

Trabajar con datos mediante código en Microsoft Dataverse

Artículo • 04/01/2024

Dataverse tiene [tablas](#) que se utilizan para modelar y administrar datos empresariales. Puede utilizar las tablas ya proporcionadas o definir sus propias tablas personalizadas para almacenar datos.

Dataverse también tiene API conocidas como *mensajes*. Los mensajes definen un conjunto de parámetros de entrada y propiedades de salida para encapsular la lógica que se ejecuta en el servidor. Cada mensaje tiene un nombre. Si tiene experiencia en bases de datos SQL, puede considerarlos como procedimientos almacenados de SQL. Puede usar mensajes que Dataverse proporciona o puede crear sus propios mensajes.

Usar servicios web para trabajar con datos

Dataverse proporciona dos formas de trabajar con datos: Web API & SDK para .NET. Elija el que coincida mejor con los requisitos y sus cualificaciones.

API Web

La API web es un extremo RESTful de OData v4. Use la API web para cualquier lenguaje de programación que admita solicitudes HTTP y autenticación mediante OAuth 2.0.

Más información: [Utilice la API web de Dataverse](#)

SDK para .NET

Utilice las clases proporcionadas en los ensamblados Dataverse SDK para .NET para aplicaciones personalizadas o para ampliar operaciones Dataverse mediante complementos personalizados y actividades de flujo de trabajo. El SDK de Dataverse para .NET admite objetivos de compilación tanto para .NET Framework como para .NET 6+. Sin embargo, los complementos y las actividades de flujos de trabajo personalizados deben codificarse con .NET Framework.

Más información: [Usar Dataverse SDK para .NET](#)

 Nota

Utilice los ensamblados Xrm.Tooling si desea utilizar el SDK para .NET usando nuestro módulo PowerShell o si está creando una aplicación cliente de Windows y desea utilizar nuestro control de inicio de sesión personalizado. Más información: [Crear aplicaciones cliente de Windows mediante las herramientas XRM](#)

Limitaciones

Hay un límite de tamaño de 1 GB en el tamaño de una respuesta que Dataverse devuelve. Pocas API o consultas son capaces de devolver tantos datos. Si encuentra este límite, debe considerar qué otras opciones están disponibles para obtener los datos en múltiples solicitudes más pequeñas.

El punto de conexión SOAP en desuso devuelve datos XML serializados que son mucho más detallados que los datos JSON serializados devueltos por la API web. Si usa utiliza el punto de conexión SOAP en desuso, debe utilizar la operación equivalente de API web. Más información: [Acerca del punto de conexión de SOAP heredado](#)

Buscar

La búsqueda de Dataverse ofrece resultados rápidos y completos en varias tablas, en una sola lista, ordenada por relevancia. También proporciona capacidades para admitir sugerencias y experiencias de autocompletado en aplicaciones.

La búsqueda tiene un punto de conexión nativo y hay mensajes de Dataverse que puede utilizar desde la API web o el servicio de organización.

Más información: [Buscar registros de Dataverse](#)

ⓘ Nota

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta. ↗ (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales (declaración de privacidad ↗).

Use la API web de Microsoft Dataverse

Artículo • 29/09/2023

Puede usar la API web o [SDK para .NET](#) para trabajar con datos y definiciones de tablas y columnas en Dataverse.

La API web de Dataverse proporciona una experiencia de desarrollo que puede usarse en una gran variedad de lenguajes de programación, plataformas, y dispositivos. La API web implementa OData (Open Data Protocol), versión 4.0, un estándar de OASIS para crear y consumir API RESTful con orígenes de datos enriquecidos. Puede obtener más información sobre este protocolo en <https://www.odata.org/>. Los detalles sobre esta norma están disponibles en <https://www.oasis-open.org/standards#odatav4.0>.

Dado que la API web se basa en estándares abiertos, no proporcionamos ensamblados para una experiencia de desarrollador específica. Puede crear solicitudes HTTP para operaciones específicas o usar bibliotecas de terceros para generar clases para cualquier idioma o plataforma que desee. Puede encontrar una lista de bibliotecas compatibles con OData versión 4.0 en <https://www.odata.org/libraries/>.

API web y el servicio de organización

Es útil reconocer que el servicio de organización es lo que define la plataforma. La API web proporciona una experiencia de programación RESTful, pero en definitiva todas las operaciones de datos pasan por el servicio de organización subyacente. El servicio de organización define las operaciones admitidas como mensajes. Cada mensaje tiene un nombre. Estos nombres están asociados con los eventos que se usan en el marco de trabajo de eventos para evaluar qué extensiones registradas hay que iniciar. Más información: [Marco de trabajo de eventos](#)

La API web permite hacer las mismas operaciones que el SDK para .NET, pero las presenta con estilo RESTful. OData v4 permite usar operaciones con nombre mediante *funciones* o *acciones*. La mayoría de los mensajes disponibles en el servicio de organización se exponen como una función o acción con nombre correspondiente. Esos mensajes que se corresponden con operaciones CRUD no están disponibles en la API web porque, al ser un servicio, RESTful, tienen implementaciones que usan los métodos HTTP GET, POST, PATCH y DELETE, pero dentro de la plataforma los mensajes *recuperar*, *crear*, *actualizar* y *eliminar* solo se invocan tal cual cuando se realizan las operaciones correspondientes mediante los ensamblados de .NET Framework.

Introducción

Ahora que ha leído una descripción general de la API web, continúe con el tema [Comenzar con la API web de Dataverse](#) para aprender a escribir su primer programa C# en Visual Studio que utiliza la API web.

Si es desarrollador de JavaScript y desea utilizar la API web en aplicaciones basadas en modelos, vaya a [JavaScript del lado del cliente utilizando la API web en aplicaciones basadas en modelos](#).

Secciones relacionadas

[Trabajar con datos mediante código](#)

[OData - lo mejor para REST ↗](#)

[OData Versión 4.0 Parte 1: Protocol Plus Errata 02 ↗](#)

[OData Versión 4.0 Parte 2: URL Conventions Plus Errata 02 ↗](#)

[OData Versión 4.0 Parte 3: Common Schema Definition Language \(CSDL\) Plus Errata 02 ↗](#)

ⓘ Nota

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta. ↗ (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad ↗](#)).

Introducción a la API web de Dataverse (C#)

Artículo • 04/08/2023

Esta sección presenta algunas formas simples de comenzar a acceder a la API web de Dataverse utilizando el lenguaje de programación C#.

- [Tutorial: ejemplo de la API web \(C#\)](#) ofrece el ejemplo más rápido de cómo acceder a los datos, ya que todo el código de origen se simplifica y suministra en un archivo.
- [Inicio rápido: muestra de la API web del servidor Blazor \(C#\)](#) muestra cómo crear una aplicación web que puede conectarse a Dataverse y recuperar datos utilizando la API web.

Para obtener más información, explore los [Ejemplos de API web \(C#\)](#).

En esta sección

[Tutorial: Ejemplo de API Web](#)

[Inicio rápido: muestra de API web de Blazor Server \(C#\)](#)

Consulte también

[Realizar operaciones mediante la API web](#)

[Ejemplos de la API web \(C#\)](#)

[Configure un entorno Postman](#)

Nota

¿Puede indicarnos sus preferencias de idioma de documentación? [Realice una breve encuesta.](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad](#)).

Tutorial: Ejemplo de API Web (C#)

Artículo • 16/11/2023

En este inicio rápido, creará una aplicación de consola simple para conectarse a su entorno de Microsoft Dataverse e invocar la función de API web [función WhoAmI](#). Esta función recupera información sobre los usuarios registrados en Dataverse. Una vez que comprenda la funcionalidad básica que se describe aquí, puede pasar a otras operaciones de API web, como crear, recuperar, actualizar y eliminar filas de tablas de Dataverse.

Este programa autenticará y usará un [HttpClient](#) para enviar una solicitud `GET` a la [función WhoAmI](#). La respuesta será un [WhoAmIResponse ComplexType](#). A continuación, el programa mostrará valor de propiedad `UserId` obtenido de la respuesta.

ⓘ Nota

Esto es un ejemplo muy sencillo para mostrar cómo estar conectado con un mínimo de código.

Puede encontrar la solución completa de Visual Studio para este proyecto de .NET 6 en el repositorio [PowerApps-Samples](#) de [dataverse/webapi/C#-NETx/QuickStart](#).

También hay una versión .NET Framework del ejemplo en [dataverse/webapi/C#/QuickStart](#).

Requisitos previos

- Visual Studio 2022 o posterior
- Conexión a Internet
- Cuenta de usuario válida para un entorno de Dataverse
- Dirección URL al entorno de Dataverse con el que quiere conectarse
- Comprensión básica de lenguaje C#

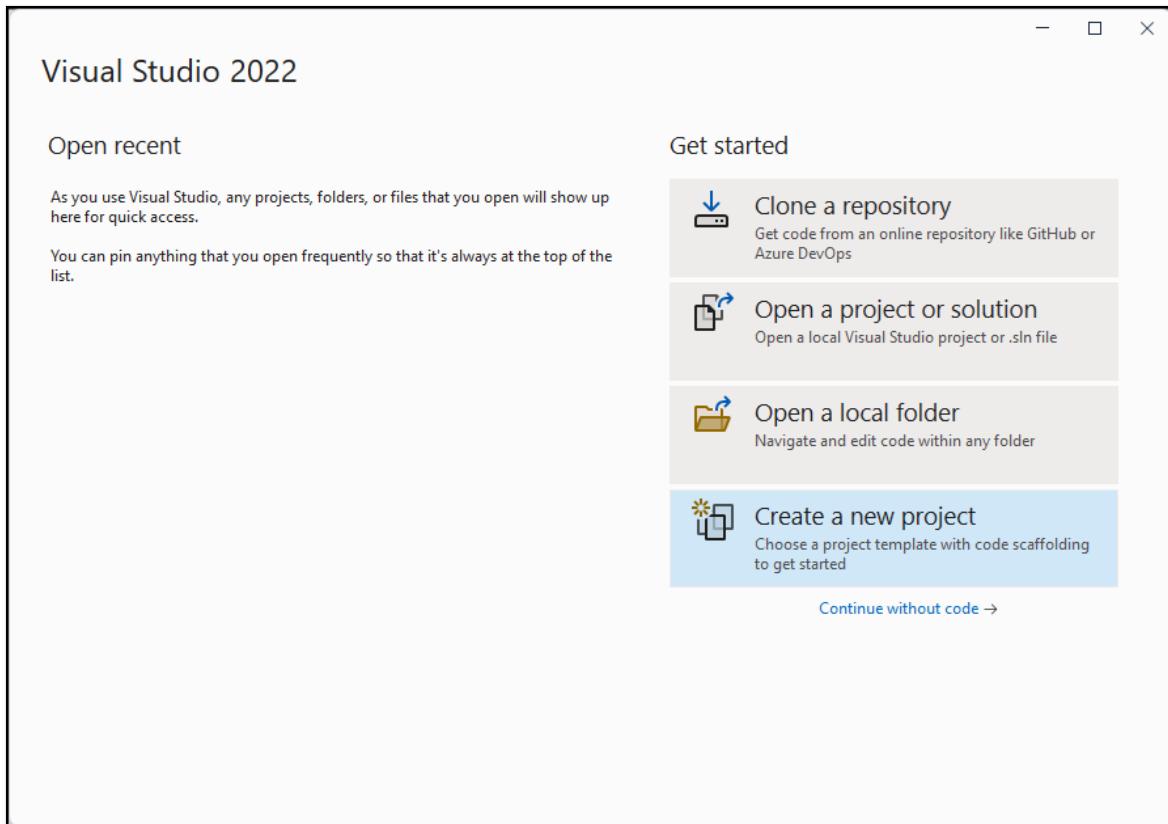
ⓘ Nota

Para autenticarse debe tener una aplicación registrada en Microsoft Entra ID. Este ejemplo de tutorial proporciona un valor `clientid` de registro de la aplicación que puede usar con el fin de ejecutar el código de instalación publicado por Microsoft. Sin embargo, para sus propias aplicaciones personalizadas, debe registrar sus

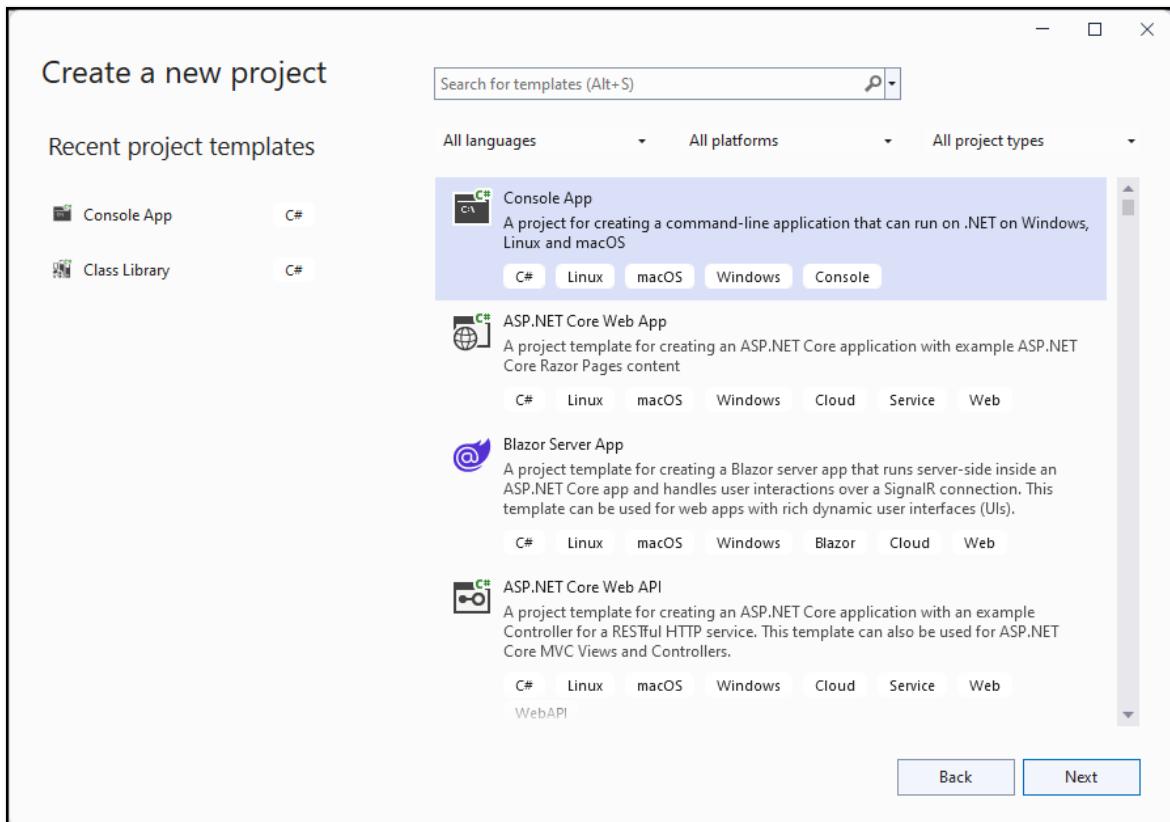
aplicaciones con AD. Más información: [Tutorial: Registrar una aplicación con Microsoft Entra ID](#)

Crear proyecto de Visual Studio

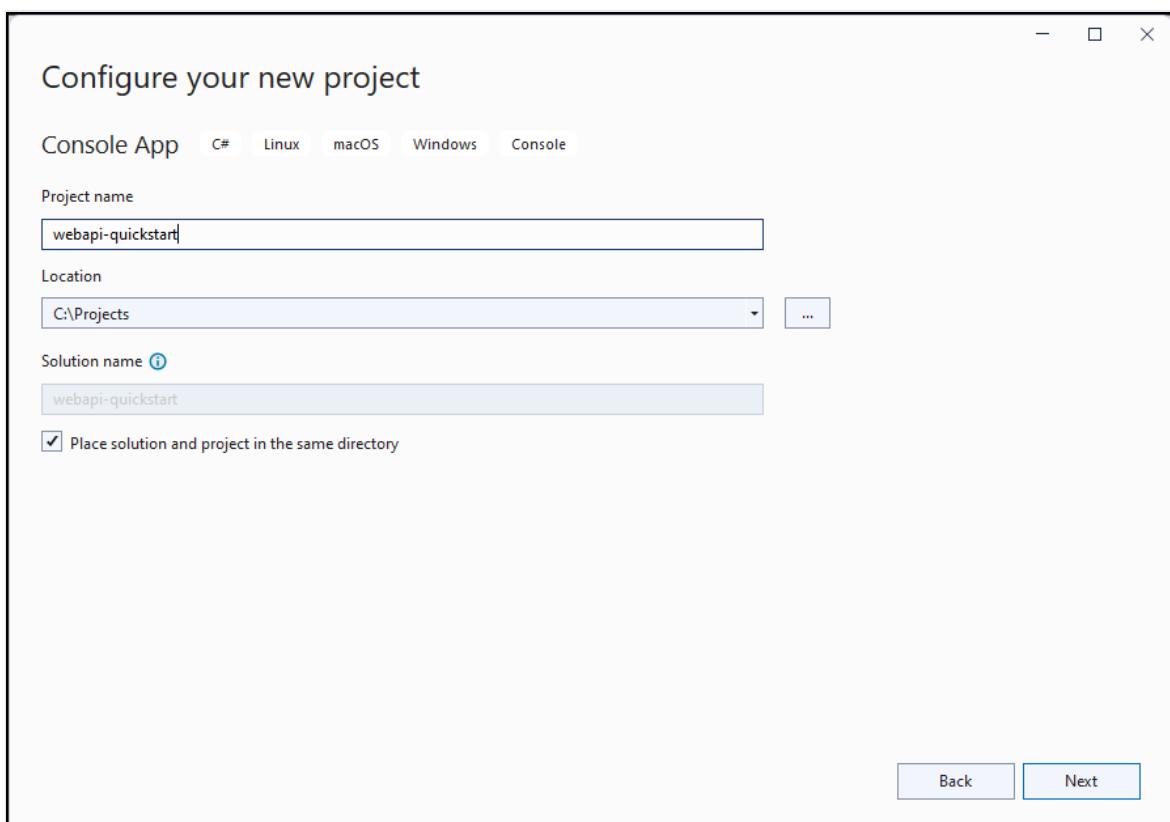
1. Inicie Visual Studio 2022 y seleccione **Crear un nuevo proyecto**.



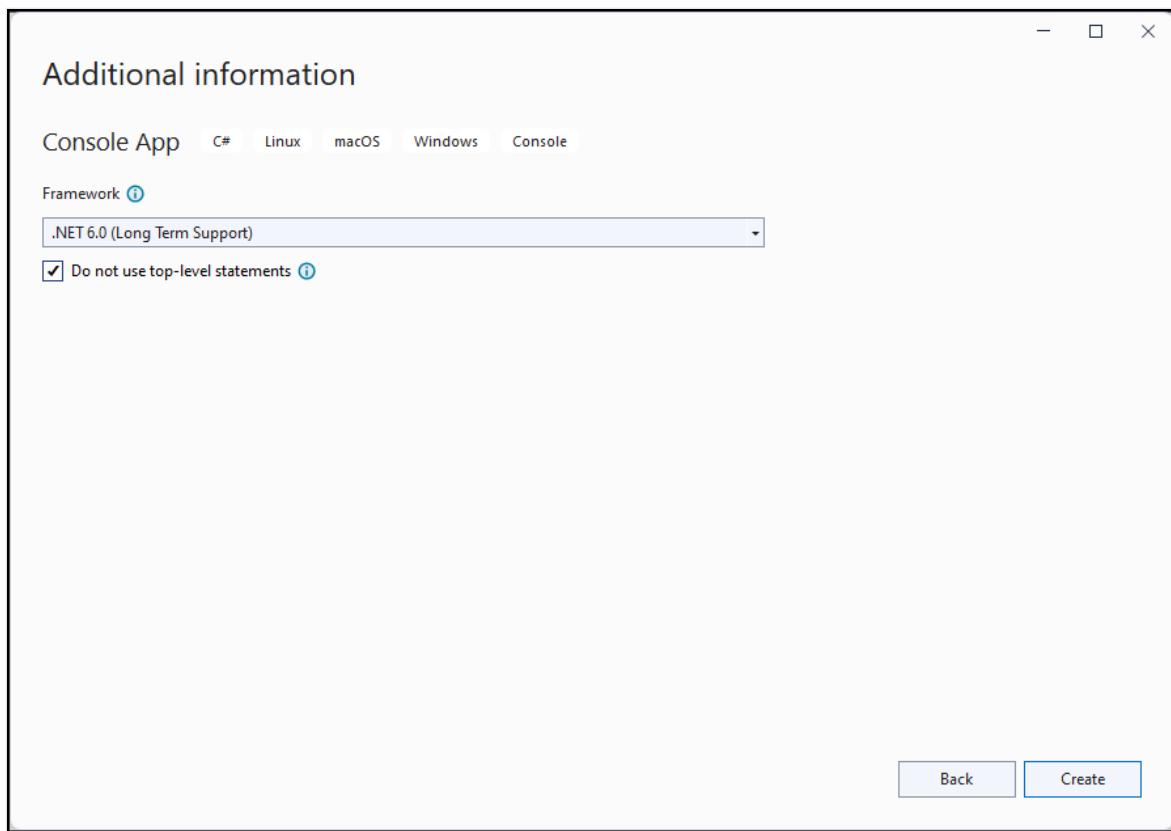
2. Crear un nuevo proyecto de aplicación de **aplicación de consola**.



