

Conector personalizado en PowerApps

Los conectores en Power Apps proporcionan una forma de establecer una conexión a una fuente de datos y nos permiten usar estas fuentes de datos en nuestras aplicaciones. Estas fuentes de datos pueden ser SharePoint, SQL Server, Outlook, Dynamics 365 y muchas más. En la actualidad, PowerApps proporciona más de 300 conectores que ya están disponibles para usar en nuestras aplicaciones.

Pero, ¿y si el conector que queremos usar no está disponible o si queremos usar nuestro propio conector en las aplicaciones? En este tipo de escenario, los conectores personalizados entran en escena. Los conectores personalizados nos permiten usar nuestros propios conectores en las aplicaciones.

Introducción a los *OpenWeatherMap*

OpenWeatherMap se utiliza para proporcionar un informe de pronóstico del tiempo con la ayuda de una API. En este blog, estamos usando **OpenWeatherMap** para proporcionar un informe meteorológico en tiempo real en nuestra aplicación de lienzo.

Registrarse en <https://openweathermap.org/>

Buscar *OpenWeatherMap* API Key

Para obtener una clave de API, debe crear una **cuenta gratis**. Después de crear una cuenta, se le proporcionará una clave de API predeterminada.

Puede usar el valor predeterminado **Clave API** o puede generar su clave API. Esta clave API actuará como su **la contraseña** que se utiliza para la autorización y evita que otros usuarios utilicen su API.

Prueba de puntos finales de API con una clave de API

Dado que obtuvimos una clave API, probaremos nuestros puntos finales utilizando una clave API. Queremos mostrar informes meteorológicos por nombre de ciudad, por lo que utilizaremos puntos finales que tengan el nombre de la ciudad como parámetro de consulta. Siga los pasos que se describen a continuación:

OpenWeather

Weather in your city

Guide API Dashboard Pricing Maps Our Initiatives Partners Blog Marketplace rohit ... Support

New Products Services API keys Billing plans Payments Block logs My orders My profile Ask a question

You can generate as many API keys as needed for your subscription. We accumulate the total load from all of them.

Key	Name	Status	Actions
6379578d19cdc764734e80ee6708d448	app	Active	Copy Edit

Create key

API key name [Generate](#)

Product Collections

- Current and Forecast APIs**
- Historical Weather Data
- Weather Maps
- Weather Dashboard
- Widgets

Subscription

- How to start
- Pricing
- Subscribe for free
- FAQ

About us

OpenWeather is a team of IT experts and data scientists that has been practising deep weather data science since 2014. For each point on the globe, OpenWeather provides historical, current and forecasted weather data via light-speed APIs. Headquarters in London, UK.

Current Weather Data

[API doc](#) [Subscribe](#)

- Access current weather data for any location including over 200,000 cities
- We collect and process weather data from different sources such as global and local weather models, satellites, radars and a vast network of weather stations
- JSON, XML, and HTML formats
- Included in both free and paid subscriptions

Hourly Forecast 4 days

[API doc](#) [Subscribe](#)

- Hourly forecast is available for 4 days
- Forecast weather data for 96 timestamps
- JSON and XML formats
- Included in the Developer, Professional and Enterprise subscription plans

One Call API

[API doc](#) [Subscribe](#)

- Make one API call and get current, forecast and historical weather data
- Minute forecast** for 1 hour
- Hourly forecast** for 48 hours
- Daily forecast** for 7 days
- Historical data** for 5 previous days
- National weather alerts**
- JSON format
- Included in both free and paid subscriptions

Please note that built-in API requests by city name, zip-codes and city id will be deprecated soon.

Built-in API request by city name

You can call by city name or city name, state code and country code. Please note that searching by states available only for the USA locations.

