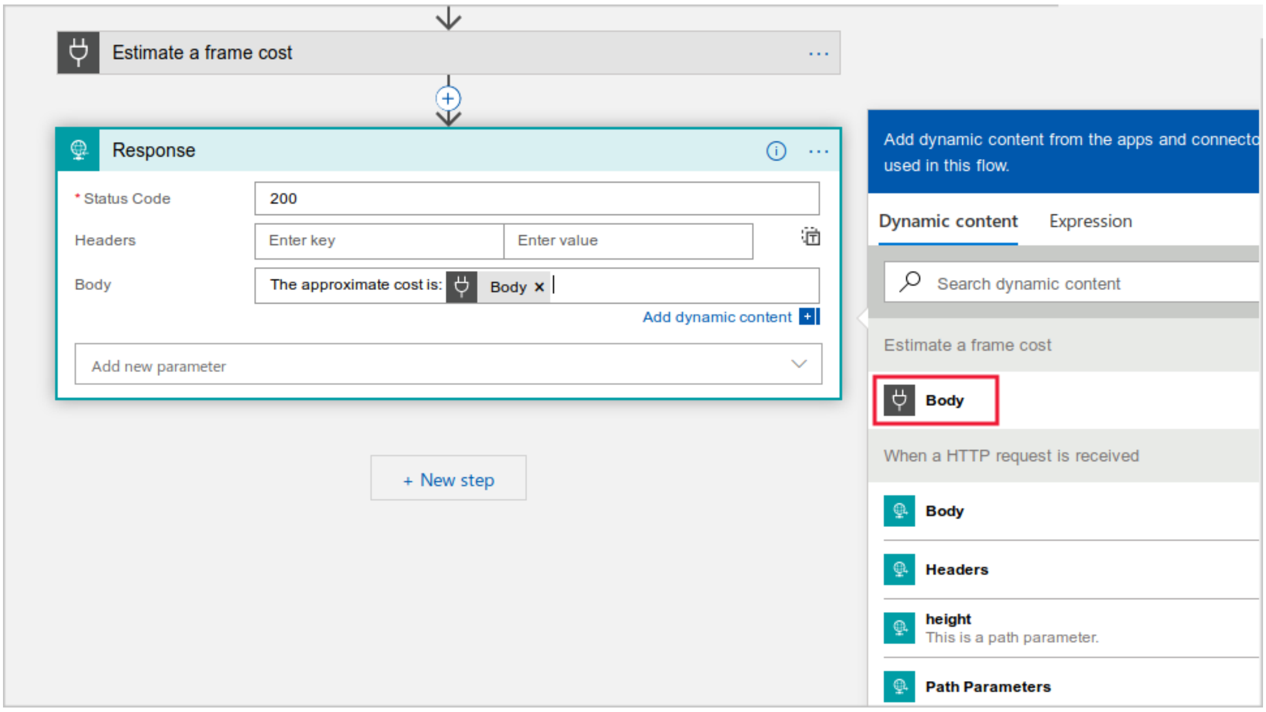
6. Select **Width** field, and then, under **Dynamic content** select **width**.

The height and width parameters that we receive from each HTTP request that triggers our Logic App are now bound to the height and width parameters we pass to the PrintFramerConnector.

Complete the Logic App

The final thing to add to the Logic App is something to display the results. Let's use an HTTP response to show the estimate to the user:

1. In the Logic App designer, click **+ New Step**.
2. In the **Search connectors and actions** textbox, type **Response**, and then select the **Response** action.
3. In the **Body** textbox, type "**The approximate cost of your frame is:$**".
4. In the **Dynamic content** pane to the right, select **Body** from the **Estimate the cost of a picture frame** section. This selection is illustrated in the following screenshot.



Status Code: 200

Body: The approximate cost is: Body

5. Save your changes by selecting **Save** at the top of the Logic Apps designer.

Test the Logic App

The Logic App is now complete, and will call the Web API through the custom connector. All you have to do is make a request to the Logic App, by using the browser:

1. In the Logic Apps designer, expand the **When an HTTP request is received** box.
2. To the right of the **HTTP GET URL** textbox, click the **Copy URL** button.
3. Open a new browser tab and paste the **URL** into the address bar.
4. In the pasted URL, replace the **{height}** parameter with **10**, and the **{width}** parameter with **4** and then press Enter. The page displays an approximate cost.
5. Try entering other heights and widths.

Congratulations! You've created a Logic Apps workflow that can call your in-hour PrintFramerAPi through a custom connector!

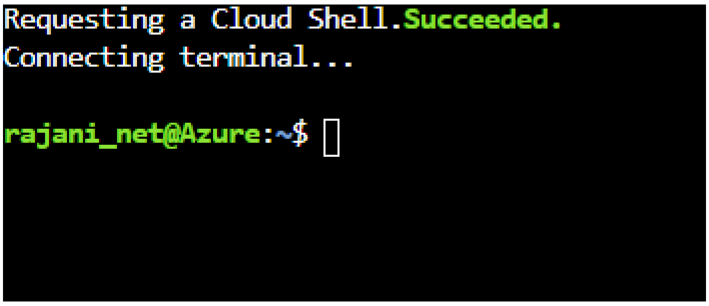
Next unit: Share a custom connector

Continue >

🌐 English (United States) ☼ Theme

Previous Version Docs Blog Contribute Privacy & Cookies Terms of Use Trademarks

Azure Cloud Shell



```
Requesting a Cloud Shell.Succeeded.
Connecting terminal...

rajani_net@Azure:~$ 
```