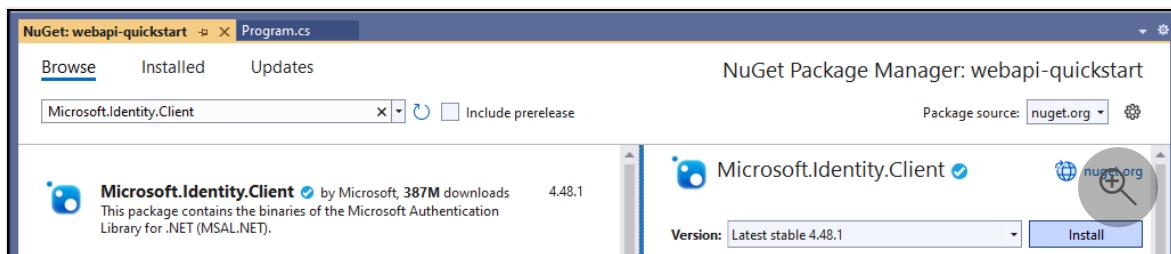
3. Configure el proyecto estableciendo una **Ubicación** y un **Nombre de proyecto**.



4. Configure el proyecto seleccionando **.NET 6.0 (Soporte a largo plazo)** y **No usar declaraciones de nivel superior**. A continuación, haga clic en **Crear**.



5. En el **Explorador de soluciones** haga clic con el botón secundario en el proyecto que ha creado y seleccione **Administrar paquetes de NuGet...** en el menú contextual. [NuGet](#) le permite incluir los ensamblajes necesarios en su proyecto.
6. Busque el paquete Microsoft Authentication Library (MSAL) NuGet con el nombre `Microsoft.Identity.Client`, selecciónelo y, luego, elija **Instalar**.



! Nota

Se le pedirá que acepte los términos de la licencia antes de la instalación.
Haga clic en **Acepto** en el diálogo **Aceptación de licencia**.

Edite Program.cs

Siga los próximos pasos para agregar código para el programa principal.

1. Reemplace el contenido completo de `Program.cs` con el siguiente código.

```
C#  
  
using Microsoft.Identity.Client; // Microsoft Authentication Library  
(MSAL)  
using System;  
using System.Net.Http;  
using System.Net.Http.Headers;  
using System.Text.Json;  
using System.Threading.Tasks;  
  
namespace PowerApps.Samples  
{  
    /// <summary>  
    /// Demonstrates Azure authentication and execution of a Dataverse  
    Web API function.  
    /// </summary>  
    class Program  
    {  
        static async Task Main()  
        {  
            // TODO Specify the Dataverse environment name to connect  
            // with.  
            // See https://learn.microsoft.com/power-  
            // apps/developer/data-platform/webapi/compose-http-requests-handle-  
            // errors#web-api-url-and-versions  
            string resource = "https://<env-name>.api.  
            <region>.dynamics.com";  
  
            // Microsoft Entra ID app registration shared by all Power  
            // App samples.  
            var clientId = "51f81489-12ee-4a9e-aaae-a2591f45987d";  
            var redirectUri = "http://localhost"; // Loopback for the  
            interactive login.  
  
            // For your custom apps, you will need to register them  
            // with Microsoft Entra ID yourself.  
            // See https://docs.microsoft.com/powerapps/developer/data-  
            // platform/walkthrough-register-app-azure-active-directory  
  
            #region Authentication  
  
            var authBuilder =  
                PublicClientApplicationBuilder.Create(clientId)  
  
.WithAuthority(AadAuthorityAudience.AzureAdMultipleOrgs)  
                .WithRedirectUri(redirectUri)  
                .Build();  
            var scope = resource + "/user_impersonation";  
            string[] scopes = { scope };  
  
            AuthenticationResult token =  
                await  
                authBuilder.AcquireTokenInteractive(scopes).ExecuteAsync();  
            #endregion Authentication
```

```

#region Client configuration

    var client = new HttpClient
    {
        // See
        https://docs.microsoft.com/powerapps/developer/data-
        platform/webapi/compose-http-requests-handle-errors#web-api-url-and-
        versions
        BaseAddress = new Uri(resource + "/api/data/v9.2/"),
        Timeout = new TimeSpan(0, 2, 0)      // Standard two
        minute timeout on web service calls.
    };

        // Default headers for each Web API call.
        // See https://docs.microsoft.com/powerapps/developer/data-
        platform/webapi/compose-http-requests-handle-errors#http-headers
        HttpRequestHeaders headers = client.DefaultRequestHeaders;
        headers.Authorization = new
        AuthenticationHeaderValue("Bearer", token.AccessToken);
        headers.Add("OData-MaxVersion", "4.0");
        headers.Add("OData-Version", "4.0");
        headers.Accept.Add(
            new
            MediaTypeWithQualityHeaderValue("application/json"));
    #endregion Client configuration

    #region Web API call

        // Invoke the Web API 'WhoAmI' unbound function.
        // See https://docs.microsoft.com/powerapps/developer/data-
        platform/webapi/compose-http-requests-handle-errors
        // See https://docs.microsoft.com/powerapps/developer/data-
        platform/webapi/use-web-api-functions#unbound-functions
        var response = await client.GetAsync("WhoAmI");

        if (response.IsSuccessStatusCode)
        {
            // Parse the JSON formatted service response
            (WhoAmIResponse) to obtain the user ID value.
            // See https://learn.microsoft.com/power-
            apps/developer/data-platform/webapi/reference/whoamiresponse
            Guid userId = new();

            string jsonContent = await
            response.Content.ReadAsStringAsync();

            // Using System.Text.Json
            using (JsonDocument doc =
            JsonDocument.Parse(jsonContent))
            {
                JsonElement root = doc.RootElement;
                JsonElement userIdElement =
                root.GetProperty("UserId");
                userId = userIdElement.GetGuid();
            }
        }
    
```

```

        }

        // Alternate code, but requires that the WhoAmIResponse
        // class be defined (see below).
        // WhoAmIResponse whoAmIresponse =
        JsonSerializer.Deserialize<WhoAmIResponse>(jsonContent);
        // userId = whoAmIresponse.UserId;

        Console.WriteLine($"Your user ID is {userId}");
    }
    else
    {
        Console.WriteLine("Web API call failed");
        Console.WriteLine("Reason: " + response.ReasonPhrase);
    }
    #endregion Web API call
}
}

/// <summary>
/// WhoAmIResponse class definition
/// </summary>
/// <remarks>To be used for JSON deserialization.</remarks>
/// <see cref="https://learn.microsoft.com/power-
apps/developer/data-platform/webapi/reference/whoamiresponse"/>
public class WhoAmIResponse
{
    public Guid BusinessUnitId { get; set; }
    public Guid UserId { get; set; }
    public Guid OrganizationId { get; set; }
}
}

```

2. Directamente debajo del comentario TODO en el código anterior, reemplace el valor variable de `resource` con la URL real de su entorno de prueba de Dataverse. Para encontrar el valor de URL para su entorno de prueba, siga estos pasos:
 - a. En su navegador, vaya a [Power Apps](#).
 - b. Seleccione el ícono de entornos (a la derecha del campo de búsqueda) y elija un entorno de prueba.
 - c. Seleccione el ícono de configuración  y elija **Herramientas y recursos de desarrollo**.
 - d. Copie la URL del punto de conexión de la API web desde "https:" hasta ".com" dejando el final `/api/data/v9.2`.
 - e. Reemplace el valor de la cadena de recursos en el código del programa con ese valor de URL del punto de conexión. Por ejemplo:

```
string resource = "https://contoso.api.crm.dynamics.com";
```

Ejecutar el programa

1. Presione F5 para crear y ejecutar el programa.

Se abrirá una ventana del navegador y le pedirá que elija una cuenta. Elija la cuenta que utiliza para acceder a su entorno Dataverse . Si esa cuenta no aparece en la lista, haga clic en **Usar otra cuenta**.

Una vez seleccionada la cuenta, introduzca su contraseña y haga clic en **iniciar sesión**.

2. Mire la ventana de la aplicación de la consola. El resultado deberá ser ahora similar a esto:

```
Your user ID is 4026be43-6b69-e111-8f65-78e7d1620f5e  
C:\Projects\webapi-quickstart\bin\Debug\net6.0\webapi-quickstart.exe  
(process 21524) exited with code 0.  
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.  
Press any key to close this window . . .
```

Enhorabuena.

Se ha conectado con éxito a la API web.

El ejemplo de inicio rápido muestra un enfoque básico para crear un proyecto de Visual Studio sin ningún control de las excepciones o sin método para actualizar el token de acceso. Esto es suficiente para verificar que puede conectarse y probar diferentes operaciones.

Para ver un ejemplo más completo que demuestre los patrones de diseño recomendados, revise la [Biblioteca de clases WebAPIService \(C#\)](#). Este es el proyecto que usamos para nuestros [Ejemplos de operaciones de datos de la API web \(C#\)](#).

Demuestra:

- Administrar los [límites de la API de protección del servicio](#) de Dataverse con la resiliencia de .NET y la biblioteca de manejo de fallas transitorias [Polly](#).
- Administrar un [HttpClient](#) en .NET usando [IHttpClientFactory](#).
- Usar datos de configuración para gestionar el comportamiento del cliente.
- Gestionar los errores devueltos por la API web de Dataverse.

- Un patrón de reutilización de código mediante:
 - Creando clases que heredan de [HttpRequestMessage](#) y [HttpResponseMessage](#).
 - Métodos que usan esas clases.
 - Un patrón modular para agregar nuevas capacidades según sea necesario.

Pasos siguientes

Pruebe a crear una aplicación web.

[Inicio rápido: muestra de API web de Blazor Server \(C#\)](#)

Aprender más acerca de las Capacidades de API web de Dataverse mediante la comprensión de los documentos de servicio.

[Tipos y operaciones de API web](#)

 Nota

¿Puede indicarnos sus preferencias de idioma de documentación? [Realice una breve encuesta.](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad](#)).

Inicio rápido: muestra de API web de Blazor Server (C#)

Artículo • 02/11/2023

En este inicio rápido, creará una aplicación de Blazor Server para conectarse a su entorno Microsoft Dataverse utilizando la API web.

Se autenticará y usará [HttpClient](#) para enviar una solicitud `GET` que contiene la función `WhoAmI`. La respuesta es un [WhoAmIResponse ComplexType](#). Después de completar la llamada, se muestran las propiedades `WhoAmIResponse`.

ⓘ Nota

Esto es un ejemplo muy sencillo para mostrar cómo estar conectado con un mínimo de código.

Requisitos previos

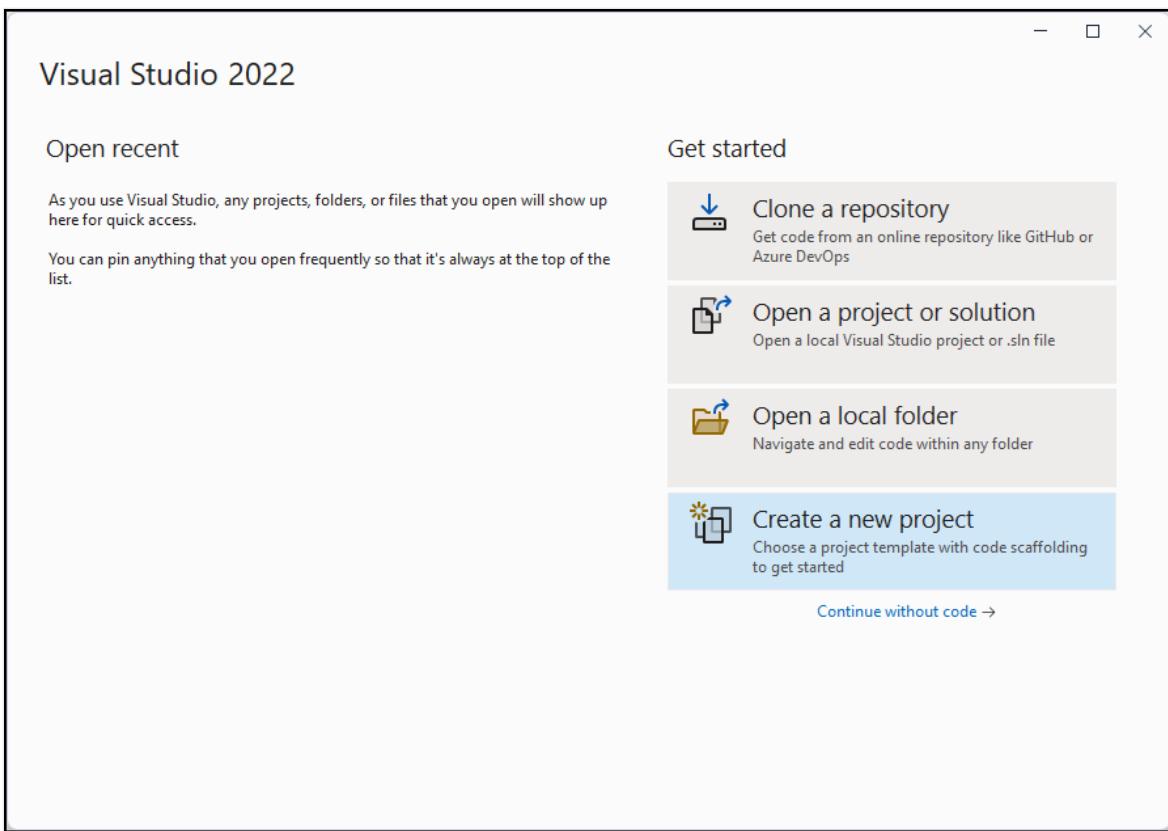
- Visual Studio 2022 con la carga de trabajo **ASP.NET y desarrollo web** .
- [.NET 7.0 SDK](#) .
- Familiaridad con el portal Microsoft Azure.
- Conexión a Internet.
- Cuenta de usuario válida para una instancia de Dataverse.
- Acceso de administrador para otorgar registros de aplicaciones.
- Dirección URL al entorno de Dataverse con el que quiere conectarse.
- Comprensión básica de lenguaje Visual C#.

ⓘ Nota

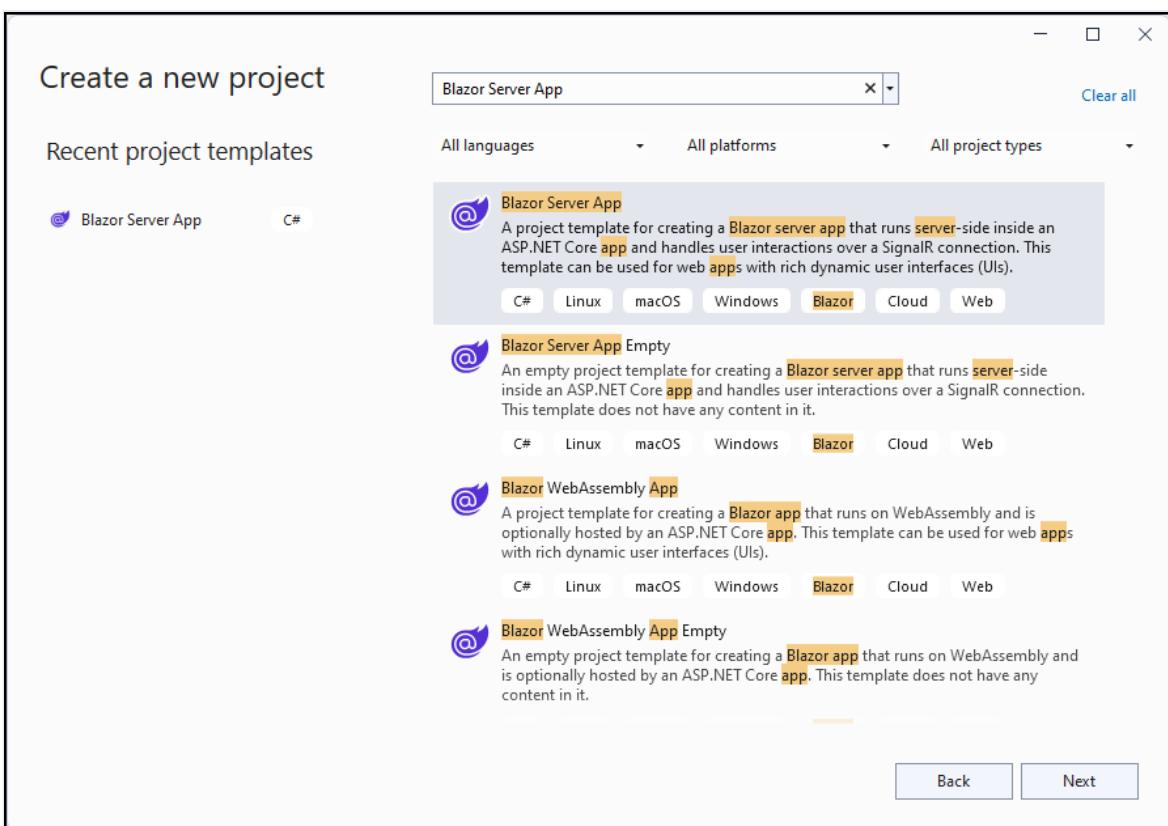
Para autenticarse debe tener una aplicación registrada en Microsoft Entra ID. El registro se realizará automáticamente como parte de la creación de la plantilla, pero requerirá actualizaciones adicionales en el Azure Portal.

Crear un proyecto de Visual Studio

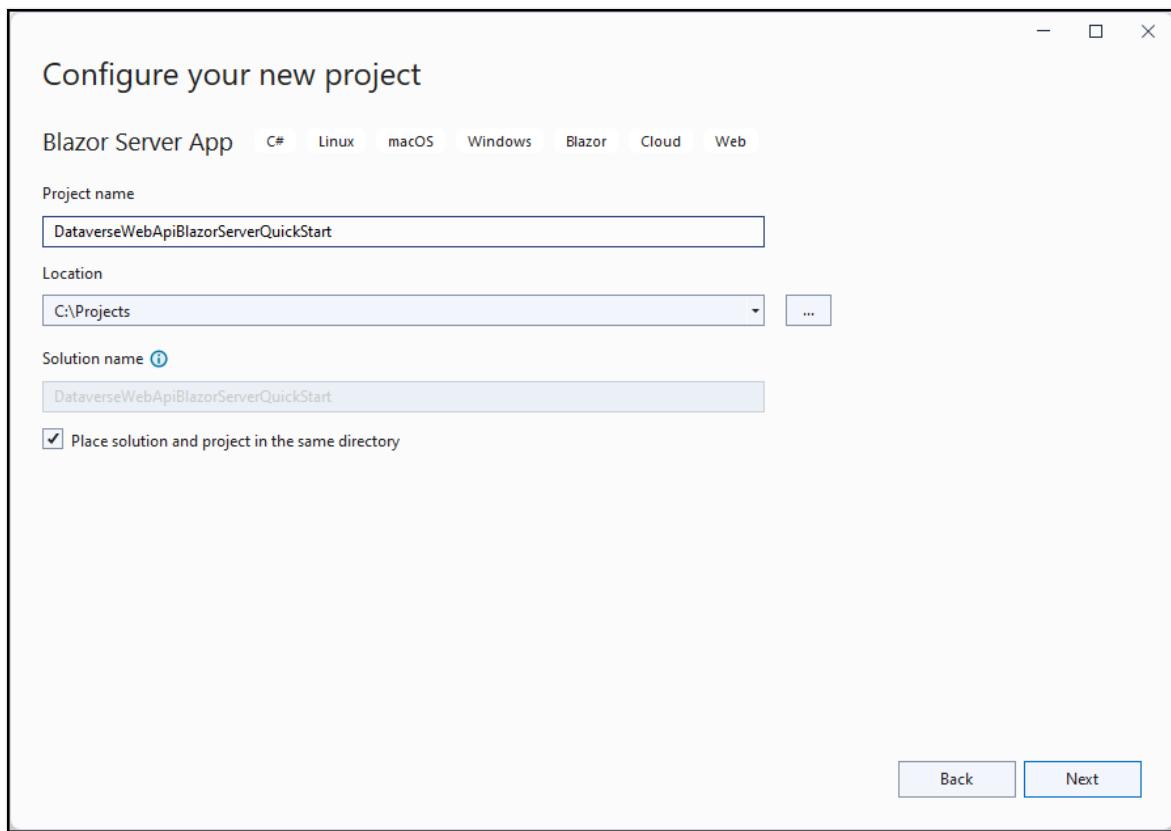
1. Abra Visual Studio 2022 y seleccione **Crear un nuevo proyecto**.