API call

```
api.openweathermap.org/data/2.5/weather?q={city name}&appid={API key}
```

```
api.openweathermap.org/data/2.5/weather?q={city name},{state code}&appid={API key}
```

```
api.openweathermap.org/data/2.5/weather?q={city name},{state code},{country code}&appid={API key}
```

Call current weather data

- How to make an API call
- Bulk downloading
- Weather fields in API response
- JSON
- XML
- List of condition codes
- Min/max temperature in current weather
- API and forecast API
- Other features
- Geocoding API
- Built-in geocoding
- Built-in API request by city name
- Built-in API request by city ID
- Built-in API request by ZIP code
- Format
- Units of measurement
- Multilingual support
- Call back function for JavaScript code

Ahora ve al navegador y prueba tu **punto final**. Solo proporcione su **Nombre de la ciudad** (q) y su **Clave API** (ID de la aplicación).

```
// https://api.openweathermap.org/data/2.5/weather?q=mumbai&appid=6379578d19cdc764734e80ee6708d448
```

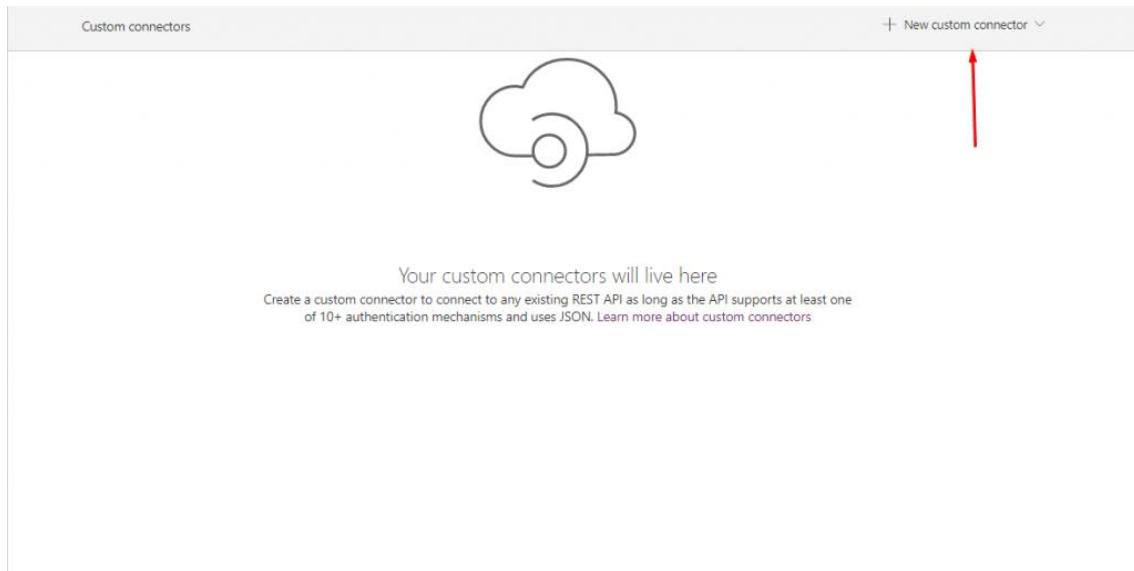
Verá la respuesta JSON si tiene éxito como se muestra a continuación. Necesitará su clave API y esta respuesta JSON más adelante, así que manténgala abierta en el navegador.

Creación de un nuevo conector personalizado

Ahora que hemos obtenido una clave API y la hemos probado con éxito, podemos continuar para crear nuestro conector personalizado. Cree una cuenta o inicie sesión en las aplicaciones de energía.

Seleccione Dataverse y haga clic en conectores personalizados.

A continuación, en la parte superior derecha, haga clic en Nuevo conector personalizado.



Ahora, verá que hay múltiples opciones para elegir. En este blog, vamos a crear desde cero.

A continuación, proporcione un nombre significativo para su conector. Puedes nombrar lo que quieras.

Llenar la pestaña de información general

En la sección General, puede cargar el icono, aplicar el color de fondo al icono y dar una descripción de su conector.

Seleccione el esquema como HTTPS ya que nuestro punto final es de https.

The screenshot shows the 'Custom connector' configuration window. The 'General information' tab is active. It includes fields for 'Icon background color' (set to #0000FF), 'Description' (set to 'How to use custom connector in powerapps'), and 'Scheme' (set to HTTPS). A red arrow points to the 'Scheme' dropdown.

tipo de esquema

// https //api.openweathermap.org/data/2.5/weather?q=mumbai&appid=6379578d19cdc764734e80ee6708d448

Nombre de host

// https://api.openweathermap.org/data/2.5/weather?q=mumbai&appid=6379578d19cdc764734e80ee6708d448

Definición de secciones de seguridad

En la sección de seguridad, seleccione la clave API como tipo de autenticación, ya que estamos usando la clave API para acceder a nuestro punto final.