2. En el cuadro de diálogo **Crear un nuevo proyecto**, busque *Aplicación Blazor Server*. Seleccione la plantilla y luego **Siguiente**.



3. En el cuadro de diálogo **Configurar su nuevo proyecto**, configure el **Nombre del proyecto** y la **Ubicación**. Seleccione **Siguiente**.



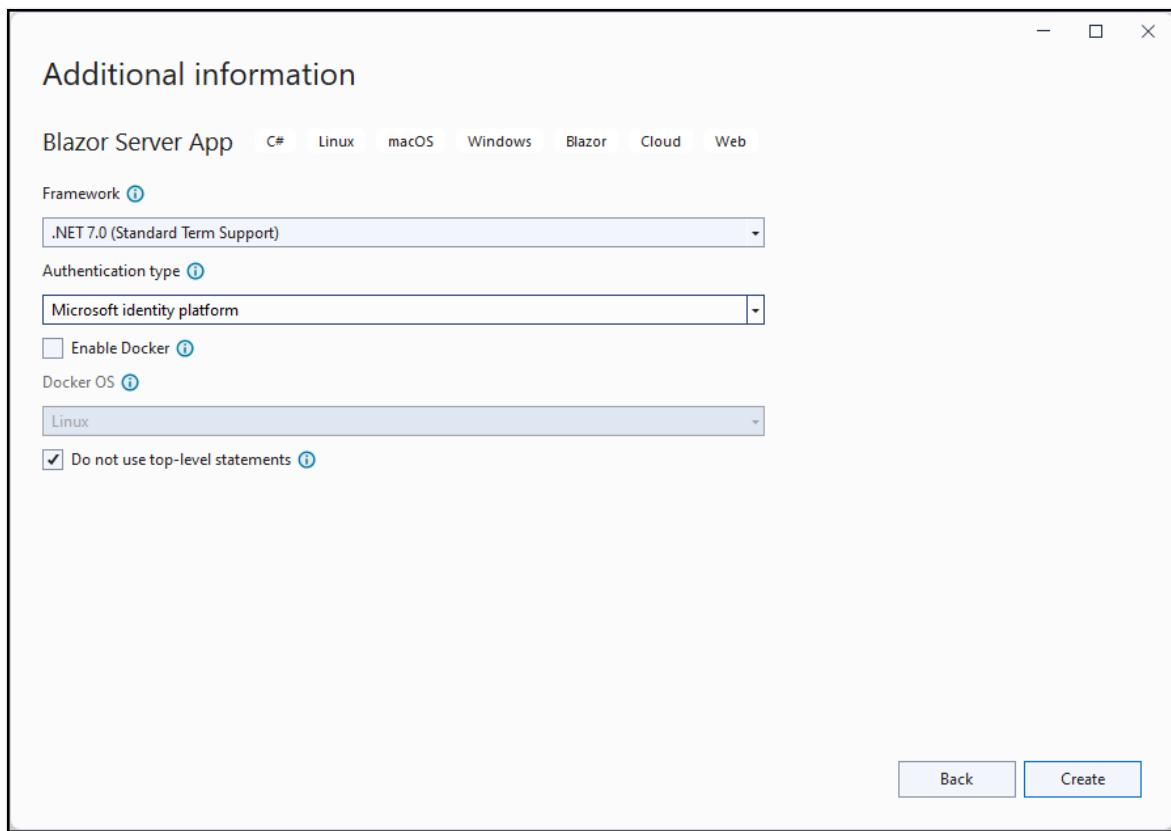
En este ejemplo, usaremos **DataverseWebApiBlazorServerInicio rápido** como **Nombre del proyecto**.

4. En el cuadro de diálogo **Información adicional**, especifique **Marco** y **Tipo de autenticación**.

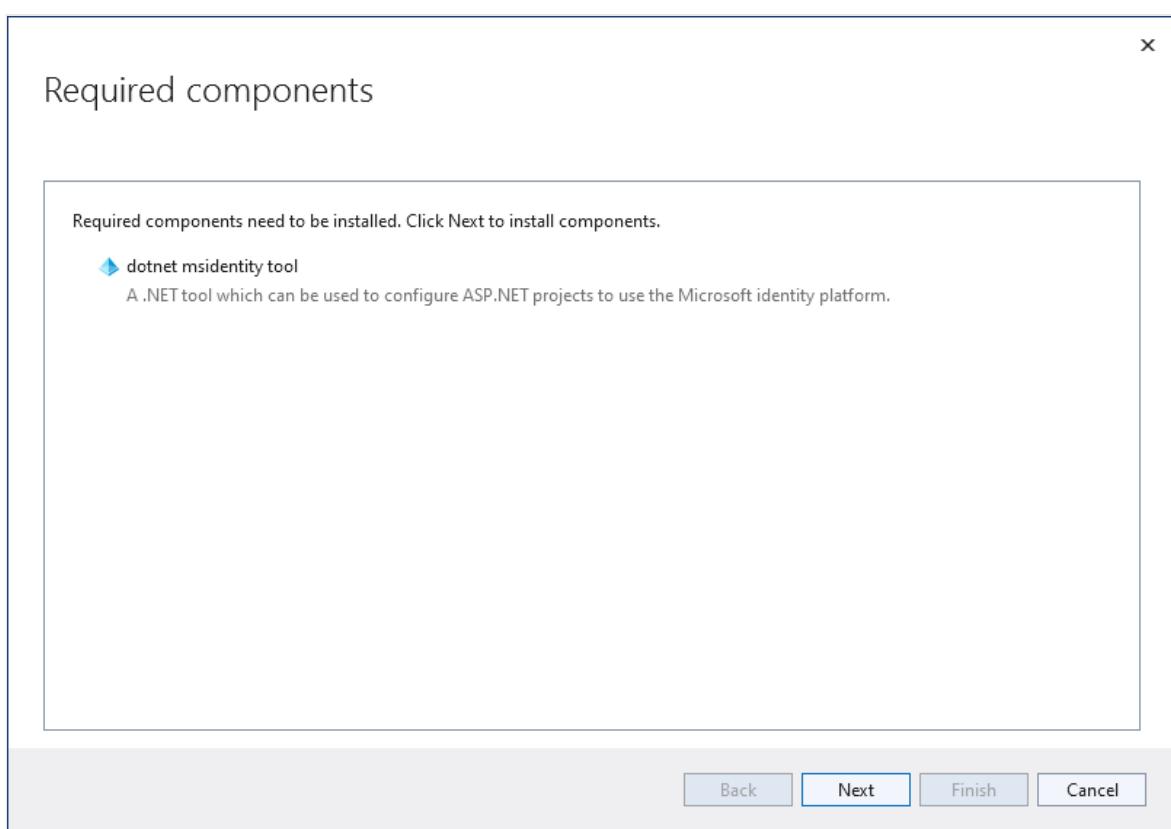
En este ejemplo, el marco es **.NET 7.0 (Soporte de plazo estándar)**.

ⓘ Importante

Establezca el **Tipo de autenticación** en **Plataforma de identidad de Microsoft**.



5. Seleccione **Crear** para crear el proyecto.
6. La plantilla del proyecto abre un cuadro de diálogo **Componentes necesarios**.
Seleccione **Siguiente**.



7. En la **plataforma de identidad de Microsoft**, asegúrese de que la cuenta de Azure seleccionada tenga permisos para administrar aplicaciones en Microsoft Entra ID y

el inquilino seleccionado es el asociado con su entorno de Power Platform.

The screenshot shows the Microsoft Identity Platform configuration interface. At the top, it displays "Microsoft identity platform" and "Configuring DataverseWebApiBlazorServerQuickStart". On the left, there's a "Tenants" dropdown set to "Your Tenant (yourorg.onmicrosoft.com)". The main area is titled "Owned applications" and lists columns: "Display name", "Application (client) ID", and "Created on". A red box highlights the "+ Create new" button at the top right of this list. At the bottom, there are navigation buttons: "Back", "Next", "Finish", and "Cancel".

8. Seleccione **Crear nuevo**.

9. En el cuadro de diálogo **Registrar una aplicación**, configure el **nombre** y seleccione **Registro** para cerrar el diálogo.

The screenshot shows the "Register an application" dialog box. It has a "Selected tenant" field set to "Your Tenant (yourorg.onmicrosoft.com)" and a "Display name" field containing "Dataverse Web Api Blazor Server Quick Start". At the bottom, there are "Register" and "Cancel" buttons.

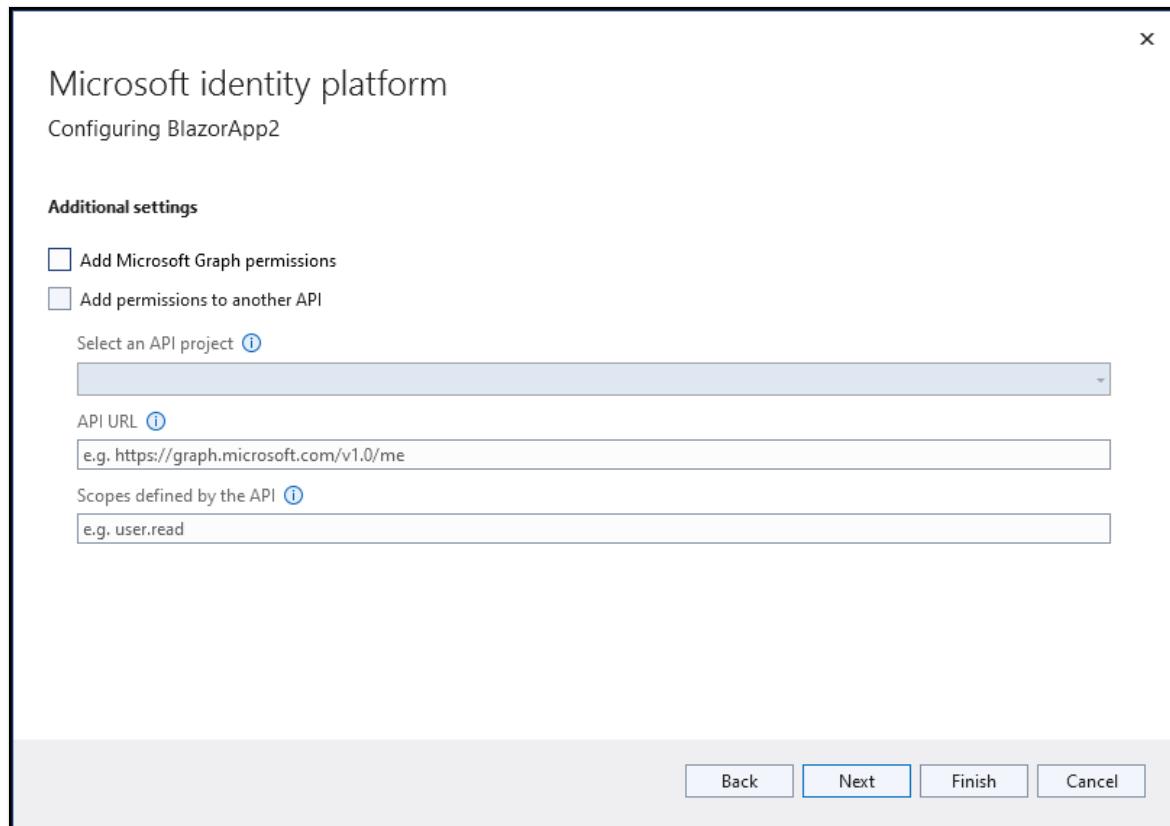
En este ejemplo, estamos usando el nombre *Inicio rápido del servidor Web Api Blazor de Dataverse*. Buscaremos la aplicación con este nombre en un paso posterior.

10. Seleccione Siguiente.

ⓘ Nota

No necesita hacer nada en este paso.

Este paso proporciona capacidades para conectarse a Microsoft Graph u otra API, pero no es necesario conectarse a otra API para este inicio rápido.



11. Seleccione Siguiente. Este paso resume los cambios realizados en el proyecto.

Summary of changes

Project changes for dependencies being added or configured:

- ◆ Microsoft identity platform
 - Code
 - Code will be added to the project to enable Azure AD
 - NuGet packages
 - NuGet packages will be modified to ensure optimal operation of the new dependency
 - Update Azure AD
 - Calls the dotnet msidentity command line tool to finish updating Azure AD

Back Next Finish Cancel

12. Seleccione Finalizar.

El cuadro de diálogo **Progreso de la configuración de dependencias** muestra los pasos automatizados realizados por la plantilla para registrar la aplicación.

Dependency configuration progress

```
Connecting to Microsoft identity platform dependency identityapp1 in the project...
Updating app registration in Azure AD: dotnet msidentity --update-app-registration --username [REDACTED].onmicrosoft.com --
tenant-id [REDACTED] --client-id [REDACTED] --enable-id-token --json --
redirect-uris http://localhost:15483/ https://localhost:44360/ https://localhost:7056/ http://localhost:5078/
Installing NuGet packages to project...
Installing package 'Microsoft.Identity.Web' with version '1.24.1'.
Installing package 'Microsoft.Identity.Web.UI' with version '1.16.0'.
Skipping package 'Microsoft.Identity.Web.UI', same version or a newer version is already installed.
Installing package 'Microsoft.AspNetCore.Authentication.JwtBearer' with version '6.0.5'.
Skipping package 'Microsoft.AspNetCore.Authentication.JwtBearer', same version or a newer version is already installed.
Installing package 'Microsoft.AspNetCore.Authentication.OpenIdConnect' with version '6.0.5'.
Skipping package 'Microsoft.AspNetCore.Authentication.OpenIdConnect', same version or a newer version is already installed.
Updating project code and settings: dotnet msidentity --update-project --username [REDACTED].onmicrosoft.com --tenant-id
[REDACTED] --client-id [REDACTED] --project-file-path "C:\Projects
\DataverseWebApiBlazorServerQuickStart\DataverseWebApiBlazorServerQuickStart.csproj" --calls-graph=False --calls-downstream-
api=False --code-update=true --json
Modified Program.cs
Modified code file Program.csModified code file App.razorModified code file MainLayout.razorAdded code file Shared/
LoginDisplay.razor
Serializing new Microsoft identity platform dependency metadata to disk...
 Complete. Microsoft identity platform identityapp1 is configured.
```

Automatically close when succeeded

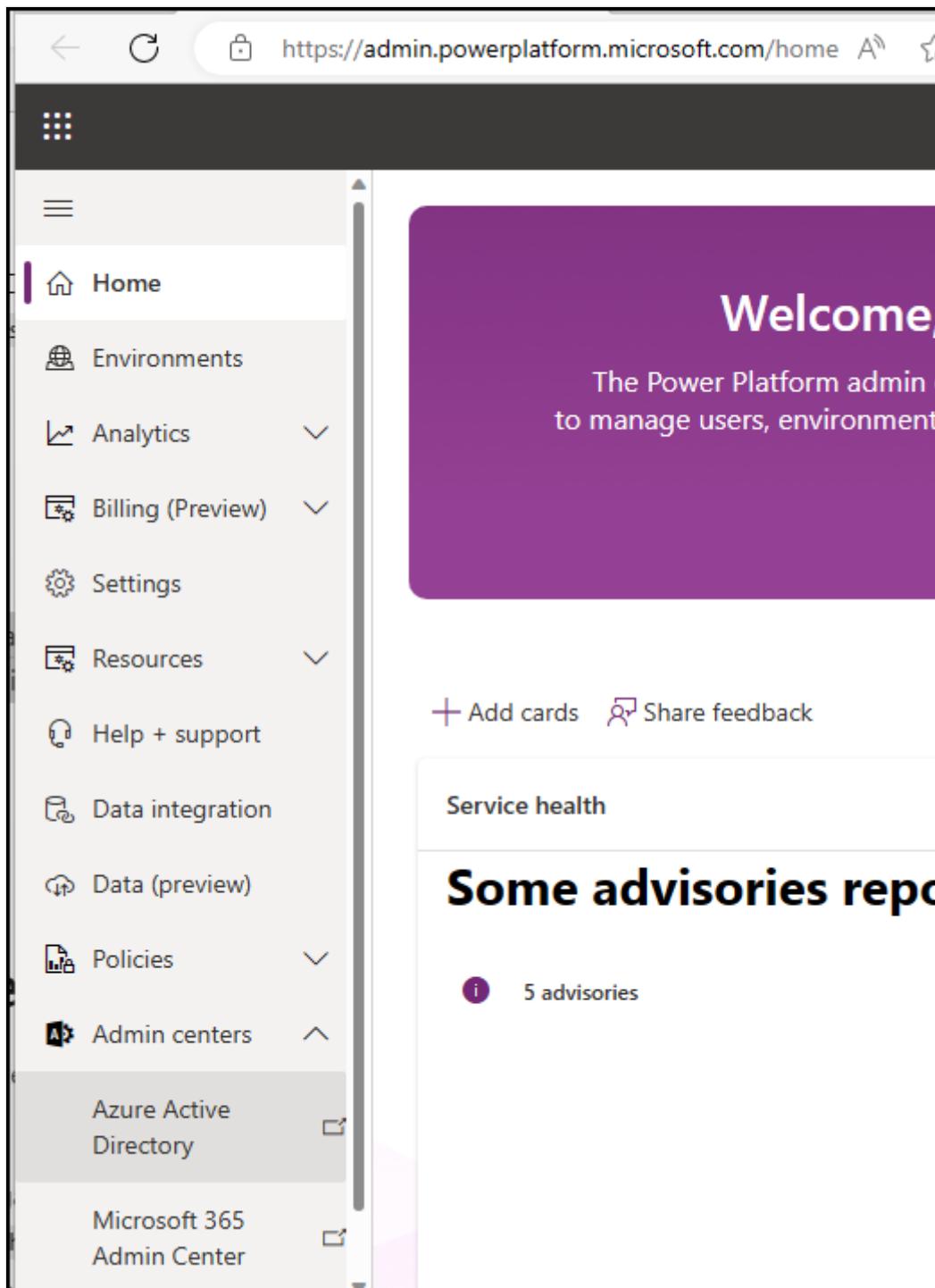
Back Next Close Cancel

13. Seleccione Cerrar para cerrar el cuadro de diálogo.

Configurar la aplicación en Microsoft Entra ID

La plantilla de Visual Studio creó una aplicación registrada utilizando la información que proporcionó. La conexión a Dataverse requiere más permisos.

1. En el [Centro de administración de Power Platform](#), seleccione **Centros de administración > Microsoft Entra ID**.



2. En el **Centro de administración de Microsoft Entra**, busque la aplicación creada por nombre dentro de **Registros de aplicaciones**.

The screenshot shows the Microsoft Entra admin center interface. The top navigation bar includes 'Home', 'Azure Active Directory', 'Overview', 'Users', and 'App Registrations (1)'. The main content area is titled 'App Registrations' and shows a single entry: 'Dataverse Web Api Blazor Server Quick Start' (Application). There is a search bar at the bottom with placeholder text 'Start typing a display name or application (client) ID to filter these results...' and a 'Give feedback' link.

3. Abra la aplicación y seleccione **Permisos de API**. Seleccione **Agregar permiso**.

This screenshot shows the 'API permissions' section of the application registration. On the left, there's a sidebar with options like 'Overview', 'Quickstart', 'Integration assistant', 'Manage', 'Branding & properties', 'Authentication', 'Certificates & secrets', 'Token configuration', 'API permissions' (which is selected), 'Expose an API', 'App roles', and 'Owners'. The main content area is titled 'Configured permissions' with a note about consent. It features a 'Add a permission' button with a red box around it. Below it is a table with columns 'API / Permissions name', 'Type', 'Description', and 'Admin cor...'. The table currently shows 'No permissions added'. A note at the bottom suggests checking 'Enterprise applications' for more information.

4. En el menú flotante **Solicitar permisos de API**, seleccione la pestaña **Las API que usa mi organización** y busque *Dataverse*.

This screenshot shows the 'Request API permissions' dialog. At the top, there are tabs for 'Microsoft APIs', 'APIs my organization uses' (which is selected and highlighted with a red box), and 'My APIs'. Below the tabs, it says 'Apps in your directory that expose APIs are shown below'. A search bar contains the text 'Dataverse' (also highlighted with a red box). A table lists one application: 'Name' (Dataverse) and 'Application (client) ID' (00000007-0000-0000-c000-000000000000).

5. Seleccione **Dataverse** y se abrirá **Dynamics CRM API**.

6. Seleccione el permiso delegado `user_impersonation` y seleccione **Agregar permisos**.

Request API permissions

[All APIs](#)



Dynamics CRM

<https://admin.services.crm.dynamics.com/> [Docs](#)

What type of permissions does your application require?

Delegated permissions

Your application needs to access the API as the signed-in user.

Select permissions

Start typing a permission to filter these results

Permission

Permissions (1)



user_impersonation ⓘ

Access Common Data Service as organization users

[Add permissions](#)

[Discard](#)

7. Seleccione Certificados y secretos y seleccione Nuevo secreto de cliente.

[Home](#) > [Dataverse Web Api Blazor Server Quick Start](#)



Dataverse Web Api Blazor Server Quick Start | Certificates & secrets



[Overview](#)

[Quickstart](#)

[Integration assistant](#)

Manage

[Branding & properties](#)

[Authentication](#)

[Certificates & secrets](#)

[Token configuration](#)

[API permissions](#)

[Expose an API](#)

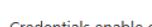
[App roles](#)

[Owners](#)

[Roles and administrators](#)

[Manifest](#)

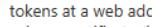
Got feedback?



Got feedback?



Got feedback?



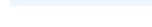
Got feedback?



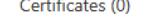
Got feedback?



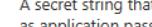
Got feedback?



Got feedback?



Got feedback?



Got feedback?



Got feedback?



Got feedback?



Got feedback?



Got feedback?



Got feedback?



Got feedback?



Got feedback?



Got feedback?

8. En el menú emergente **Agregar un secreto de cliente**, ingrese una **Descripción** y una duración de **Caducidad**, luego seleccione **Agregar**.

Add a client secret

Description	Confidential Client Required App Secret
Expires	Recommended: 6 months

Add **Cancel**

ⓘ Importante

Copie el secreto ahora. No puede acceder después de salir de esta página.

New client secret			
Description	Expires	Value ⓘ	Copy to clipboard
Confidential Client Required App Se...	6/19/2023	[REDACTED]	

9. En Visual Studio, dentro de su proyecto de aplicación Blazor Server, abra `appsettings.json` y agregue una entrada para `ClientSecret` con el valor secreto. El contenido del archivo `appsettings.json` debe ser de este tipo:

JSON

```
{
  "AzureAd": {
    "Instance": "https://login.microsoftonline.com/",
    "Domain": "<your org>.onmicrosoft.com",
    "TenantId": "<your tenantid>",
    "ClientId": "<your clientid>",
    "CallbackPath": "/signin-oidc",
    "ClientSecret": "<your secret>"
  },
  "Logging": {
    "LogLevel": {
      "Default": "Information",
      "Microsoft.AspNetCore": "Warning"
    }
  },
  "AllowedHosts": "*"
}
```

Edición de la aplicación

Para habilitar las llamadas a Dataverse, debe editar tres archivos en la aplicación:

- appsettings.json
- Program.cs
- Pages/FetchData.razor

appsettings.json

Hay varios lugares en los otros archivos que requieren una referencia al uri de base utilizado para acceder a la API web de Dataverse . Agregar estos datos a `appsettings.json` le permite configurar estos datos en un solo lugar.

Agregue lo siguiente debajo de `"AllowedHosts": "*"` donde `<your org>` representa la URL base para acceder a la API web de Dataverse. Si no está seguro de qué se trata, consulte [URL y versiones de API web](#).

JSON

```
"AllowedHosts": "*",
"DataverseConfig": {
    "BaseUri": "https://<your org>.api.crm.dynamics.com/"
}
```

ⓘ Nota

Compruebe lo siguiente:

- Agregue la coma después de `"AllowedHosts": "*"` para que el JSON sea válido.
- Finalice el `BaseUri` con `"/"`.

Program.cs

1. Para acceder al valor del URI de base de la API web en la configuración, agregue la línea a continuación comentada con `// Get BaseUri from appsettings.json` en el método `Main`, debajo de la línea:

```
var builder = WebApplication.CreateBuilder(args);
```

C#

```

public static void Main(string[] args)
{
    var builder = WebApplication.CreateBuilder(args);

    // Get BaseUri from appsettings.json
    string dataverseBaseUri =
        builder.Configuration.GetSection("DataverseConfig").GetValue<string>
            ("BaseUri");

```

2. Agregue las siguientes dos líneas debajo de la línea:

```
.AddMicrosoftIdentityWebApp(builder.Configuration.GetSection("AzureAd"))
```

C#

```

// Add services to the container.
builder.Services.AddAuthentication(OpenIdConnectDefaults.Authentication
Scheme)

.AddMicrosoftIdentityWebApp(builder.Configuration.GetSection("AzureAd"))
    .EnableTokenAcquisitionToCallDownstreamApi(new string[] { $""
{dataverseBaseUri}user_impersonation" })
    .AddInMemoryTokenCaches();

```

- El método

`MicrosoftIdentityWebApiClientAuthenticationBuilder.EnableTokenAcquisitionToCallDownstreamApi` agrega compatibilidad para que la aplicación web adquiera tokens para llamar a una API. Al pasar el alcance de `user_impersonation`, el usuario puede aceptar la capacidad de usar la API web de Dataverse.

- `AddInMemoryTokenCaches Method` Habilita el almacenamiento en caché del token emitido para las solicitudes.

Pages/FetchData.razor

El componente predeterminado `Pages/FetchData.razor` recupera algunos datos de pronóstico del tiempo. Vamos a reemplazar esto por completo.

Copie el siguiente código y reemplace todo el código en `Pages/FetchData.razor`:

razor

```

@page "/fetchdata"
@using Microsoft.Identity.Client;
@using Microsoft.Identity.Web
@using System.Text.Json;
@inject MicrosoftIdentityConsentAndConditionalAccessHandler ConsentHandler

```

```

@inject IHttpConnectionFactory HttpClientFactory
@inject Microsoft.Identity.Web.ITokenAcquisition TokenAcquisitionService
@inject IConfiguration configuration;

<PageTitle>WhoAmI</PageTitle>

<h1>WhoAmI</h1>

<p>This component demonstrates fetching data from Dataverse.</p>

@if (WhoIAm == null)
{
    <p><em>Loading...</em></p>
}
else
{
    <table class="table">
        <thead>
            <tr>
                <th>Property</th>
                <th>Value</th>
            </tr>
        </thead>
        <tbody>
            <tr>
                <td>BusinessUnitId</td>
                <td>@WhoIAm.BusinessUnitId</td>
            </tr>
            <tr>
                <td>OrganizationId</td>
                <td>@WhoIAm.OrganizationId</td>
            </tr>
            <tr>
                <td>UserId</td>
                <td>@WhoIAm.UserId</td>
            </tr>
        </tbody>
    </table>
}

@code {
    private WhoAmIResponse WhoIAm;
    private HttpClient _httpClient;

    protected override async Task OnInitializedAsync()
    {
        string baseUrl = configuration["DataverseConfig:BaseUri"];

        // Get the HttpClient
        _httpClient = HttpClientFactory.CreateClient();

        // Get the token
        var token = string.Empty;
        try
        {

```

```

        token = await
TokenAcquisitionService.GetAccessTokenForUserAsync(new string[] { $""
{baseUrl}user_impersonation" });
    }
    // Microsoft Identity Web specific exception class for use in Blazor
or Razor pages to process the user challenge.
    // Handles the MsalUiRequiredException.
    catch (MicrosoftIdentityWebChallengeUserException ex)
    {
        ConsentHandler.HandleException(ex);
    }
    catch (Exception)
    {
        throw new Exception("Error getting access token.");
    }

    // Set the auth token
    _httpClient.DefaultRequestHeaders.Authorization = new
System.Net.Http.Headers.AuthenticationHeaderValue("Bearer", token);

    // Send the request
    var dataRequest = await _httpClient.GetAsync($""
{baseUrl}api/data/v9.2/WhoAmI");

    if (dataRequest.IsSuccessStatusCode)
    {
        var jsonString = System.Text.Json.JsonDocument.Parse(await
dataRequest.Content.ReadAsStreamAsync());
        WhoIAm = JsonSerializer.Deserialize<WhoAmIResponse>(jsonString);
    }
    else
    {
        throw new Exception("Error sending request.");
    }
}

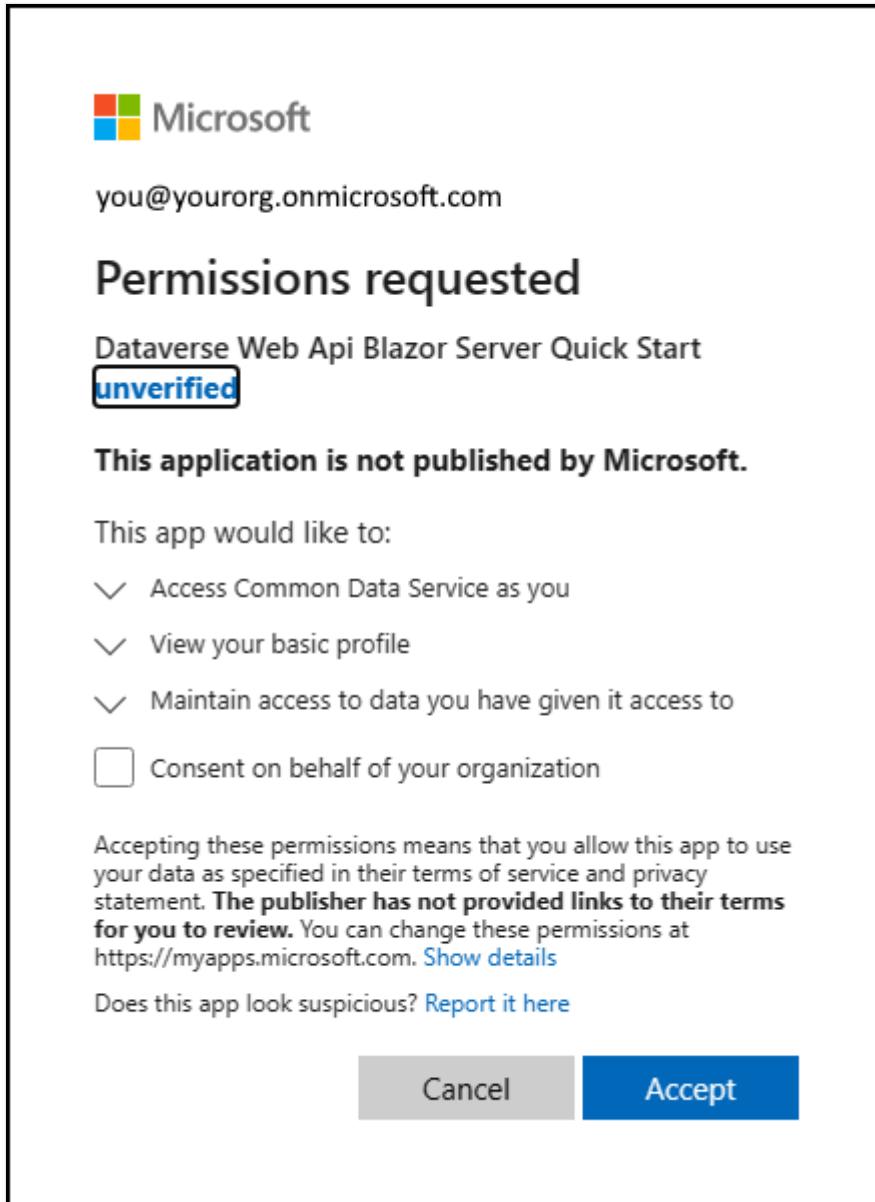
// Used with System.Text.Json.JsonSerializer.Deserialize to deserialize
response body
public class WhoAmIResponse
{
    public Guid BusinessUnitId { get; set; }
    public Guid OrganizationId { get; set; }
    public Guid UserId { get; set; }
}
}

```

Ejecutar el programa

La aplicación ya está lista.

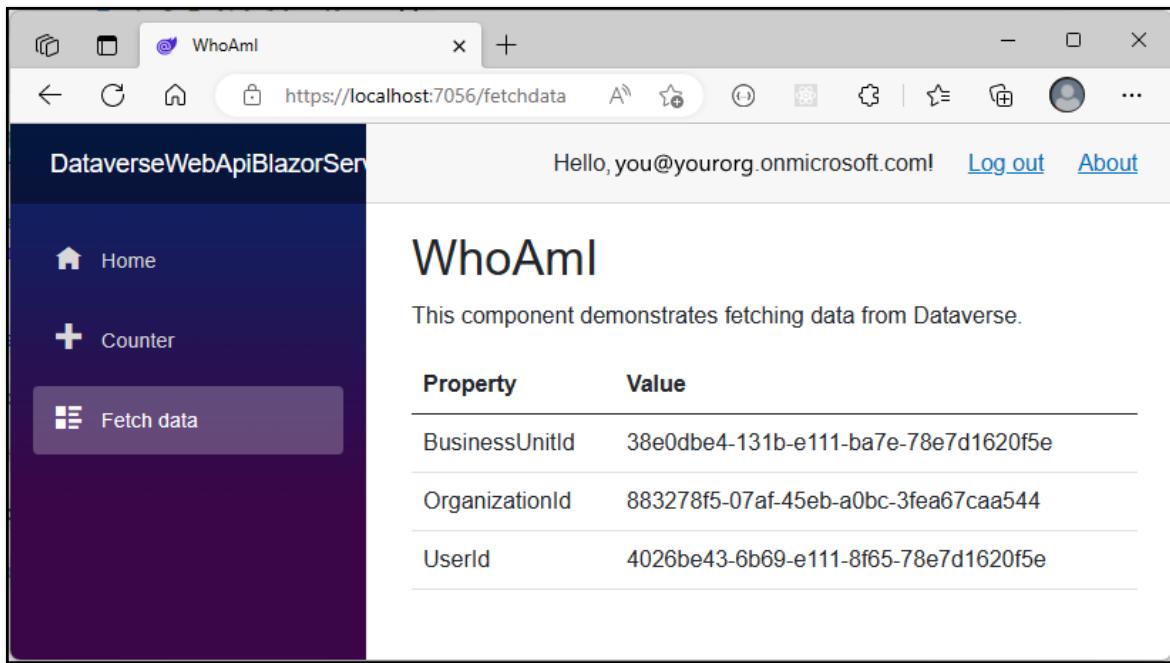
1. Pulse F5 para ejecutar el programa. La primera vez que se ejecute el programa, debería ver este cuadro de diálogo de consentimiento:



2. Seleccione Aceptar.

3. Seleccione Obtener datos.

Los resultados deberán tener un aspecto similar al siguiente:



Enhorabuena. Se ha conectado con éxito a la API web.

Este inicio rápido muestra un enfoque simple para crear una aplicación web de Blazor Server que se conecta a los datos en Dataverse.

Aprender más acerca de las Capacidades de API web de Dataverse mediante la comprensión de los documentos de servicio.

Tipos y operaciones de API web

Consulte también

[Tutorial: crear una aplicación ASP.NET Core Blazor WebAssembly con Dataverse](#)

[Tutorial: Cree una aplicación Blazor Server que use la plataforma de identidad de Microsoft para la autenticación](#)

ⓘ Nota

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta. ↗ (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales (declaración de privacidad ↗).

Autenticarse en Microsoft Dataverse con la API web

Artículo • 09/04/2023

Debe usar OAuth como se describe en [Usar OAuth con Dataverse](#).

El código que escribe para administrar la autenticación cuando usa la API web dependen del tipo de implementación y dónde está su código.

Autenticarse con JavaScript en recursos web

Cuando usa la API web con JavaScript en recursos web HTML, los scripts de formularios o los comandos de la cinta de opciones no necesita incluir ningún código para autenticación. En cada uno de estos casos la aplicación ya autentica al usuario y administra la autenticación.

Si está creando una aplicación de una sola página (SPA) mediante JavaScript puede usar la biblioteca msal.js como se explica en [Usar OAuth con uso compartido de recursos entre orígenes para conectar una aplicación de una sola página](#).

Consultar también

[Usar la API web de Dataverse](#)

[Tipos y operaciones de API web](#)

[Realizar operaciones mediante la API web](#)

[Utilizar OAuth con Dataverse](#)

[Usar OAuth con uso compartido de recursos de origen cruzado para conectar una aplicación de una sola página](#)

ⓘ Nota

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta. [↗](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad ↗](#)).

Tipos y operaciones de API web

Artículo • 09/04/2023

Los temas de esta sección le ayudarán a entender la API web de Dataverse ya que presentan conceptos importantes y le muestran formas de obtener la información que necesitará a través de los documentos de servicio.

Al igual que la [API de Microsoft Graph](#), la API de Dataverse es un servicio RESTful de OData. A diferencia de la API de Microsoft Graph, cada entorno tiene una dirección URL de organización diferente e incluye tablas y operaciones personalizadas que pueden ser exclusivas de cada entorno. Cada entorno es diferente debido al conjunto único de personalizaciones o soluciones instaladas.

Primeros pasos

Lo primero que debe hacer es buscar en los documentos de servicio un entorno de Dataverse al que puede acceder. Para ello necesitará conocer la URL de su entorno. Luego, le recomendamos que configure un entorno Postman para administrar la autenticación en el servicio para ver los documentos del servicio.

Encuentre la URL del extremo de la API web

Use las instrucciones en [Ver recursos para desarrolladores](#) para identificar un extremo de la API web para un entorno al que pueda acceder. Debería tener aspecto similar a éste:

`https://yourorg.api.crm.dynamics.com/api/data/v9.2`

Configurar un entorno de Postman

Debido a que debe autenticarse para ver los datos sobre documentos de servicio, el [cliente API de Postman](#) proporciona una excelente manera de conectarse a sus entornos y explorar la API web de Dataverse. Postman puede simplificar en gran medida la administración de obtener un token de acceso utilizando la API web de Dataverse.

Utilice los pasos en [Configurar un entorno Postman](#) y vea este vídeo [Empezar a usar Postman con la API web Microsoft Dataverse](#).

Con el cliente API de Postman, puede recuperar fácilmente los documentos de servicio y ver ejemplos que muestran cómo la información de los temas de esta sección se aplica a su entorno. El cliente API de Postman también le ayuda a realizar cualquier otro tipo de operación para probar las funciones de la API web.

Estándar de OData v4.0

Los temas de esta sección resumen la información relevante para la implementación de la API web de Dataverse del estándar OData v4.0. Un servicio que implementa OData debe seguir los estándares establecidos en la especificación, pero no está obligado a implementar todas las partes de la especificación.

Para obtener más información, consulte estos documentos:

- [OData Version 4.0. Part 1: Protocol Plus Errata 03 ↗](#)
- [OData Version 4.0. Part 2: URL Conventions Plus Errata 03 ↗](#)
- [OData Version 4.0. Part 3: Common Schema Definition Language \(CSDL\) Plus Errata 03 ↗](#)
- [OData JSON Format Version 4.0 ↗](#)

Pasos siguientes

Obtenga información sobre los documentos de servicio de la API web.

[Documentos de servicio](#)

Consulte también

- [Documentos de servicio de la API web](#)
- [EntityTypes de la API web](#)
- [Propiedades de API web](#)
- [Propiedades de navegación de la API web](#)
- [Acciones de la API web](#)
- [Funciones de la API web](#)
- [Tipos complejos y de enumeración de API web](#)
- [Usar la API web de Dataverse](#)
- [Autenticarse en Dataverse con la API web](#)
- [Realizar operaciones mediante la API web](#)

Nota

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta. [↗](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad ↗](#)).

Documentos de servicio de la API web

Artículo • 09/04/2023

Los extremos de OData proporcionan documentos de servicio que describen las funciones del servicio. Comprender estos documentos de servicio le ayudará a utilizar los recursos que están disponibles en el entorno en el que está trabajando.

Documento de servicio

Basta con realizar una solicitud `GET` en el extremo de la API web para ver el documento de servicio para su entorno. Si no ha configurado un entorno Postman, bastará con pegar al extremo de la API web en el navegador.

El extremo de la API web tendrá un aspecto similar al siguiente:

`https://yourorg.api.crm.dynamics.com/api/data/v9.2/.`

ⓘ Nota

Si está utilizando el entorno de Postman recomendado, simplemente use la variable de entorno `{{webapiurl}}`. Para obtener más información: [Configure un entorno Postman](#).

Si no está utilizando Postman, puede instalar una extensión de navegador para formatear el JSON devuelto y facilitar su lectura. Si está usando Microsoft Edge, busque extensiones con el [formateador de JSON](#).

Debería ver resultados como este:

```
JSON

{
  "@odata.context": "https://yourorg.api.crm.dynamics.com/api/data/v9.2/$metadata",
  "value": [
    {
      "name": "accountleadscollection",
      "kind": "EntitySet",
      "url": "accountleadscollection"
    },
    {
      "name": "accounts",
      "kind": "EntitySet",
      "url": "accounts"
    }
  ]
}
```

```
 },  
 ...
```

El documento de servicio proporciona una lista de todos los *EntitySets* disponibles en su entorno. Un *EntitySet* es el nombre de un recurso que hace referencia a una tabla en Dataverse. Utilizará el nombre del conjunto de entidades en la URL para realizar operaciones en los datos de una tabla específica.

💡 Sugerencia

Use `Ctrl+F` en los resultados que obtiene para encontrar el nombre correcto del conjunto de entidades.

Nombre del conjunto de entidades

Utilice siempre el nombre del conjunto de entidades en lugar del nombre de la colección lógica.

ⓘ Nota

De manera predeterminada, el nombre del conjunto de entidades coincide con el valor de propiedad `EntityMetadata.LogicalCollectionName` de la tabla, pero no debería depender de esto.