The screenshot shows the 'Custom connector' configuration window with the 'Security' tab active. The 'Authentication type' dropdown is open, and 'API Key' is selected. A red arrow points to the 'API Key' option.

A continuación, complete la etiqueta del parámetro como Clave y el nombre del parámetro como autorización.

Seleccione **Encabezamiento** como tu **Parámetro** ubicación.

The screenshot shows the 'API Key' configuration dialog. It has a purple header with the text 'API Key'. Below the header, it says 'Users will be required to provide the API Key when creating a connection'. There are three input fields: 'Parameter label' with the value 'Key', 'Parameter name' with the value 'Authorization', and 'Parameter location' with a dropdown menu set to 'Header'. Red arrows point to the 'Key' and 'Authorization' fields. At the bottom left is an 'Edit' button with a pencil icon. At the bottom right, there are two tabs: 'General' and 'Definition', with 'Definition' being the active tab and highlighted with a red box.

Definición de una acción para nuestro conector personalizado

En la sección de definición, podemos definir más de una acción para nuestro conector personalizado. En este blog, solo tendremos una acción, pero si lo desea, puede agregar más acciones más adelante para diferentes puntos finales. Haga clic en la acción Nueva y en la sección general complete el resumen y la descripción.

En el campo ID de operación, asigne un nombre al campo que sea fácil de recordar y significativo, ya que este campo se requerirá más adelante en la aplicación de lienzo.

The screenshot shows the '3. Definition' tab in the Swagger Editor. The top bar includes 'Connector Name' set to 'Custom connector', a 'Swagger Editor' toggle, and buttons for 'Create connector' and 'Cancel'. The main area is divided into three sections: 'Actions (0)', 'References (0)', and 'Policies (0)'. Each section has a description and a '+ New action', '+ New reference', or '+ New policy' button. A red arrow points to the '+ New action' button. To the right of the sections is a 'Start by adding an action or trigger on the left.' instruction, a 'Security' tab, and a 'Code (Preview)' button.

3. Definition Swagger Editor Create connector Cancel

Actions (1)

Actions determine the operations that users can perform. Actions can be used to read, create, update or delete resources in the underlying connector.

1 place ...

+ New action

References (0)

References are reusable parameters used by both actions and triggers.

Policies (0)

Policies are used to change the behavior of actions and triggers.

General

Summary Learn more

GetWeatherReport

Description Learn more

Allow us to get weather reports by city

Operation ID *

This is the unique string used to identify the operation.

place

Visibility Learn more

☐ none
☐ advanced
☐ internal
☒ important

Importar desde muestra

Aquí, definiremos el requisito necesario para la solicitud.

Usaremos GET ya que estamos obteniendo datos de la API.

En la URL, proporcione el punto final que usamos anteriormente para probar por nombre de ciudad y haga clic en importar.

Request

It defines the pre-requirements needed in order to make a request. Describes a single operation parameter. A unique parameter is defined by a combination of a name and location.

+ Import from sample

Response

It defines the shape of response returned by the underlying connector when making the request.

default default

+ Add default response

Import from sample

Verb *

☒ GET ☐ DELETE ☐ POST ☐ PUT ☐ HEAD ☐ OPTIONS

☐ PATCH

URL *

This is the request URL.

Headers

Headers separated by a new line, e.g.:
Content-Type application/json
Accept application/json

These are custom headers that are part of the request.

ENG 7:04 PM

Definición de una respuesta en nuestro conector personalizado

A continuación, completaremos la sección de respuesta, simplemente haga clic en Agregar respuesta predeterminada.

Copie la respuesta que obtuvimos anteriormente del punto final que probamos y péguela en la sección del cuerpo y haga clic en importar.

Response

It defines the shape of response returned by the underlying connector when making the request.

default default

+ Add default response

Import from sample

Headers

Headers separated by a new line, e.g.:
Content-Type application/json
Accept application/json

These are custom headers that are part of the response.