Si tiene una tabla personalizada a la que desea dirigirse con otro nombre de conjunto de entidades, puede actualizar el valor de propiedad `EntityMetadata.EntitySetName` de la tabla para usar un nombre diferente. Solo debe cambiar el `EntitySetName` de una tabla cuando la crea y antes de que se escriba cualquier código usando la tabla.

Documento \$metadata de CSDL

Adjuntar `$metadata` al extremo de la API web para recuperar el documento `$metadata` de Common Schema Definition Language (CSDL).

Por ejemplo: [https://yourorg.api.crm.dynamics.com/api/data/v9.2/\\$metadata](https://yourorg.api.crm.dynamics.com/api/data/v9.2/$metadata)

Este documento XML describe todas las tablas y operaciones que puede utilizar en su entorno.

ⓘ Importante

Este documento es el origen central para todo lo relacionado con la API web. Se recomienda hacer referencia al mismo con frecuencia. Use `Ctrl+F` en este documento para ubicar las `EntityType`, `Action`, `Function`, `ComplexType`, o `EnumType` específicas que utilizará. Los nombres distinguen mayúsculas de minúsculas.

Anotaciones de metadatos

Para obtener aún más información de `$metadata`, agregue `?annotations=true` a la URL.

Por ejemplo: `https://yourorg.api.crm.dynamics.com/api/data/v9.2/$metadata?annotations=true`

Esto incluirá muchos tipos diferentes de anotaciones que pueden ser útiles. La mayoría de las anotaciones no se incluyen de forma predeterminada porque aumentan el tamaño total del documento.

Estas anotaciones también se pueden devolver agregando el encabezado de solicitud `Prefer: odata.include-annotations="*"`. Este encabezado de solicitud también funciona para otros tipos de solicitudes. El parámetro de consulta `annotations=true` solo funciona para el documento `$metadata`.

Espacio de nombres de servicio

Cerca de la parte superior del documento `$metadata`, encontrará este elemento XML:

XML

```
<edmx:DataServices>
  <Schema Namespace="Microsoft.Dynamics.CRM" Alias="mscrm"
    xmlns="http://docs.oasis-open.org/odata/ns/edm">
```

Esto le informa que todos los elementos en el servicio que están dentro del espacio de nombres `Microsoft.Dynamics.CRM` y que `mscrm` es el alias para el espacio de nombres. En algunas situaciones, deberá usar el nombre completo de un objeto, por lo que será necesario usar el valor del espacio de nombres.

Pasos siguientes

Tipos de entidad

Consulte también

[Tipos y operaciones de API web](#)

[EntityTypes de la API web](#)

[Propiedades de API web](#)

[Propiedades de navegación de la API web](#)

[Acciones de la API web](#)

[Funciones de la API web](#)

[Tipos complejos y de enumeración de API web](#)

[Usar la API web de Dataverse](#)

ⓘ Nota

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta. [↗](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad ↗](#)).

EntityTypes de la API web

Artículo • 25/08/2023

En el documento [\\$metadata de CSDL](#), inmediatamente debajo del [espacio de nombres de servicio](#) hay una lista de *EntityTypes*. Un tipo de entidad es un tipo estructurado con nombre con una clave. Define las propiedades y las relaciones con nombre de una tabla.

Los elementos `EntityType` suelen tener los atributos siguientes:

Attribute	Descripción
<code>Name</code>	El nombre del tipo, el LogicalName de la tabla.
<code> BaseType</code>	El <code>EntityType</code> del que se hereda el tipo de entidad.

Por ejemplo, este es el elemento XML es el `EntityType` para la entidad `account`, excluyendo propiedades y propiedades de navegación.

XML
<pre><EntityType Name="account" BaseType="mscrm.crmbaseentity"> <Key> <PropertyRef Name="accountid" /> <!--The name of the primary key --> </Key> <!--Properties and navigation properties removed for brevity--> <Annotation Term="Org.OData.Core.V1.Description" String="Business that represents a customer or potential customer. The company that is billed in business transactions." /> </EntityType></pre>

Excepto por tres excepciones, todos los tipos de entidades tienen los siguientes elementos secundarios:

Elemento	Descripción
<code>Key</code>	Contiene un elemento <code><PropertyRef></code> donde el atributo <code>Name</code> representa la clave principal de la tabla.
<code>Property</code>	Contiene detalles sobre una propiedad de <code>EntityType</code> . Consulte Propiedades de API web .
<code>NavigationProperty</code>	Contiene detalles sobre una relación con este <code>EntityType</code> . Consulte Propiedades de navegación de la API web

Tipos de entidad especiales

Hay tres tipos de entidades que no tienen los elementos `Key`, `Property` o `NavigationProperty`.

crmbaseentity

Este elemento define un tipo abstracto común para cualquier tabla que contienen datos comerciales.

```
<EntityType Name="crmbaseentity" Abstract="true" />
```

Debido a que todos los tipos de entidad que contienen datos comerciales heredan de `crmbaseentity`, `crmbaseentity` es referenciado cuando un valor no es específico de una tabla.

expando

Este elemento define un tipo de entidad que hereda de `crmbaseentity`, pero también es un [OpenType de OData ↗](#).

```
<EntityType Name="expando" BaseType="mscrm.crmbaseentity" OpenType="true" />
```

Un tipo de entidad expando se puede utilizar como parámetro para una acción o como propiedad de respuesta desde una función o acción.

Más información: [Usar tipos abiertos con API personalizadas](#)

crmmodebaseentity

Este elemento existe cerca de la parte inferior del documento \$metadata:

```
<EntityType Name="crmmodebaseentity" Abstract="true" />
```

Este elemento define un tipo abstracto común para cualquier definición de esquema. Es el tipo básico para otra clase básica abstracta utilizada para definiciones de tablas. A menos que desee crear y modificar tablas, columnas y relaciones usando la API web, no necesitará usar tipos de entidad que hereden de este tipo. Más información: [Use la API web con definiciones de tabla](#).

Herencia de EntityType

Para los datos comerciales, hay dos tipos de entidades abstractas más que se heredan de `crmbaseentity`:

EntityType	Descripción
<code>principal</code>	<p>Los tipos de entidad <code>systemuser</code> y <code>team</code> heredan del tipo de entidad <code>principal</code>. La entidad de seguridad solo proporciona la propiedad <code>ownerid</code>, que tiene cada tabla propiedad de un usuario. Esta herencia es lo que permite que los registros propiedad del usuario se asignen a un usuario o a un equipo.</p>
	<p>La propiedad <code>ownerid</code> es la clave principal para los EntityTypes <code>systemuser</code> y <code>team</code>.</p>
<code>activitypointer</code>	<p>Cualquier tabla que esté configurada como una actividad heredada del tipo de entidad <code>activitypointer</code>. Este tipo proporciona propiedades comunes que se encuentran en tipos de entidad como: <code>appointment</code>, <code>email</code>, <code>fax</code>, <code>letter</code>, <code>phonecall</code> y <code>task</code>. También puede crear una tabla personalizada que represente una actividad. Estas propiedades comunes hacen posible recuperar una lista de actividades de diferentes tipos usando estas propiedades comunes</p>
	<p>La propiedad <code>activityid</code> es la clave principal para todos los tipos de entidad que heredan de <code>activitypointer</code>.</p>

Cuando se trabaja con definiciones de tablas, existe otra jerarquía de herencia. El tipo de entidad `MetadataBase` hereda del resumen `crmmodebaseentity` para proporcionar las propiedades `MetadataId` y `HasChanged` comunes. Más información: [Use la API web con definiciones de tabla](#).

Claves alternativas

Cuando un tipo de entidad tiene definidas claves alternativas, hay una `Annotation` que describe las propiedades que participan en la definición de la clave alternativa.

El siguiente ejemplo muestra la anotación cuando la entidad `account` ha sido configurada para habilitar la propiedad `accountnumber` como clave alternativa.

XML

```

<Annotation Term="OData.Community.Keys.V1.AlternateKeys">
  <Collection>
    <Record Type="OData.Community.Keys.V1.AlternateKey">
      <PropertyValue Property="Key">
        <Collection>
          <Record Type="OData.Community.Keys.V1.PropertyRef">
            <PropertyValue Property="Alias" String="accountnumber" />
            <PropertyValue Property="Name" PropertyPath="accountnumber" />
          </Record>
        </Collection>
      </PropertyValue>
    </Record>
  </Collection>
</Annotation>

```

```
</PropertyValue>
</Record>
</Collection>
</Annotation>
```

Más información: [Recuperar registro mediante una clave alternativa](#)

Pasos siguientes

Más información sobre las propiedades.

[Propiedades](#)

Consulte también

[Tipos y operaciones de API web](#)

[Documentos de servicio de la API web](#)

[Propiedades de API web](#)

[Propiedades de navegación de la API web](#)

[Acciones de la API web](#)

[Funciones de la API web](#)

[Tipos complejos y de enumeración de API web](#)

[Usar la API web de Dataverse](#)

Nota

¿Puede indicarnos sus preferencias de idioma de documentación? [Realice una breve encuesta.](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad](#)).

Propiedades de API web

Artículo • 25/08/2023

En el [documento \\$metadata de CSDL](#), cada tipo de entidad que no sea abstracta tiene elementos `Property`. Cada elemento de `Property` tiene los siguientes atributos:

Attribute	Descripción
<code>Name</code>	El nombre de la propiedad; normalmente el LogicalName de la columna y siempre en minúsculas. Una excepción a esta regla son las propiedades de búsqueda .
<code>Type</code>	El tipo primitivo de la propiedad. Vea Tipos primitivos utilizados por Dataverse

Cuando recupere el documento `$metadata` con [Anotaciones de metadatos](#) puede ver algunos elementos `Annotation` que pueden resultar útiles. Por ejemplo, las siguientes propiedades `name` y `openrevenue`:

XML
<pre><Property Name="name" Type="Edm.String" Unicode="false"> <Annotation Term="Org.OData.Core.V1.Description" String="Type the company or business name." /> </Property> <Property Name="openrevenue" Type="Edm.Decimal" Scale="Variable"> <Annotation Term="Org.OData.Core.V1.Description" String="Sum of open revenue against an account and its child accounts." /> <Annotation Term="Org.OData.Core.V1.Computed" Bool="true" /> <Annotation Term="Org.OData.Core.V1.Permissions"> <EnumMember>Org.OData.Core.V1.PermissionType/Read</EnumMember> </Annotation> </Property></pre>

Con las anotaciones en este ejemplo, puede saber que el campo `openrevenue` es de solo lectura.

La siguiente tabla describe las anotaciones que se devuelven con [anotaciones de metadatos](#) aplicadas.

Vigencia de la anotación	Descripción
<code>Org.OData.Core.V1.Description</code>	Una descripción de la propiedad
<code>Org.OData.Core.V1.Computed</code>	Si se computa la propiedad; solo lectura porque lo calcula Dataverse.

Vigencia de la anotación	Descripción
Org.OData.Core.V1.Permissions	Incluye los tipos de permisos disponibles para la propiedad. Solo se incluye cuando los permisos son limitados y siempre contiene el valor <EnumMember>Org.OData.Core.V1.PermissionType/Read</EnumMember> lo que significa que es de solo lectura.

ⓘ Nota

Las anotaciones `Org.OData.Core.V1.Computed` y `Org.OData.Core.V1.Permissions` se encuentran frecuentemente juntas, pero no siempre. Cualquiera de ellas significa que la propiedad es de solo lectura.

Tipos primitivos que usa Dataverse

Las propiedades de los tipos de entidad que se usan para los datos profesionales solo incluirán solo tipos primitivos. La siguiente tabla describe los tipos primitivos que usa Dataverse.

Type	Descripción
<code>Edm.Binary</code>	Datos binarios. Se utiliza para los datos de archivos e imágenes.
<code>Edm.Boolean</code>	Lógica de valor binario.
<code>Edm.Date</code>	Fecha sin desplazamiento de zona horaria.
<code>Edm.DateTimeOffset</code>	Fecha y hora con un desplazamiento de zona horaria, no hay segundos bisiestos
<code>Edm.Decimal</code>	Valores numéricos con precisión y escala fijas
<code>Edm.Double</code>	Número de coma flotante binaryIEEE 754 binary64 (15-17 dígitos decimales)
<code>Edm.Guid</code>	Identificador único de 16 bytes (128 bits)
<code>Edm.Int16</code>	Entero de 16 bits firmado Se usa solo para ciertas propiedades de EntityType de esquema
<code>Edm.Int32</code>	Entero de 32 bits firmado
<code>Edm.Int64</code>	Entero de 64 bits firmado

Type	Descripción
Edm.String	<p>Secuencia de caracteres UTF-8 Este tipo puede incluir la faceta <code>Unicode="false"</code>.</p> <p>Cuando existe esta faceta, la propiedad de cadena solo contiene y acepta valores de cadena con caracteres limitados para el juego de caracteres ASCII.</p> <p>Si no se especifica ningún valor, el valor predeterminado de la faceta Unicode es verdadero.</p>

Propiedades de búsqueda

Algunas propiedades utilizan la siguiente convención de nomenclatura: `_<name>_value`.

Estas se denominan *propiedades de búsqueda*. Las propiedades de búsqueda son propiedades calculadas de solo lectura que tienen un valor `Edm.Guid`. Puede utilizar estas propiedades de búsqueda en un filtro de consulta para hacer coincidir todos los registros asociados con el único registro en la relación de varios a uno.

OData representa las columnas de búsqueda como [Propiedades de navegación de valor único](#) en lugar de propiedades. Dataverse proporciona propiedades de búsqueda para simplificar escenarios para recuperar datos utilizando la clave principal de un registro relacionado. Para cambiar el valor de una propiedad de búsqueda, debe cambiar la propiedad de navegación de un solo valor que representa. Más información: [Propiedades de búsqueda](#)

Al incluir propiedades de búsqueda en una consulta, puede solicitar la inclusión de anotaciones que proporcionen más información sobre los datos que se establecen para estos atributos subyacentes que no están representados por una propiedad de navegación de un solo valor. Más información: [Datos de propiedades de búsqueda](#)

Pasos siguientes

Obtenga más información sobre las propiedades de navegación.

[Propiedades de navegación](#)

Consulte también

[Usar la API web de Dataverse](#)

[Tipos y operaciones de API web](#)

Documentos de servicio de la API web

EntityTypes de la API web

ⓘ Nota

¿Puede indicarnos sus preferencias de idioma de documentación? **Realice una breve encuesta.** (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales (**declaración de privacidad**).

Propiedades de navegación de la API web

Artículo • 24/08/2023

En el [documento \\$metadata de CSDL](#), cada tipo de entidad que no sea abstracta tiene elementos `NavigationProperty`. Los elementos `NavigationProperty` describen los datos relacionados con el tipo de entidad actual. Cuando recupere un registro, puede expandir las propiedades de navegación para incluir los datos relacionados.

La siguiente tabla describe los atributos de los elementos `NavigationProperty`.

Attribute	Descripción
<code>Name</code>	El nombre de la propiedad de navegación; distingue mayúsculas de minúsculas
<code>Type</code>	El tipo de entidad realcionado; puede ser para un solo valor o una colección de un tipo
<code>Partner</code>	El nombre de la propiedad de navegación en el otro lado de la relación
<code>Nullable="false"</code>	Si el valor puede ser NULL

ⓘ Importante

Hay dos tipos de propiedades de navegación: **un solo valor** y **valoradas como colección**. Esta distinción es importante porque las funciones de cada tipo de propiedad de navegación son diferentes.

Propiedades de navegación de un solo valor

Cuando una propiedad de navegación `Type` se refiere a un solo valor, representa una relación de uno a varios que crea una referencia a otro registro de la tabla. Esta relación habitualmente se denomina *búsqueda*. El siguiente ejemplo es la propiedad de navegación `account` tabla `createdby`:

XML

```
<NavigationProperty  
    Name="createdby"  
    Type="mscrm.systemuser"  
    Nullable="false"
```

```

    Partner="lk_accountbase_createdby">
    <ReferentialConstraint
        Property="_createdby_value"
        ReferencedProperty="systemuserid" />
</NavigationProperty>

```

Esta propiedad de navegación de un solo valor conecta múltiples registros `account` a un solo registro `systemuser`. Cada registro `systemuser` tiene una propiedad de navegación valorada como colección denominada `lk_accountbase_createdby` que lo conecta con los registros `account` que el usuario ha creado.

Estos valores se almacenan en definiciones de relaciones. Puede acceder a ellos utilizando el tipo de entidad SDK [OneToManyRelationshipMetadata](#) o Web API [OneToManyRelationshipMetadata](#), como se describe en la siguiente tabla.

Attribute	Propiedad OneToManyRelationshipMetadata	Descripción
<code>Name</code>	<code>ReferencingEntityPropertyName</code>	El nombre de la propiedad de navegación de un solo valor.
<code>Partner</code>	<code>ReferencedEntityPropertyName</code>	El nombre de la colección de navegación de un solo valor.

Propiedades de búsqueda

Las propiedades de navegación de un solo valor tienen un atributo `ReferentialConstraint` con un atributo `Property` que hace referencia a una propiedad de búsqueda. Las propiedades de búsqueda utilizan la siguiente convención de nomenclatura: `_<name>_value`. [Obtenga más información acerca de las propiedades de búsqueda.](#)

El atributo `ReferentialConstraint` tiene un atributo `ReferencedProperty` que identifica el nombre de la clave principal del tipo de entidad relacionada.

En la mayoría de los casos el `<name>` encontrado en la propiedad de búsqueda coincide con el nombre de la propiedad de navegación, excepto cuando la propiedad de navegación de un valor representa una búsqueda en varias tablas.

Búsquedas en varias tablas

Cuando la propiedad de navegación de un solo valor forma parte de una búsqueda de varias tablas, o polimórfica, hay una única propiedad de búsqueda que es

`ReferentialConstraint` para más de una propiedad de navegación de un solo valor.

Un tipo de entidad puede tener algo como la siguiente combinación, donde una sola propiedad de búsqueda `_customerid_value` admite varias propiedades de navegación de un solo valor que representan una búsqueda en varias tablas. Hay una propiedad de navegación de un solo valor para cada tipo de tabla admitida por la búsqueda de varias tablas.

XML

```
<EntityType
  Name="socialprofile"
  BaseType="mscrm.crmbaseentity">
  <Key>
    <PropertyRef Name="socialprofileid" />
  </Key>
  <Property
    Name="_customerid_value" <!-- lookup property -->
    Type="Edm.Guid">
    <Annotation
      Term="Org.OData.Core.V1.Description"
      String="Shows the customer that this social profile belongs to."
    />
    </Property>
    <NavigationProperty
      Name="customerid_contact" <!-- Name different from lookup property - ->
      Type="mscrm.contact"
      Nullable="false"
      Partner="Socialprofile_customer_contacts">
      <ReferentialConstraint
        Property="_customerid_value" <!-- Reference to lookup property -->
        ReferencedProperty="contactid" />
    </NavigationProperty>
    <NavigationProperty
      Name="customerid_account" <!-- Name different from lookup property - ->
      Type="mscrm.account"
      Nullable="false"
      Partner="Socialprofile_customer_accounts">
      <ReferentialConstraint
        Property="_customerid_value" <!-- Reference to lookup property -->
        ReferencedProperty="accountid" />
    </NavigationProperty>
  </EntityType>
```

En estos casos, establecer el valor de cualquiera de las propiedades de navegación de un solo valor establece todas el resto de propiedades de navegación de un solo valor

participantes en NULL. El valor GUID de la propiedad de búsqueda correspondiente cambia, pero debe recuperar las anotaciones específicas disponibles para saber a qué tabla se refiere ahora. Más información: [Datos de propiedades de búsqueda](#)

Propiedades de navegación valoradas como colección

Cuando una propiedad de navegación `Type` se refiere a un valor de colección, representa una relación de varios a uno o de varios a varios. El ejemplo siguiente es la propiedad de navegación `Account_Tasks` de la entidad de cuenta:

XML

```
<NavigationProperty
    Name="Account_Tasks"
    Type="Collection(mscrm.task)"
    Partner="regardingobjectid_account_task"
/>
```

Esta propiedad de navegación conecta un registro `account` a varios registros `task`. Cada `task` tiene una propiedad de navegación de un solo valor llamada `regardingobjectid_account_task` que se refiere a la `account` como el objeto referente.

La forma en que trabaja con las propiedades de navegación valoradas como colección mediante OData es la misma independientemente de si la relación es de uno a varios o de varios a varios. Ambas se consideran colecciones y se interactúa con ellas de la misma manera.

Relaciones de varios a uno

Una relación de varios a uno es una imagen reflejada de una relación de uno a varios. Tiene una propiedad de navegación de un solo valor de partner. En el ejemplo [Propiedades de navegación de un solo valor](#) anterior, vimos la propiedad `createdby` de navegación de un solo valor para el tipo de entidad `account`.

En el tipo de entidad `systemuser`, existe el partner de la propiedad de navegación valorada como colección denominada `lk_accountbase_createdby`.

XML

```
<NavigationProperty Name="lk_accountbase_createdby"
    Type="Collection(mscrm.account)"
```

```
Partner="createdby" />
```

Relaciones de varios a varios

Cuando `Name` y `Partner` de la propiedad de navegación valorada como colección son iguales, representa una relación de varios a varios.

Las relaciones varios a varios tiene algunos detalles de implementación que puede encontrar en los documentos de servicio. Para la mayoría de los casos de uso, puede ignorarlos.

Por ejemplo, cada relación de varios a varios tiene una *tabla de intersección* que la respalda. Estas tablas de intersección tienen tipos de entidad que normalmente tienen solo cuatro propiedades de solo lectura. En el siguiente ejemplo, el tipo de entidad `teammembership` es una tabla de intersección que admite una relación de varios a varios entre tipos de entidad `systemuser` y `team`:

XML

```
<EntityType Name="teammembership"
    BaseType="mscrm.crmbaseentity">
    <Key>
        <PropertyRef Name="teammembershipid" />
    </Key>
    <Property Name="systemuserid"
        Type="Edm.Guid" />
    <Property Name="versionnumber"
        Type="Edm.Int64" />
    <Property Name="teammembershipid"
        Type="Edm.Guid" />
    <Property Name="teamid"
        Type="Edm.Guid" />
</EntityType>
```

No puede trabajar con tipos de entidad que representan tablas de intersección directamente porque todas las propiedades son de solo lectura. Realice operaciones en las respectivas propiedades de navegación valoradas como colección para cada tipo de entidad. Más información: [Asociar y anular la asociación de filas de tabla usando la API web](#).

Para esta relación de varios a varios, el tipo de entidad `systemuser` tiene la siguiente propiedad de navegación valorada como colección:

XML

```
<NavigationProperty Name="teammembership_association"  
Type="Collection(mscrm.team)"  
Partner="teammembership_association" />
```

El tipo de entidad `team` tiene esta propiedad de navegación valorada como colección:

XML

```
<NavigationProperty Name="teammembership_association"  
Type="Collection(mscrm.systemuser)"  
Partner="teammembership_association" />
```

Estos valores se almacenan en definiciones de relaciones. Puede acceder a ellos utilizando el tipo de entidad SDK [ManyToManyRelationshipMetadata](#) o Web API [ManyToManyRelationshipMetadata](#), como se describe en la siguiente tabla.

Attribute	Propiedad	Descripción
ManyToManyRelationshipMetadata		
<code>Name</code>	<code>Entity1NavigationPropertyName</code>	El nombre de la propiedad de navegación valorada como colección para uno de los tipos de entidad
<code>Partner</code>	<code>Entity2NavigationPropertyName</code>	El nombre de la propiedad de navegación valorada como colección para el otro tipo de entidad

Pasos siguientes

Obtenga más información sobre las definiciones de acción.

[Definiciones de acción](#)

Consulte también

[Tipos y operaciones de API web](#)

[Documentos de servicio de la API web](#)

[EntityTypes de la API web](#)

[Propiedades de API web](#)

[Acciones de la API web](#)

[Funciones de la API web](#)

[Tipos complejos y de enumeración de API web](#)

Usar la API web de Dataverse

OData Version 4.0. Part 3: Common Schema Definition Language (CSDL) Plus Errata 03

7.1 Element `edm:NavigationProperty` ↗

ⓘ Nota

¿Puede indicarnos sus preferencias de idioma de documentación? **Realice una breve encuesta.** ↗ (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad](#) ↗).

Acciones de la API web

Artículo • 09/04/2023

En el [documento \\$metadata de CSDL](#), encontrará elementos `Action` que describen operaciones que modifican datos.

Toda acción tiene un atributo `Name`. Este nombre formará parte de la URL cuando utilice `HTTP``POST`, enviando cualquier parámetro definido para la acción en el cuerpo de la solicitud. Más información: [Usar acciones web API](#).

Acciones enlazadas

Las acciones pueden tener un atributo `IsBound="true"`. Esto significa que es una acción *enlazada*. Las acciones sin el atributo `IsBound` están *sin enlazar*. Una acción enlazada significa que el primer parámetro es una referencia a un registro de tabla o a un conjunto de entidades.

Cuando se enlaza una acción, tendrá una referencia a un elemento específico dentro del espacio de nombres del servicio. Para utilizar la acción, debe utilizar el nombre completo, incluido el espacio de nombres `Microsoft.Dynamics.CRM`. Más información: [Espacio de nombres de servicio](#).

Parámetros

Las acciones suelen tener uno o más elementos `Parameter`. Cada parámetro tendrá los siguientes atributos:

Atributo	Descripción
<code>Name</code>	El nombre del parámetro. El nombre es único a menos que <code>Action</code> esté sobrecargado. Para obtener más información: consulte Acciones sobrecargadas .
<code>Type</code>	El tipo del parámetro.
<code>Nullable="false"</code>	Si el parámetro puede aceptar un valor NULL.

ReturnType

Las acciones pueden devolver valores. Cuando una acción devuelve un valor, tendrá un elemento `ReturnType`.

Atributo	Descripción
<code>Type</code>	El tipo del parámetro.
<code>Nullable="false"</code>	Si el valor puede ser NULL.

Ejemplos de definición de la acción

Los ejemplos son casos ficticios de definiciones de `Action` que muestran diferentes patrones de enlace. Cada uno de estos ejemplos devuelve un valor entero.

Acciones sin enlazar

Una acción independiente con un solo parámetro `Number` entero.

XML

```
<Action Name="UnBoundActionExample">
    <Parameter Name="Number" Type="Edm.Int32" Nullable="false" />
    <ReturnType Type="Edm.Int32" Nullable="false" />
</Action>
```

Acción enlazada a una entidad

Una acción enlazada a la entidad `account` con un parámetro `Number` entero.

XML

```
<Action Name="EntityBoundActionExample" IsBound="true">
    <Parameter Name="entity" Type="mscrm.account" Nullable="false" />
    <Parameter Name="Number" Type="Edm.Int32" Nullable="false" />
    <ReturnType Type="Edm.Int32" Nullable="false" />
</Action>
```

Acción enlazada a un conjunto de entidades

Una acción enlazada al EntitySet de `account` con un parámetro `Number` entero.

XML

```
<Action Name="EntityCollectionBoundActionExample" IsBound="true">
  <Parameter Name="entityset" Type="Collection(mscrm.account)"
    Nullable="false" />
  <Parameter Name="Number" Type="Edm.Int32" Nullable="false" />
  <ReturnType Type="Edm.Int32" Nullable="false" />
</Action>
```

Acciones sobrecargadas

Por lo general, cada acción que encuentre en \$metadata será la única acción con ese nombre. Sin embargo, las acciones enlazadas pueden tener múltiples definiciones cuando están enlazadas a diferentes tipos. La acción `AddItemCampaign` incluida en la solución de marketing es un ejemplo. No puede crear una acción sobrecargada usando la [API personalizada](#).

Para obtener más información: [Reglas de acciones sobrecargadas](#).

Pasos siguientes

Obtenga información sobre las definiciones de funciones.

[Definiciones de funciones](#)

Consulte también

[Tipos y operaciones de API web](#)

[Documentos de servicio de la API web](#)

[EntityTypes de la API web](#)

[Propiedades de API web](#)

[Propiedades de navegación de la API web](#)

[Funciones de la API web](#)

[Tipos complejos y de enumeración de API web](#)

[Usar la API web de Dataverse](#)

[Especificación de la acción OData 4.0](#)

ⓘ Nota

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta. [\(tenga en cuenta que esta encuesta está en inglés\)](#)

La encuesta durará unos siete minutos. No se recopilan datos personales
[\(declaración de privacidad ↗\)](#).

Funciones de la API web

Artículo • 09/04/2023

En el documento [\\$metadata de CSDL](#), encontrará elementos `Function`. Las operaciones de Function son diferentes de las acciones porque no deben cambiar los datos. Por lo general, se utilizan solo para recuperar datos.

Toda función tiene un atributo `Name`. Este nombre formará parte de la URL cuando utilice `HTTP GET`, enviando cualquier parámetro definido para la función en la URL de la solicitud.

Funciones enlazadas

Al igual que las acciones, las funciones pueden tener un atributo `IsBound="true"`. Esto significa que es una función *enlazada*. Las funciones sin el atributo `IsBound` están *sin enlazar*. Una función enlazada significa que el primer parámetro es una referencia a un registro de tabla o a un conjunto de entidades.

Cuando se enlaza una función, tendrá una referencia a un elemento específico dentro del espacio de nombres del servicio. Para utilizar la función, debe utilizar el nombre completo, incluido el espacio de nombres `Microsoft.Dynamics.CRM`. Más información: [Espacio de nombres de servicio](#).

Funciones que admiten composición

Las funciones pueden tener un atributo `IsComposable="true"`. Esto significa que puede agregar algunas opciones de consulta del sistema, como `$filter` o `$select` a la URL para especificar los resultados devueltos. Esta opción solo está disponible para funciones específicas del sistema. No puede crear una función usando la [API personalizada](#) que es componible.

Parámetros

Las funciones suelen tener uno o más elementos `Parameter`. Cada parámetro tendrá los siguientes atributos:

Atributo	Descripción
----------	-------------

Atributo	Descripción
Name	El nombre del parámetro. El nombre es único a menos que <code>Function</code> esté sobrecargado. Para obtener más información: consulte Funciones sobrecargadas .
Type	El tipo del parámetro.
Nullable="false"	Si el parámetro puede aceptar un valor NULL.

ReturnType

Las funciones deben devolver valores. Una función siempre tendrá un elemento `ReturnType`.

Atributo	Descripción
Type	El tipo del parámetro.
Nullable="false"	Si el valor puede ser NULL.

Ejemplos de definición de la función

Los ejemplos son casos ficticios de definiciones de `Function` que muestran diferentes patrones de enlace. Cada uno de estos ejemplos devuelve un valor entero.

Funciones sin enlazar

Una función sin enlazar con un solo parámetro `Number` entero.

XML

```
<Function Name="UnBoundFunctionExample">
    <Parameter Name="Number" Type="Edm.Int32" Nullable="false" />
    <ReturnType Type="Edm.Int32" Nullable="false" />
</Function>
```

Función enlazada a una entidad

Una función enlazada a la entidad `account` con un parámetro `Number` entero.

XML

```
<Function Name="EntityBoundFunctionExample" IsBound="true">
  <Parameter Name="entity" Type="mscrm.account" Nullable="false" />
  <Parameter Name="Number" Type="Edm.Int32" Nullable="false" />
  <ReturnType Type="Edm.Int32" Nullable="false" />
</Function>
```

Función enlazada a un conjunto de entidades

Una función enlazada a EntitySet `account` con un parámetro `Number` entero.

XML

```
<Function Name="EntityCollectionBoundFunctionExample" IsBound="true">
  <Parameter Name="entityset" Type="Collection(mscrm.account)" Nullable="false" />
  <Parameter Name="Number" Type="Edm.Int32" Nullable="false" />
  <ReturnType Type="Edm.Int32" Nullable="false" />
</Function>
```

Acciones sobrecargadas

Por lo general, cada función que encuentre en \$metadata será la única función con ese nombre. Sin embargo, las funciones enlazadas pueden tener múltiples definiciones cuando están enlazadas a diferentes tipos. El sistema [RetrieveUnpublished](#) y las funciones [RetrieveUnpublishedMultiple](#) son algunos ejemplos. No puede crear una función sobrecargada usando la [API personalizada](#).

Para obtener más información: [Reglas de acciones sobrecargadas de la función de OData 4.0 ↗](#).

Pasos siguientes

Más información sobre los tipos complejos y de enumeración.

[Tipos complejos y de enumeración](#)

Consulte también

[Usar la API web de Dataverse](#)

[Tipos y operaciones de API web](#)

[Documentos de servicio de la API web](#)

[EntityTypes de la API web](#)

[Propiedades de API web](#)

[Propiedades de navegación de la API web](#)

[Acciones de la API web](#)

[Tipos complejos y de enumeración de API web](#)

[OData Version 4.0. Part 3: Common Schema Definition Language \(CSDL\) Plus Errata 03](#)

[Element edm:Function ↗](#)

 **Nota**

¿Puede indicarnos sus preferencias de idioma de documentación? [Realice una breve encuesta.](#) ↗ (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad](#) ↗).

Tipos complejos y de enumeración de API web

Artículo • 09/04/2023

En el documento [\\$metadata de CSDL](#), encontrará elementos `ComplexType` y `EnumType`.

Tipos complejos

Los tipos complejos son tipos estructurados con nombre sin clave que constan de un conjunto de propiedades. Los tipos complejos suelen utilizarse como valores de propiedad en definiciones de tabla o como parámetros o valores devueltos para operaciones.

Por ejemplo, [WhoAmI Function](#) devuelve este [WhoAmIResponse ComplexType](#):

XML

```
<ComplexType Name="WhoAmIResponse">
  <Property Name="BusinessUnitId" Type="Edm.Guid" Nullable="false" />
  <Property Name="UserId" Type="Edm.Guid" Nullable="false" />
  <Property Name="OrganizationId" Type="Edm.Guid" Nullable="false" />
</ComplexType>
```

Tipos de enumeración

Los tipos de enumeración son tipos primitivos con nombre cuyos valores son constantes con nombre con valores enteros subyacentes.

Por ejemplo, esta es la definición de [AccessRights EnumType](#)

XML

```
<EnumType Name="AccessRights">
  <Member Name="None" Value="0" />
  <Member Name="ReadAccess" Value="1" />
  <Member Name="WriteAccess" Value="2" />
  <Member Name="AppendAccess" Value="4" />
  <Member Name="AppendToAccess" Value="16" />
  <Member Name="CreateAccess" Value="32" />
  <Member Name="DeleteAccess" Value="65536" />
  <Member Name="ShareAccess" Value="262144" />
  <Member Name="AssignAccess" Value="524288" />
</EnumType>
```

El tipo de enumeración `AccessRights` se utiliza para la propiedad `AccessMask` de `PrincipalAccess ComplexType`, que se utiliza para establecer el parámetro `PrincipalAccess` para `ModifyAccess Action`. Esta es la acción utilizada para cambiar el acceso al compartir un registro.

El siguiente ejemplo otorga derechos de acceso `ReadAccess`, `WriteAccess`, `DeleteAccess`, `AppendAccess`, y `AssignAccess` al registro `account` especificado por el parámetro `Target` al `systemuser` designado por la propiedad `Principal` del tipo complejo `PrincipalAccess`.

Solicitud

HTTP

```
POST [Organization URI]/api/data/v9.0/ModifyAccess
OData-Version: 4.0
OData-MaxVersion: 4.0
Content-Type: application/json; charset=UTF-8
Accept: application/json

{
    "Target": {
        "accountid": "cbcfc8bbc-aa41-ec11-8c62-000d3a53893c",
        "@odata.type": "Microsoft.Dynamics.CRM.account"
    },
    "PrincipalAccess": {
        "Principal": {
            "systemuserid": "8061643d-ebf7-e811-a974-000d3a1e1c9a",
            "@odata.type": "Microsoft.Dynamics.CRM.systemuser"
        },
        "AccessMask":
        "ReadAccess,WriteAccess,DeleteAccess,AppendAccess,AssignAccess"
    }
}
```

Respuesta

HTTP

```
HTTP/1.1 204 No Content
```

Pasos siguientes

Realizar operaciones mediante la API web.

Consulte también

[Tipos y operaciones de API web](#)

[Documentos de servicio de la API web](#)

[EntityTypes de la API web](#)

[Propiedades de API web](#)

[Propiedades de navegación de la API web](#)

[Acciones de la API web](#)

[Funciones de la API web](#)

[Usar la API web de Dataverse](#)

[Tipo complejo OData Versión 4.0 ↗](#)

[Tipo de enumeración OData Versión 4.0 ↗](#)

Nota

¿Puede indicarnos sus preferencias de idioma de documentación? [Realice una breve encuesta.](#) ↗ (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad](#) ↗).

Realizar operaciones mediante la API web

Artículo • 23/06/2023

La API web proporciona una interfaz de servicio web RESTful que puede utilizar para interactuar con datos en Microsoft Dataverse utilizando una amplia variedad de lenguajes de programación y bibliotecas.

💡 Sugerencia

Antes de comenzar, puede que le resulte útil revisar los temas en [Tipos y operaciones de API web](#) para comprender los documentos del servicio OData que brindan información importante sobre lo que puede hacer con la API web.

ⓘ Nota

La información de esta sección también es aplicable a los usuarios de Dynamics 365 Customer Engagement (on-premises).

En esta sección

[Componer solicitudes HTTP y administrar errores](#)

[Consultar datos utilizando la API web](#)

[Recuperar y ejecutar consultas predefinidas](#)

[Crear una fila de tabla usando la API web](#)

[Recuperar una fila de tabla usando la API web](#)

[Actualizar y eliminar filas de tablas usando la API web](#)

[Asociar y anular la asociación de filas de tabla mediante la API web](#)

[Usar funciones de la API web](#)

[Usar acciones de la API web](#)

[Ejecutar las operaciones por lotes mediante API web](#)

[Suplantar a otro usuario utilizando la API web](#)

[Realizar operaciones condicionales mediante la API web](#)

[Administrar la detección de duplicados durante las operaciones de creación y actualización](#)

Consultar también

[Usar la API web de Dataverse](#)

[Autenticarse en Dataverse con la API web](#)

[Tipos, funciones y acciones de la API web](#)

 **Nota**

¿Puede indicarnos sus preferencias de idioma de documentación? [Realice una breve encuesta.](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad](#)).

Componer solicitudes HTTP y administrar errores

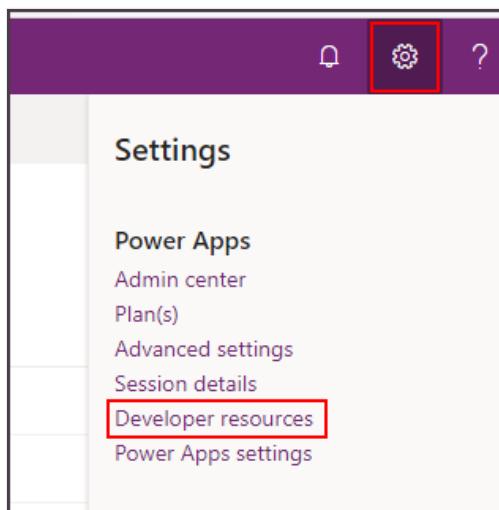
Artículo • 26/09/2023

La interacción con la API web se realiza creando y enviando solicitudes HTTP. Debe saber cómo establecer los encabezados HTTP apropiados y administrar los errores incluidos en la respuesta.

Dirección URL y versiones de API web

Para buscar el URL de la API Web para su entorno:

1. Inicie sesión en [Power Apps](#) y luego seleccione su entorno en la esquina superior derecha.
2. Seleccione el botón **Configuración** en la esquina superior derecha y , a continuación, seleccione **Recursos de desarrollador**.



Desde aquí, puede copiar el valor para el **Extremo de la API web**. Más información: [Ver recursos para desarrolladores](#)

En la siguiente tabla se describen las partes del URL:

Parte	Descripción
Protocolo	<code>https://</code>
Nombre del entorno	El nombre único que se aplica al entorno. Si el nombre de su empresa es <i>Contoso</i> , entonces puede ser <code>contoso</code> .
Región	El entorno normalmente está en un centro de datos que tenga cercano geográficamente. Para Norteamérica es <code>crm</code> . Para América del Sur <code>crm2</code> , para Japón <code>crm7</code> . Para obtener la lista completa, consulte Regiones de centros de datos
URL base	Suele ser <code>dynamics.com</code> , pero algunos centros de datos usan valores diferentes. Consulte Regiones de centros de datos .
Ruta de la API web	La ruta a la API web es <code>/api/data/</code> .
Versión	La versión se expresa de esta manera: <code>v[Major_version].[Minor_version][PatchVersion]/</code> . La versión actual es <code>v9.2</code> .
Recurso	El <code>EntitySetName</code> de la tabla o el nombre de función o la acción que desea usar.

La URL que utiliza se compone de estas partes:

Longitud máxima de la dirección URL

La longitud máxima de la URL aceptada es de 32 KB (32768 caracteres). Esto debería ser adecuado para la mayoría de los tipos de solicitudes, excepto ciertas `GET` operaciones que requieren parámetros de consulta de cadenas muy largas, como consultas que utilizan FetchXml. Si envía solicitudes dentro del cuerpo de una solicitud `$batch`, puede enviar solicitudes con URL de hasta 64 KB (65 536 caracteres). [Obtenga más información sobre cómo enviar FetchXml dentro de una solicitud \\$batch](#).

Compatibilidad de versiones

Esta versión incorpora funciones que no están disponibles en las versiones anteriores. Las versiones secundarias posteriores pueden proporcionar más capacidades que no se agregarán retroactivamente a las versiones secundarias anteriores. El código redactado para v9.0 seguirá funcionando en futuras versiones cuando haga referencia a v9.0 en su dirección URL.

A medida que se lancen nuevas versiones, es posible que entren en conflicto con las versiones anteriores. Estos cambios importantes son necesarios para ir mejorando el servicio. La mayoría de las veces, las funciones permanecen igual entre las versiones, pero no debería suponer que lo harán.