Body

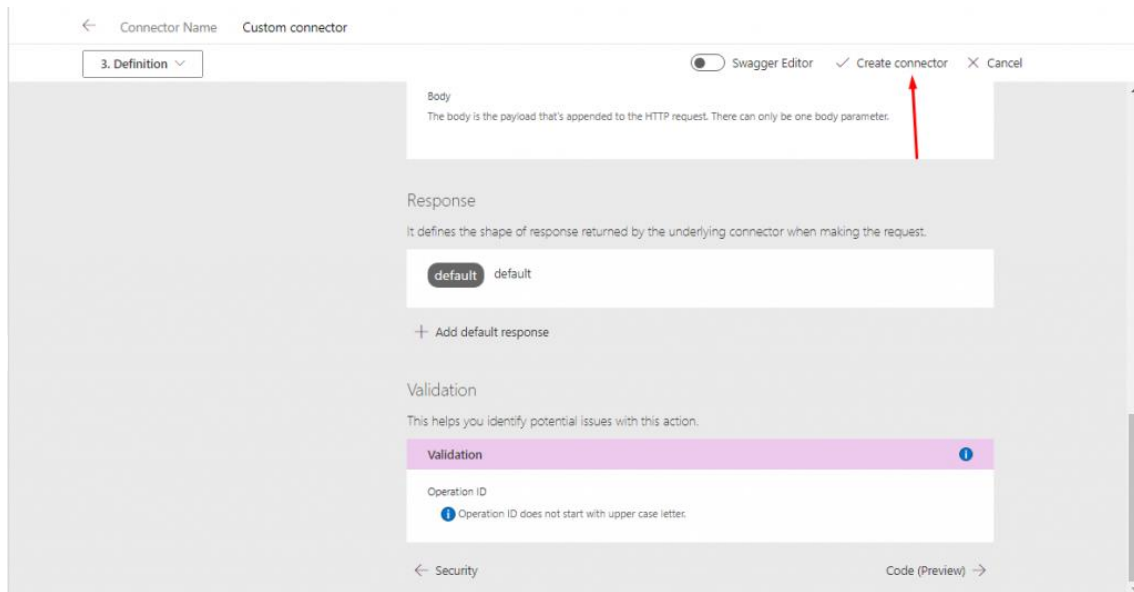
```
{  
  "dt": 1645967295,  
  "sys": {  
    "country": "IN",  
    "sunrise": 1645924609,  
    "sunset": 1645965886  
  },  
  "timezone": 19800,  
  "id": 1273313,  
  "name": "Dehradun",  
  "cod": 200  
}
```

The payload that is available on the response. These are the tokens that will show up as the outputs in designer.

Import

Close

Ahora, haga clic en **Crear conector**.

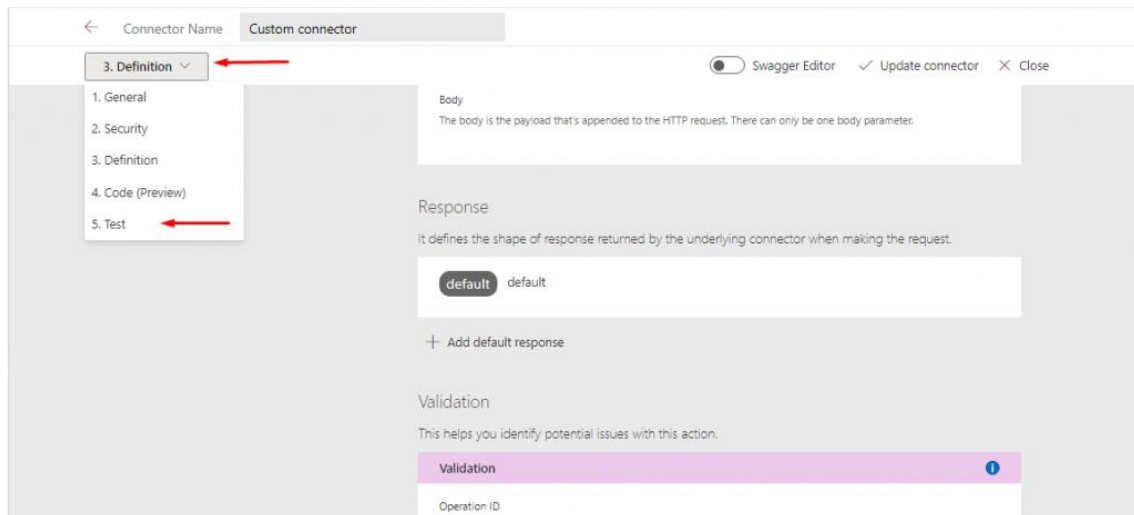


Ahora, verá el mensaje de que nuestro conector personalizado se ha creado correctamente.