Nota

A diferencia de las versiones v8.x de menor importancia, las nuevas funciones u otros cambios que se agreguen a las versiones futuras no se aplicarán a las versiones anteriores. Preste atención a la versión del servicio que usa y probar el código si cambia la versión utilizada.

Métodos HTTP

La siguiente tabla enumera los métodos HTTP que puede usar con la API web de Dataverse.

método	Uso
<code>GET</code>	Use para recuperar datos, incluida la llamada a funciones. El código de estado esperado para una recuperación correcta es <code>200 OK</code> .
<code>POST</code>	Use para crear entidades o llamar a acciones.
<code>PATCH</code>	Use para actualizar entidades o realizar operaciones de upsert.
<code>DELETE</code>	Use para eliminar entidades o propiedades individuales de entidades.
<code>PUT</code>	Use en situaciones limitadas para actualizar propiedades individuales de entidades. Este método no se recomienda para actualizar la mayoría de las entidades. Use <code>PUT</code> para actualizar definiciones de tabla. Más información: Usar la API web con definiciones de tabla

Encabezados HTTP

Todas las solicitudes HTTP deben incluir al menos los siguientes encabezados.

HTTP
<code>Accept: application/json</code>
<code>OData-MaxVersion: 4.0</code>

`OData-Version: 4.0`
`If-None-Match: null`

Aunque el protocolo OData permite los formatos JSON y ATOM, la API web solo admite JSON. Cada solicitud debe incluir el valor del encabezado `Accept` de `application/json`, aunque no se espere ningún cuerpo de la respuesta. Los errores devueltos en la respuesta se devolverán como JSON. Pese a que su código debería funcionar aunque este encabezado no esté incluido, se recomienda incluirlo como procedimiento recomendado.

La versión de OData actual es la 4.0, pero las versiones futuras pueden permitir nuevas funciones. Para asegurarse de que no haya ambigüedades sobre la versión de OData que será aplicada al código del futuro, debe incluir siempre una instrucción explícito de la versión actual de OData y la versión máxima para aplicar en el código. Use los dos encabezados `OData-Version` y `OData-MaxVersion` establecidos en un valor de `4.0`.

Las consultas que expanden propiedades de navegación valorada como colección pueden devolver datos en caché para las propiedades que no reflejan cambios recientes. Incluya el encabezado `If-None-Match: null` en el cuerpo de la solicitud para reemplazar la memoria caché del explorador de la solicitud de la API web. Más información: [Protocolo de transferencia de hipertexto \(HTTP/1.1\): Solicitud condicional 3.2: If-None-Match](#).

Cada solicitud que incluya datos de JSON en el cuerpo de la solicitud debe incluir un encabezado `Content-Type` con un valor de `application/json`.

HTTP

`Content-Type: application/json`

Puede usar otros encabezados para habilitar funcionalidades específicas.

Preferir encabezados

Puede usar el encabezado [Preferir](#) con los valores a continuación para especificar preferencias.

Valor preferido	Descripción
<code>return=representation</code>	Utilice esta preferencia para devolver datos al crear operaciones (<code>POST</code>) o actualizar operaciones (<code>PATCH</code>) para entidades. Cuando esta preferencia se aplica a una solicitud de <code>POST</code> , una respuesta correcta tendrá el estado <code>201 Created</code> . Para una solicitud <code>PATCH</code> , una respuesta correcta tendrá un estado <code>200 OK</code> . Sin esta preferencia aplicada, ambas operaciones devolverán el estado <code>204 No Content</code> para reflejar que no se devuelve ningún dato en el cuerpo de la respuesta de forma predeterminada. Más información: Crear con datos devueltos & Actualizar con datos devueltos
<code>odata.include-annotations</code>	Ver Solicitar anotaciones
<code>odata.maxpagesize</code>	Utilice esta preferencia para especificar cuántas páginas desea devolver en una consulta. Más información: Página de resultados
<code>odata.track-changes</code>	La característica de seguimiento de cambios permite mantener los datos sincronizados de forma eficiente detectando qué datos se han modificado desde que los datos se extrajeron inicialmente o se sincronizaron por última vez. Más información: Uso del seguimiento de cambios para sincronizar los datos con sistemas externos
<code>respond-async</code>	Especifica que la solicitud debe procesarse de forma asíncrona. Más información: Operaciones en segundo plano (versión preliminar)

! Nota

Se pueden especificar varios encabezados Preferir utilizando valores separados por comas; por ejemplo:

```
Prefer: respond-async,odata.include-annotations="*"
```

Solicitar anotaciones

Puede solicitar que se devuelvan diferentes datos de anotación de OData con los resultados mediante el encabezado de solicitud `Prefer: odata.include-annotations`. Puede optar por devolver todas las anotaciones o especificar anotaciones específicas. La siguiente tabla describe las anotaciones que admite la web API de Dataverse:

Annotation	Descripción
<code>OData.Community.Display.V1.FormattedValue</code>	Devuelve valores de cadena formateados que puede usar en su aplicación. Más información: Valores con formato
<code>Microsoft.Dynamics.CRM.associatednavigationproperty</code> <code>Microsoft.Dynamics.CRM.lookuplogicalname</code>	Devuelve información sobre columnas de búsqueda relacionadas. Más información: Datos de propiedades de búsqueda
<code>Microsoft.Dynamics.CRM.totalrecordcount</code> <code>Microsoft.Dynamics.CRM.totalrecordcountlimitexceeded</code>	Cuando utiliza la opción de consulta <code>\$count</code> , la anotación <code>@odata.count</code> indica la cantidad de registros, pero solo se pueden devolver 5000 registros a la vez. Solicite <code>Microsoft.Dynamics.CRM.totalrecordcountlimitexceeded</code> para obtener un valor booleano que le indicará si el número total de registros que coinciden con la consulta supera los 5000. Más información: Recuento del número de filas
<code>Microsoft.Dynamics.CRM.globalmetadataversion</code>	Esta anotación se devuelve en la solicitud y puede almacenarla en caché en su aplicación. El valor cambia cuando se produce un cambio de esquema, lo que indica que es posible que deba actualizar cualquier dato de esquema que su aplicación haya almacenado en caché. Más información: Almacenar en caché datos de esquema
<code>Microsoft.PowerApps.CDS.ErrorDetails.OperationStatus</code> <code>Microsoft.PowerApps.CDS.ErrorDetails.SubErrorCode</code> <code>Microsoft.PowerApps.CDS.HelpLink</code> <code>Microsoft.PowerApps.CDS.TraceText</code> <code>Microsoft.PowerApps.CDS.InnerException.Message</code>	Estas anotaciones proporcionan más detalles cuando se devuelven errores. Más información: Incluir más detalles con errores

Si solo desea anotaciones específicas, puede solicitarlas como valores separados por comas. También puede utilizar el carácter '*' como comodín. Por ejemplo, el siguiente encabezado de solicitud `Prefer` solo incluye los valores formateados y las anotaciones de detalles de errores adicionales:

```
Prefer: odata.include-annotations="OData.Community.Display.V1.FormattedValue,Microsoft.PowerApps.CDS.ErrorDetails*"
```

💡 Sugerencia

Es común usar simplemente el encabezado de solicitud `Prefer: odata.include-annotations="*"` para devolver todas las anotaciones.

Otros encabezados

Encabezado	valor	Descripción
<code>CallerObjectId</code>	Identificador de objeto de usuario de Azure Active Directory	Utilice este encabezado para suplantar a otro usuario cuando la persona que llama tenga los privilegios para hacerlo. Establezca el valor con el identificador del objeto de Azure Active Directory del usuario al que desea suplantar. Estos datos están en el atributo (columna) AzureActiveDirectoryObjectId de la tabla/entity de usuario (SystemUser) . Más información: Suplantar a otro usuario utilizando la API web
<code>If-Match</code>	Valor de <code>Etag</code> o <code>*</code>	Utilice este encabezado para aplicar simultaneidad optimista para asegurarse de no sobrescribir los cambios que otra persona aplicó en el servidor desde que recuperó un registro. Más información: Aplicar la simultaneidad optimista e If-Match También puede utilizar este encabezado con <code>*</code> para prevenir que una operación <code>PATCH</code> cree un registro. Más información: Evitar realizar operaciones de creación en upsert
<code>If-None-Match</code>	<code>null</code> o <code>*</code>	Este encabezado debe usarse en todas las solicitudes con un valor de <code>null</code> como se describe en Encabezados HTTP , pero también se puede utilizar para prevenir que una operación <code>POST</code> realice una actualización. Más información: Evitar actualización en upsert e If-None-Match
<code>MSCRM.SolutionUniqueName</code>	nombre único de la solución	Utilice este encabezado cuando desee crear un componente de solución y asociarlo con una solución no administrada. Más información: Crear y actualizar definiciones de tabla mediante la API web
<code>MSCRM.SuppressDuplicateDetection</code>	<code>false</code>	Use este encabezado con el valor <code>false</code> para habilitar detección de duplicados al crear o actualizar un registro. Más información: Búsqueda de registros de duplicados
<code>MSCRM.BypassCustomPluginExecution</code>	<code>true</code>	Use este encabezado cuando desee omitir el código de complemento personalizado y la persona que llama tiene el privilegio <code>prvBypassCustomPlugins</code> . Más información: Omitir la lógica empresarial personalizada
<code>Consistency</code>	<code>strong</code>	Utilice este encabezado cuando deba tener la versión más reciente de un elemento almacenado en caché. Los elementos almacenados en caché incluyen: definiciones de metadatos, etiquetas, permisos de usuario y permisos de equipo. Por ejemplo, si aplica un cambio a alguna definición, etiqueta o permiso de metadatos y tiene un código que debe usar la última definición dentro de los 30 segundos posteriores al cambio, puede usar este encabezado para asegurarse de obtener la última versión. Usar este encabezado incurre en una pequeña penalización de rendimiento, por lo que no debe usarse todo el tiempo.

Al ejecutar operaciones por lotes, deberá aplicar muchos encabezados en la solicitud y con cada parte enviada en el cuerpo. Más información: [Ejecute las operaciones por lotes mediante API web](#)

Identificar códigos de estado

Tanto si la solicitud HTTP es correcta como si no, la respuesta incluye un código de estado. La siguiente tabla describe los códigos de estado devueltos por la API web de Microsoft Dataverse.

Código	Descripción	Type
<code>200 OK</code>	Espere este código de estado cuando la operación devuelva datos en el cuerpo de la respuesta.	Correcto
<code>201 Created</code>	Cuento con este código de estado cuando la operación POST de la entidad se realice correctamente y haya especificado la preferencia <code>return=representation</code> en su solicitud.	Correcto

Código	Descripción	Type
204 No Content	Espere este código de estado cuando la operación sea correcta, pero no devuelva datos en el cuerpo de la respuesta.	Correcto
304 Not Modified	Espere este código de estado al comprobar si se ha modificado una entidad desde la última vez que se recuperó. Más información: Recuperaciones condicionales .	Redirección
403 Forbidden	Espere este código de estado para los siguientes tipos de errores: - AccessDenied - AttributePermissionReadIsMissing - AttributePermissionUpdateIsMissingDuringUpdate - AttributePrivilegeCreateIsMissing - CannotActOnBehalfOfAnotherUser - CannotAddOrActOnBehalfAnotherUserPrivilege - CrmSecurityError - InvalidAccessRights - PrincipalPrivilegeDenied - PrivilegeCreateIsDisabledForOrganization - PrivilegeDenied - unManagedinvalidprincipal - unManagedinvalidprivilegedepth	Error de cliente
401 Unauthorized	Espere este código de estado para los siguientes tipos de errores: - BadAuthTicket - ExpiredAuthTicket - InsufficientAuthTicket - InvalidAuthTicket - InvalidUserAuth - MissingCrmAuthenticationToken - MissingCrmAuthenticationTokenOrganizationName - RequestIsNotAuthenticated - TamperedAuthTicket - UnauthorizedAccess - UnManagedInvalidSecurityPrincipal	Error de cliente
413 Payload Too Large	Espere este código de estado cuando la longitud de la solicitud sea demasiado grande.	Error de cliente
400 BadRequest	Espere este código de estado cuando un argumento no sea válido.	Error de cliente
404 Not Found	Espere este código de estado cuando no exista el recurso.	Error de cliente
405 Method Not Allowed	Este error se produce porque el método y las combinaciones de recursos no son correctas. Por ejemplo, no puede utilizar DELETE o PATCH en una colección de entidades. Esto se produce para los siguientes tipos de errores: - CannotDeleteDueToAssociation - InvalidOperation - NotSupported	Error de cliente
412 Precondition Failed	Espere este código de estado para los siguientes tipos de errores: - ConcurrencyVersionMismatch - DuplicateRecord	Error de cliente
429 Too Many Requests	Espere este código de estado se superan los límites de la API. Más información: Límites de API de protección de servicios	Error de cliente

Código	Descripción	Type
501 Not Implemented	Espere este código de estado cuando alguna operación solicitada no se implementa.	Error de servidor
503 Service Unavailable	Espere este código de estado cuando el servicio de la API web no está disponible.	Error de servidor

Errores de análisis de la respuesta

Los detalles sobre los errores se incluyen como JSON en la respuesta en el siguiente formato.

JSON

```
{
  "error": {
    "code": "<This code is not related to the http status code and is frequently empty>",
    "message": "<A message describing the error>"
  }
}
```

Incluir más detalles con los errores

Algunos errores pueden incluir más detalles mediante *anotaciones*. Cuando una solicitud incluye el encabezado `Prefer: odata.include-annotations="*"`, la respuesta incluye todas las anotaciones que contienen más detalles sobre los errores y una URL que puede dirigirle a una guía específica para solucionar el error.

Algunos de estos detalles los pueden configurar los desarrolladores que escriben complementos. Por ejemplo, supongamos que tiene un complemento que arroja un error al usar el constructor `InvalidPluginExecutionException(OperationStatus, Int32, String)`. Este constructor le permite pasar un valor `OperationStatus`, un código de error de entero personalizado y un mensaje de error.

Un complemento sencillo podría tener este aspecto:

C#

```
namespace MyNamespace
{
    public class MyClass : IPlugin
    {
        public void Execute(IServiceProvider serviceProvider)
        {

            // Obtain the tracing service
            ITracingService tracingService =
                (ITracingService)serviceProvider.GetService(typeof(ITracingService));

            tracingService.Trace("Entering MyClass plug-in.");

            try
            {
                throw new InvalidPluginExecutionException(OperationStatus.Canceled, 12345, "Example
Error Message.");
            }
            catch (InvalidPluginExecutionException ex)
            {
                tracingService.Trace("StackTrace:");
                tracingService.Trace(ex.StackTrace);
                throw ex;
            }
        }
    }
}
```

```
}
```

Cuando este complemento se registra en la creación de una entidad account, y la solicitud para crear una cuenta incluye la preferencia `odata.include-annotations="*"`, la solicitud y la respuesta tienen el siguiente aspecto:

Solicitud:

HTTP

```
POST https://yourorg.api.crm.dynamics.com/api/data/v9.1/accounts HTTP/1.1
Content-Type: application/json;
Prefer: odata.include-annotations="*"
{
    "name": "Example Account"
}
```

Respuesta:

HTTP

```
HTTP/1.1 400 Bad Request
Content-Type: application/json; odata.metadata=minimal
{
    "error": {
        "code": "0x80040265",
        "message": "Example Error Message.",
        "@Microsoft.PowerApps.CDS.ErrorDetails.OperationStatus": "1",
        "@Microsoft.PowerApps.CDS.ErrorDetails.SubErrorCode": "12345",
        "@Microsoft.PowerApps.CDS.HelpLink": "http://go.microsoft.com/fwlink/?LinkId=398563&error=Microsoft.Crm.CrmException%3a80040265&client=platform",
        "@Microsoft.PowerApps.CDS.TraceText": "\r\n[MyNamespace: MyNamespace.MyClass ]\r\n[52e2dbb9-85d3-ea11-a812-000d3a122b89: MyNamespace.MyClass : Create of account] \r\n\r\n Entering MyClass plug-in.\r\nStackTrace:\r\n    at MyNamespace.MyClass.Execute(IServiceProvider serviceProvider)\r\n\r\n"
        "@Microsoft.PowerApps.CDS.InnerError.Message": "Example Error Message."
    }
}
```

En la tabla siguiente se describen la anotación en la respuesta.

Anotación y descripción	valor
<code>@Microsoft.PowerApps.CDS.ErrorDetails.OperationStatus</code>	1 El valor de <code>OperationStatus</code> establecido por el constructor <code>InvalidPluginExecutionException(OperationStatus, Int32, String)</code> .
<code>@Microsoft.PowerApps.CDS.ErrorDetails.SubErrorCode</code>	12345 El valor de <code>SubErrorCode</code> establecido por el constructor <code>InvalidPluginExecutionException(OperationStatus, Int32, String)</code> .
<code>@Microsoft.PowerApps.CDS.HelpLink</code>	http://go.microsoft.com/fwlink/?LinkId=398563&error=Microsoft.Crm.CrmException%3a80040265&client=platform Una URL que contiene información sobre el error que <i>puede</i> redirigirlo a una guía sobre cómo abordar el error.
<code>@Microsoft.PowerApps.CDS.TraceText</code>	[MyNamespace: MyNamespace.MyClass] [52e2dbb9-85d3-ea11-a812-000d3a122b89: MyNamespace.MyClass :Create of account] Entering MyClass plug-in. ITracingService.Trace(String, Object[]). Esta anotación

Anotación y descripción	valor
incluye el seguimiento de pila del complemento porque el autor del complemento lo registró.	StackTrace: at MyNamespace.MyClass.Execute(IServiceProvider serviceProvider)
@Microsoft.PowerApps.CDS.InnerException.Message El mensaje de error encontrado en InnerError para la excepción. Este mensaje debe ser el mismo que el mensaje de error, excepto en ciertos casos especiales que son solo para uso interno.	Example Error Message.

ⓘ Nota

No se garantiza que el `@Microsoft.PowerApps.CDS.HelpLink` proporcione una guía para cada error. La guía se *puede* proporcionar de forma proactiva, pero lo más común es que se proporcione de forma reactiva según la frecuencia con la que se utilice el vínculo. Utilice el vínculo. Si no proporciona orientación, su uso del vínculo nos ayuda a detectar que las personas necesitan más orientación sobre el error. Luego, podemos priorizar la inclusión de orientación sobre errores que más necesitan las personas. Los recursos a los que el vínculo puede dirigirlo pueden ser documentación, vínculos a recursos comunitarios o sitios externos.

Si no desea recibir todas las anotaciones en la respuesta, puede especificar qué anotaciones desea que se devuelvan. En lugar de usar `Prefer: odata.include-annotations="*"`, puede usar el siguiente valor para recibir solo valores formateados para operaciones que recuperan datos y el vínculo de ayuda si ocurre un error: `Prefer: odata.include-annotations="OData.Community.Display.V1.FormattedValue,Microsoft.PowerApps.CDS.HelpLink"`.

Más información: [Anotaciones de la solicitud](#)

Agregar una variable compartida desde la API web

Puede establecer un valor de cadena que está disponible para los complementos dentro de ExecutionContext en la colección `SharedVariables`. Más información:

- [Agregar una variable compartida al contexto de ejecución del complemento](#)
- [Variables compartidas](#)

Consulte también

[Realizar operaciones mediante la API web](#)

[Consultar datos utilizando la API web](#)

[Crear una fila de tabla usando la API web](#)

[Recuperar una fila de tabla usando la API web](#)

[Actualizar y eliminar filas de tablas usando la API web](#)

[Asociar y anular la asociación de filas de tabla mediante la API web](#)

[Usar funciones de la API web](#)

[Usar acciones de la API web](#)

[Ejecutar las operaciones por lotes mediante API web](#)

[Suplantar a otro usuario utilizando la API web](#)

[Realizar operaciones condicionales mediante la API web](#)

ⓘ Nota

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta.↗ (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales (declaración de privacidad [↗](#)).