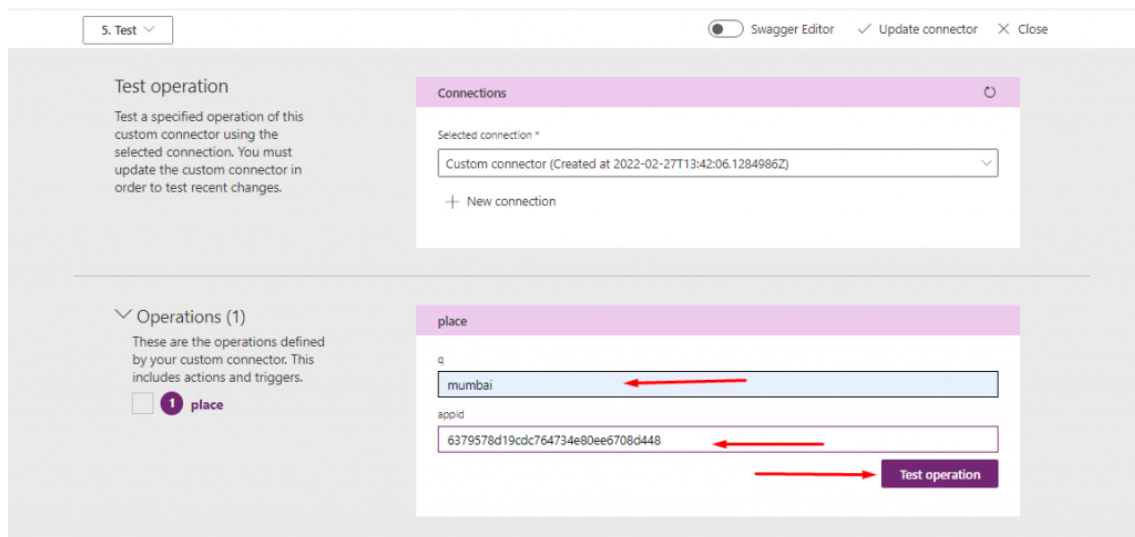
Probando nuestro conector personalizado

Seleccione Prueba de la lista desplegable

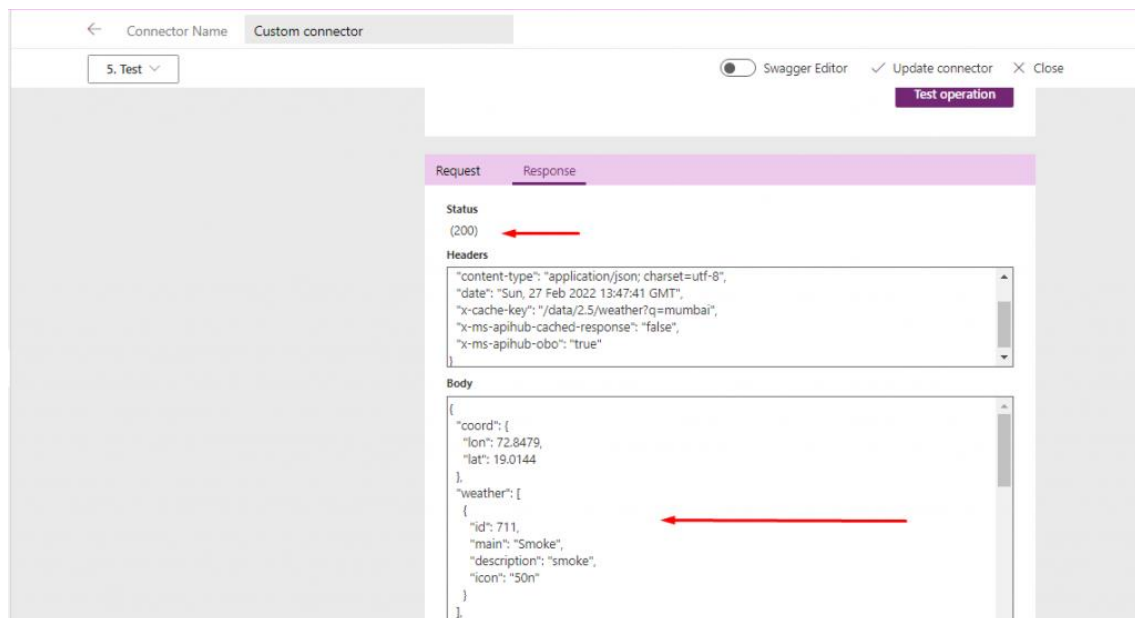


A continuación, proporcione su **API Key** y haga clic en **Para crear**.

En la sección del lugar, seleccione Mumbai o cualquier ciudad que desee en el parámetro de consulta (q) y proporcione la clave de identificación de su aplicación y haga clic en Probar operación.

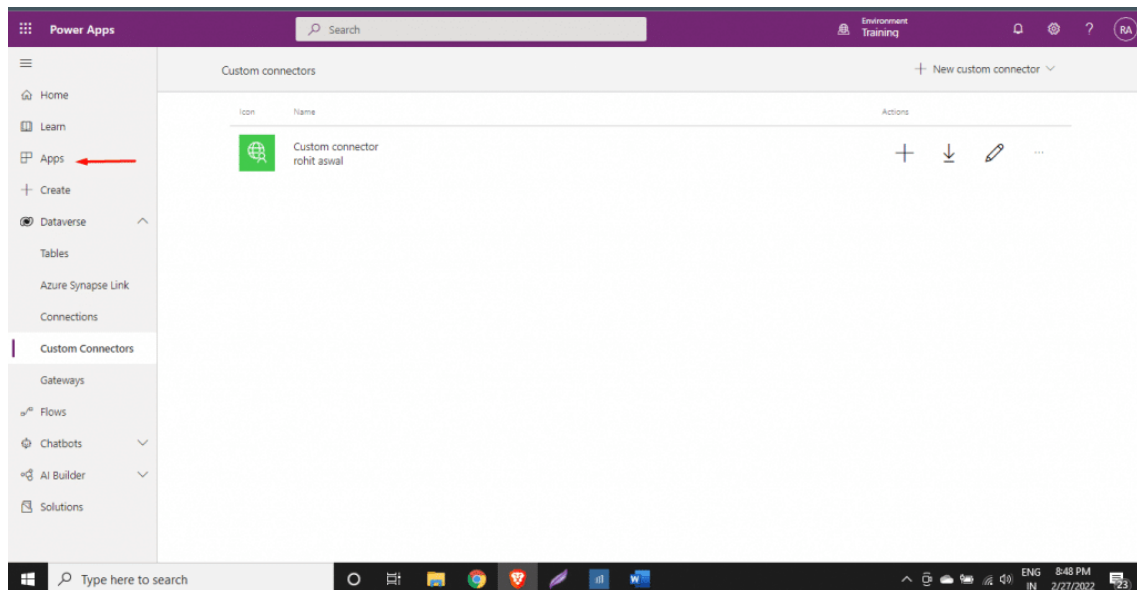


Si la solicitud es exitosa, devolverá una respuesta con un código de estado de 200 y una respuesta JSON en una sección del cuerpo como se muestra a continuación.

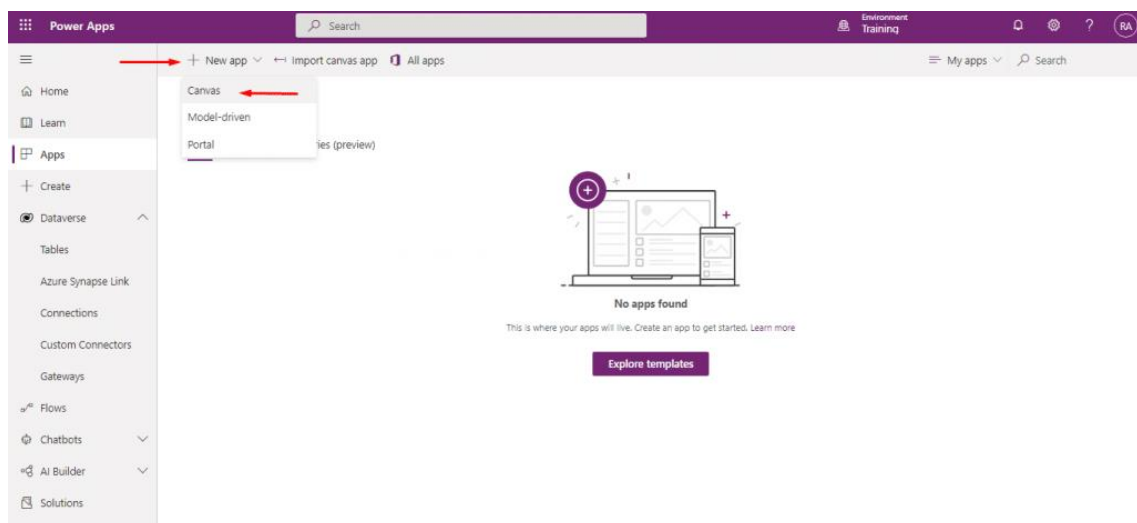


Usando nuestro conector personalizado en la aplicación de lienzo

Después de crear nuestro conector personalizado, ahora veremos cómo agregar nuestro conector personalizado a nuestra aplicación de lienzo y usarlo como fuente de datos.

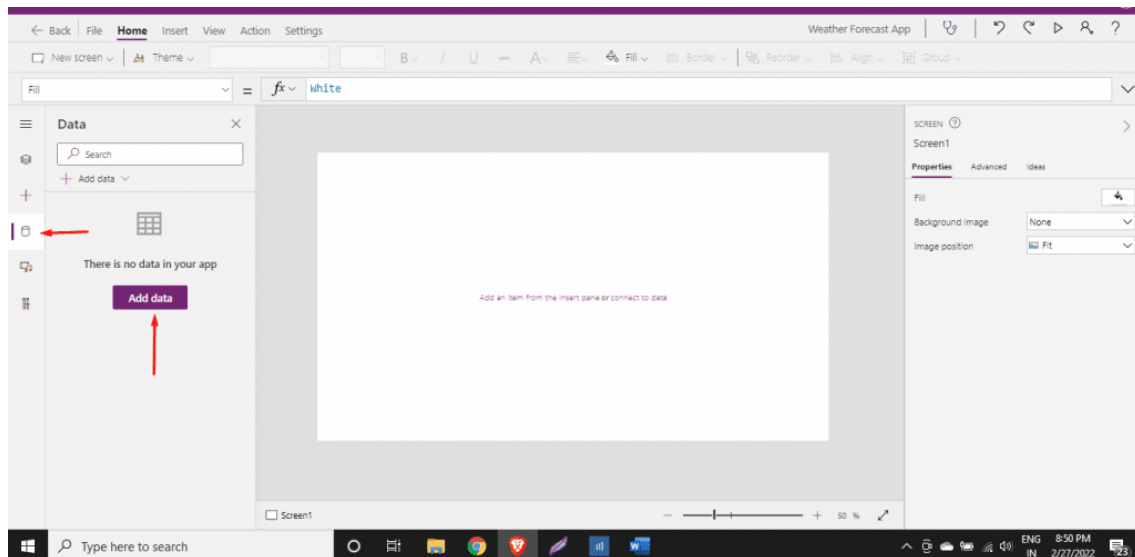


Cree una nueva aplicación de lienzo y proporcione su nombre de aplicación y haga clic en crear.



Ahora, verá una nueva pantalla en blanco.

Haga clic en el icono de datos y seleccione Agregar datos.



Como puede ver, nuestro conector personalizado está disponible. Haga clic en el conector personalizado.

Ahora, solo tienes que usar una etiqueta y ver si funciona o no.

Use la fórmula dada a continuación para probarlo.

El conector personalizado es el nombre del conector que hemos dado anteriormente y el lugar es el ID de la operación.

Nota: use su propia clave API en la identificación de la aplicación.