Consultar datos utilizando la API web

Artículo • 12/09/2023

Cuando utiliza la API web para crear una consulta en una tabla de Dataverse, debe tomar las siguientes decisiones:

Decisión	Descripción
Seleccionar columnas	Qué columnas de datos devolver
Unir tablas	Qué tablas relacionadas incluir en los resultados
Ordenar filas	En qué orden devolver los resultados
Filtrar filas	Qué filas de datos devolver
Resultados de página	Cuántas filas de datos se devuelven
Agregar datos	Cómo agrupar y agregar los datos devueltos
Recuento del número de filas	Cómo contar el número de filas

Este artículo trata sobre la consulta de datos encontrados en tablas. También puede utilizar la API web para consultar datos sobre *definiciones de tablas* o *metadatos de la entidad*. La estructura de los datos es diferente, por lo que muchas de las capacidades descritas aquí no se aplican. Más información: [Consultar definiciones de tabla mediante la API web](#) y [Consultar definiciones de esquemas](#)

Colecciones de entidades

Cada consulta comienza con una colección de entidades. Las colecciones de entidades pueden ser:

- [Recursos de EntitySet](#): una de las colecciones de EntitySet de API web.
- [Colecciones filtradas](#): un conjunto de entidades devueltas por una [propiedad de navegación con valor de colección](#) para un registro específico.
- Una propiedad de navegación valorada como colección expandida. Más información: [Expandir propiedades de navegación con valor de colección](#)

Recursos de EntitySet

Para encontrar todos los recursos de EntitySet disponibles en su entorno, envíe una solicitud `GET` al [documento de servicio](#) de la API web:

Solicitud:

HTTP

```
GET [Organization URI]/api/data/v9.2/
Accept: application/json
OData-MaxVersion: 4.0
OData-Version: 4.0
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0

{
    "@odata.context": "[Organization URI]/api/data/v9.2/$metadata",
    "value": [
        {
            "name": "aadusers",
            "kind": "EntitySet",
            "url": "aadusers"
        },
        {
            "name": "accountleadscollection",
            "kind": "EntitySet",
            "url": "accountleadscollection"
        },
        {
            "name": "accounts",
            "kind": "EntitySet",
            "url": "accounts"
        },
        ...
        ... <Truncated for brevity>
    ]
}
```

💡 Sugerencia

Estos valores suelen ser el nombre plural de la tabla, pero pueden ser diferentes. Use esta consulta para confirmar que está usando el nombre de recurso EntitySet correcto.

Para recuperar datos del [tipo de entidad de cuenta](#), empieza con el recurso EntitySet de [accounts](#).

HTTP

```
GET [Organization URI]/api/data/v9.2/accounts?$select=name
```

Colecciones filtradas

Puede consultar cualquier colección de entidades representadas por una propiedad de navegación con valor de colección de un registro específico.

Si desea recuperar datos del [tipo de entidad de cuenta](#), donde un usuario específico es el [OwningUser](#), puede usar la propiedad de navegación `user_accounts` con valor de colección del registro especificado de [systemuser](#).

HTTP

```
GET [Organization URI]/api/data/v9.2/systemusers(<systemuserid value>)/user_accounts?$select=name
```

Para localizar el nombre de la colección de navegación de un solo valor:

- Para cualquier tabla y relación de Dataverse, puede consultar la [Web API Entity Type Reference](#)
- Para cualquier tabla personalizada o relación, busque las [propiedades de navegación con valor de colección](#) dentro del [documento de servicios de \\$metadatos](#)

Opciones de consulta de OData

La siguiente tabla describe las opciones de consulta de OData que admite la API web de Dataverse.

Opción	Usar para	Más información
<code>\$select</code>	Solicite un conjunto específico de propiedades para cada entidad o tipo de complejo.	Seleccionar columnas
<code>\$expand</code>	Especifique los recursos relacionados que se incluirán en línea con los recursos recuperados.	Unir tablas
<code>\$filter</code>	Filtre una colección de recursos.	Filtrar filas
<code>\$orderby</code>	Solicite recursos en un orden particular.	Ordenar filas
<code>\$apply</code>	Agregue y agrupe sus datos.	Aregar datos

Opción	Usar para	Más información
\$top	Especifique el número de elementos de la colección consultada que se incluirán en el resultado. No use \$top cuando recupera páginas de datos.	Usar la opción de consulta \$top
\$count	Solicite un recuento de los recursos coincidentes incluidos con los recursos en la respuesta.	Recuento del número de filas

Puede aplicar varias opciones a una consulta. Separe las opciones de consulta de la ruta del recurso con un signo de interrogación (?). Separe cada opción después de la primera con un ampersand (&). Los nombres de opción distinguen entre mayúsculas y minúsculas.

La API web de Dataverse no es compatible con estas [opciones de consulta de OData](#):
\$skip, \$search, \$format.

Uso de alias de parámetro con opciones de consulta

Puede utilizar alias de parámetros para las opciones de consulta \$filter y \$orderby, pero actualmente no dentro de la opción \$expand. Los alias de parámetro permiten usar el mismo valor varias veces en una solicitud. Si al alias no se asigna un valor se da por hecho que es nulo.

Sin alias de parámetros:

HTTP

```
GET [Organization URI]/api/data/v9.2/accounts?$select=name,revenue
&$orderby=revenue asc,name desc
&$filter=revenue ne null
```

Con alias de parámetros:

HTTP

```
GET [Organization URI]/api/data/v9.2/accounts?$select=name,revenue
&$orderby=@p1 asc,@p2 desc
&$filter=@p1 ne @p3&@p1=revenue&@p2=name
```

También puede usar alias de parámetro al usar funciones. Más información: [Usar funciones web API](#)

Seleccionar columnas

ⓘ Importante

Cuando consulta datos, es importante limitar la cantidad de datos devueltos para optimizar el rendimiento. Solo seleccione las columnas con datos que necesita.

Utilice la [opción de consulta \\$select](#) para elegir qué columnas devolver con su consulta. En OData, cada columna se representa como una [propiedad](#). Si no incluye una opción de consulta `$select`, se devuelven todas las propiedades.

El siguiente ejemplo solicita las propiedades `name` y `revenue` de una fila del recurso EntitySet de `accounts`:

Solicitud:

HTTP

```
GET [Organization URI]/api/data/v9.2/accounts?$select=name,revenue&$top=1
Accept: application/json
OData-MaxVersion: 4.0
OData-Version: 4.0
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0

{
    "@odata.context": "[Organization
URI]/api/data/v9.2/$metadata#accounts(name,revenue)",
    "value": [
        {
            "@odata.etag": "W/\\"81052965\\\"",
            "name": "Litware, Inc. (sample)",
            "revenue": 20000.0000,
            "_transactioncurrencyid_value": "228f42f8-e646-e111-8eb7-
78e7d162ced1",
            "accountid": "4624eff7-53d3-ed11-a7c7-000d3a993550"
        }
    ]
}
```

La propiedad de la clave principal siempre se devuelve, por lo que no necesita incluirla en su `$select`. En este ejemplo, `accountid` es la clave principal.

También se pueden incluir otros valores de propiedad. En este caso, la [propiedad de búsqueda](#) `_transactioncurrencyid_value` para la [referencia de entidad/tabla de divisas \(TransactionCurrency\)](#) relacionada se incluye porque `revenue` es una propiedad de moneda.

¿Qué propiedades están disponibles?

Todas las propiedades disponibles para una entidad se encuentran en el [documento de servicio de \\$metadatos](#). Más información: [Propiedades de API web](#)

Los tipos de entidad incluidos con Dataverse se describen en [Web API Entity Type Reference](#).

💡 Sugerencia

La forma más fácil de descubrir rápidamente qué propiedades están disponibles es enviar una solicitud mediante la opción de consulta `$top` con un valor de `1` sin utilizar `$select`.

Valores con formato

Los valores con formato son valores de cadena generados en el servidor que puede usar en su aplicación. Los valores con formato incluyen:

- Las etiquetas localizadas para las columnas elección, opciones, sí/no, estado y razón para el estado
- El valor de nombre principal para las propiedades de búsqueda y propietario
- Valores de moneda con símbolos de moneda
- Valores de fecha con formato en la zona horaria del usuario

Para incluir los valores con formato en sus resultados, use este encabezado de solicitud:

```
Prefer: odata.include-  
annotations="OData.Community.Display.V1.FormattedValue"
```

Los valores con formato son una de varias anotaciones que puede solicitar. Utilice `Prefer: odata.include-annotations="*"` para incluir todas las anotaciones. Más información: [Anotaciones de la solicitud](#)

El valor con formato se devuelve con el registro con una anotación que sigue esta convención:

```
<property name>@odata.Community.Display.V1.FormattedValue
```

Por ejemplo:

Solicitud:

HTTP

```
GET [Organization URI]/api/data/v9.2/accounts?  
$select=name,revenue,_primarycontactid_value,customertypecode,modifiedon  
&$top=1  
Accept: application/json  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
Prefer: odata.include-  
annotations="OData.Community.Display.V1.FormattedValue"
```

La siguiente tabla describe los valores y los valores con formato que se devuelven para las propiedades solicitadas.

Property	valor	Valor con formato
name	Litware, Inc. (sample)	None
revenue	20000.0000	\$20,000.00
_primarycontactid_value	70bf4d48-34cb-ed11-b596- 0022481d68cd	Susanna Stubberod (sample)
customertypecode	1	Competitor
modifiedon	2023-04-07T21:59:01Z	4/7/2023 2:59 PM
_transactioncurrencyid_value	228f42f8-e646-e111-8eb7- 78e7d162ced1	US Dollar
accountid	78914942-34cb-ed11-b596- 0022481d68cd	None

Respuesta:

HTTP

```
HTTP/1.1 200 OK
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0
Preference-Applied: odata.include-
annotations="OData.Community.Display.V1.FormattedValue"

{
    "@odata.context": "[Organization
URI]/api/data/v9.2/$metadata#accounts(name,revenue)",
    "value": [
{
        "@odata.etag": "W/\\"81359849\\",
        "name": "Litware, Inc. (sample)",
        "revenue@OData.Community.Display.V1.FormattedValue":
"$20,000.00",
        "revenue": 20000.0000,
        "_primarycontactid_value@OData.Community.Display.V1.FormattedValue":
"Susanna Stubberod (sample)",
        "_primarycontactid_value": "70bf4d48-34cb-ed11-b596-
0022481d68cd",
        "customertypecode@OData.Community.Display.V1.FormattedValue":
"Competitor",
        "customertypecode": 1,
        "modifiedon@OData.Community.Display.V1.FormattedValue":
"4/7/2023 2:59 PM",
        "modifiedon": "2023-04-07T21:59:01Z",
        "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
"US Dollar",
        "_transactioncurrencyid_value": "228f42f8-e646-e111-8eb7-
78e7d162ced1",
        "accountid": "78914942-34cb-ed11-b596-0022481d68cd"
    }
]
}
```

Datos de propiedades de búsqueda

Cuando una [propiedad de búsqueda](#) representa una relación de varias tablas o polimórfica, debe solicitar anotaciones específicas para determinar qué tabla contiene los datos relacionados.

Por ejemplo, muchas tablas tienen registros que pueden ser propiedad de usuarios o equipos. Los datos de propiedad se almacenan en una columna de búsqueda denominada `ownerid`. Esta columna es una propiedad de navegación de un solo valor

en OData. Podría usar `$expand` para crear una combinación para obtener este valor, pero no puede usar `$select`. Sin embargo, puede usar la propiedad de búsqueda `_ownerid_value` correspondiente con `$select`.

Cuando incluyes la propiedad de búsqueda `_ownerid_value` con su `$select`, devuelve un valor GUID. Este valor no indica si el propietario del registro es un usuario o un equipo. Debe solicitar anotaciones para obtener estos datos.

Para incluir estas anotaciones en sus resultados, use este encabezado de solicitud:

```
Prefer: odata.include-  
annotations="Microsoft.Dynamics.CRM.associatednavigationproperty,Microsoft.D  
ynamics.CRM.lookuplogicalname"
```

💡 Sugerencia

O puede usar `Prefer: odata.include-annotations="*"` para incluir todas las anotaciones. Más información: [Anotaciones de la solicitud](#)

Solicitud:

HTTP

```
GET [Organization URI]/api/data/v9.2/accounts?  
$select=name,_ownerid_value&$top=2  
Accept: application/json  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
Prefer: odata.include-  
annotations="Microsoft.Dynamics.CRM.associatednavigationproperty,Microsoft.D  
ynamics.CRM.lookuplogicalname"
```

La siguiente respuesta devuelve dos registros de cuenta diferentes. `team` posee el primero, y `systemuser` posee el segundo. La anotación `_ownerid_value@Microsoft.Dynamics.CRM.lookuplogicalname` proporciona esta información.

Respuesta:

HTTP

```

HTTP/1.1 200 OK
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0
Preference-Applied: odata.include-
annotations="Microsoft.Dynamics.CRM.associatednavigationproperty,Microsoft.D
ynamics.CRM.lookuplogicalname"

{
    "@odata.context": "[Organization
URI]/api/data/v9.2/$metadata#accounts(name,_ownerid_value)",
    "value": [
        {
            "@odata.etag": "W/\\"81550512\\",
            "name": "Adventure Works (sample)",

            "_ownerid_value@Microsoft.Dynamics.CRM.associatednavigationproperty":
            "ownerid",
                "_ownerid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
            "team",
                "_ownerid_value": "39e0dbe4-131b-e111-ba7e-78e7d1620f5e",
                "accountid": "1adef0b8-54d3-ed11-a7c7-000d3a993550"
            },
            {
                "@odata.etag": "W/\\"81359849\\",
                "name": "Litware, Inc. (sample)",

                "_ownerid_value@Microsoft.Dynamics.CRM.associatednavigationproperty":
                "ownerid",
                    "_ownerid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
                "systemuser",
                    "_ownerid_value": "4026be43-6b69-e111-8f65-78e7d1620f5e",
                    "accountid": "78914942-34cb-ed11-b596-0022481d68cd"
                }
            ]
    }
}

```

- <lookup property name>@Microsoft.Dynamics.CRM.lookuplogicalname es el nombre lógico de la tabla relacionada.
- <lookup property name>@Microsoft.Dynamics.CRM.associatednavigationproperty es el nombre de la propiedad de navegación de un solo valor correspondiente. Puede utilizar \$expand utilizando este valor en otra solicitud para obtener más datos del registro relacionado.

Unir tablas

Para controlar qué datos se devuelven se los registros de tabla relacionados, use la opción de consulta \$expand con propiedades de navegación.

- Puedes incluir hasta 15 opciones `$expand` en una consulta. Cada una de las opciones `$expand` crea una unión que puede afectar al rendimiento.
- Las consultas que expanden propiedades de navegación valorada como colección pueden devolver datos en caché para las propiedades que no reflejan cambios recientes. Se recomienda usar el encabezado `If-None-Match` con valor `null` para reemplazar el almacenamiento en la memoria caché del explorador. Más información: [Encabezados HTTP](#) para obtener más detalles.

La siguiente tabla describe las [opciones de consulta](#) que puedes aplicar en ciertas opciones `$expand`:

Opción	Descripción
<code>\$select</code>	Seleccione qué propiedades se devuelven. Más información: Seleccionar columnas
<code>\$filter</code>	Para las propiedades de navegación con valor de colección, limite los registros devueltos. Más información: Filtrar filas
<code>\$orderby</code>	Para las propiedades de navegación con valor de colección, controle el orden de los registros devueltos. No es compatible con <code>\$expand</code> anidado. Más información: \$expand anidado en propiedades de navegación valorada en una colección
<code>\$top</code>	Para las propiedades de navegación con valor de colección, limite el número de registros devueltos. No es compatible con <code>\$expand</code> anidado. Más información: \$expand anidado en propiedades de navegación valorada en una colección
<code>\$expand</code>	Expanda las propiedades de navegación en el conjunto de entidades relacionadas. El uso de <code>\$expand</code> en <code>\$expand</code> se llama <code>\$expand anidado</code> . Más información: Expansión anidada de propiedades de navegación de un único valor y \$expand anidado en propiedades de navegación con valor de colección

Estas opciones son un subconjunto de las opciones de consulta descritas en la sección [11.2.4.2.1 Opciones de ampliación](#) de [OData versión 4.0 parte 1, Protocol Plus Errata 02](#). Las opciones `$skip`, `$count`, `$search` y `$levels` no se admiten para la API web de Dataverse.

Use estas opciones con `$expand` agregándolas entre paréntesis después del nombre de la propiedad de navegación. Separe cada opción con punto y coma (;).

Por ejemplo, la siguiente consulta:

- Solicita la propiedad `account.name`
- Une la solicitud de propiedad de navegación con valor de colección `AccountTasks`:
 - La propiedad `task.subject`
 - Donde `task.subject` contiene la cadena "Task"

- o Ordenado por la fecha `task.createdon` de forma descendente

HTTP

```
/accounts?
$select=name&$expand=Account_Tasks($select=subject;$filter=contains(subject,
'Task'));$orderby=createdon desc)
```

Limitar columnas con `$select`

Como con cualquier consulta, siempre limite las columnas devueltas usando `$select` cuando use `$expand`. Por ejemplo, la siguiente solicitud devuelve los valores `contact.fullname` y `task.subject` en los resultados expandidos del tipo de entidad `account`:

Solicitud:

HTTP

```
GET [Organization URI]/api/data/v9.2/accounts?$select=name
&$expand=primarycontactid($select=fullname),Account_Tasks($select=subject)
Prefer: odata.maxpagesize=1
If-None-Match: null
Accept: application/json
OData-MaxVersion: 4.0
OData-Version: 4.0
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0
Preference-Applied: odata.maxpagesize=1

{
    "@odata.context": "[Organization
URI]/api/data/v9.2/$metadata#accounts(name,primarycontactid(fullname),Accoun
t_Tasks(subject))",
    "value": [
        {
            "@odata.etag": "W/\\"80649578\\\"",
            "name": "Litware, Inc. (sample)",
            "accountid": "78914942-34cb-ed11-b596-0022481d68cd",
            "primarycontactid": {
                "fullname": "Susanna Stubberod (sample)",
                "contactid": "70bf4d48-34cb-ed11-b596-0022481d68cd"
            },
        }
    ]
}
```

```

"Account_Tasks": [
  {
    "@odata.etag": "W/\\"80649460\\",
    "subject": "Task 1 for Litware",
    "_regardingobjectid_value": "78914942-34cb-ed11-b596-
0022481d68cd",
    "activityid": "f68393c1-34cb-ed11-b597-000d3a993550"
  }
],
"Account_Tasks@odata.nextLink": "[Organization
URI]/api/data/v9.2/accounts(78914942-34cb-ed11-b596-
0022481d68cd)/Account_Tasks?$select=subject"
},
"@odata.nextLink": "[Organization URI]/api/data/v9.2/accounts?
$select=name&$expand=primarycontactid($select=fullname),Account_Tasks($selec
t=subject)&$skiptoken=%3Ccookie%20pagenumber=%222%22%20pagingcookie=%22%253c
cookie%2520page%253d%25221%2522%253e%253caccountid%2520last%253d%2522%257b78
914942-34CB-ED11-B596-
0022481D68CD%257d%2522%2520first%253d%2522%257b78914942-34CB-ED11-B596-
0022481D68CD%257d%2522%2520%252f%253e%253c%252fcookie%253e%22%20istracking=%
22False%22%20/%3E"
}

```

Diferencias de tipo de propiedad de navegación

Es importante recordar que hay dos tipos de propiedades de navegación. Más información: [Propiedades de navegación de la API web](#)

- Las propiedades de navegación *de un solo valor* corresponden a atributos de búsqueda que admiten relaciones de varios a uno y permiten establecer una referencia a otro registro.
- Las propiedades de navegación *valoradas como colección* corresponden a relaciones de uno a varios o de varios a varios.

Expandir una propiedad de navegación con valor de colección puede hacer que el tamaño de la respuesta sea grande en formas difíciles de anticipar. Es importante que incluya límites para controlar la cantidad de datos que se devuelven. Puede limitar el número de registros utilizando la paginación. Más información: [Página de resultados](#)

Hay una diferencia significativa en cómo se aplica la paginación a las opciones \$expand anidadas que se aplican a las propiedades de navegación con valor de colección. Más información: [Expandir propiedades de navegación con valor de colección](#)

Expandir propiedades de navegación de un solo valor

El siguiente ejemplo demuestra cómo recuperar registros de contacto, incluido el contacto principal y el usuario que creó los registros.

Solicitud:

HTTP

```
GET [Organization URI]/api/data/v9.2/accounts?$select=name  
&$expand=primarycontactid($select=contactid,fullname),createdby($select=full  
name)  
Prefer: odata.maxpagesize=2  
If-None-Match: null  
Accept: application/json  
OData-MaxVersion: 4.0  
OData-Version: 4.0
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK  
Content-Type: application/json; odata.metadata=minimal  
Preference-Applied: odata.maxpagesize=2  
OData-Version: 4.0  
  
{  
    "@odata.context": "[Organization  
URI]/api/data/v9.2/$metadata#accounts(name,primarycontactid(contactid,fullna  
me),createdby(fullname))",  
    "value": [  
        {  
            "@odata.etag": "W/\\"80649578\\\"",  
            "name": "Litware, Inc. (sample)",  
            "accountid": "78914942-34cb-ed11-b596-0022481d68cd",  
            "primarycontactid": {  
                "contactid": "70bf4d48-34cb-ed11-b596-0022481d68cd",  
                "fullname": "Susanna Stubberod (sample)"  
            },  
            "createdby": {  
                "fullname": "System Administrator",  
                "systemuserid": "4026be43-6b69-e111-8f65-78e7d1620f5e",  
                "ownerid": "4026be43-6b69-e111-8f65-78e7d1620f5e"  
            }  
        },  
        {  
            "@odata.etag": "W/\\"80649580\\\"",  
            "name": "Adventure Works (sample)",  
            "accountid": "7a914942-34cb-ed11-b596-0022481d68cd",  
            "primarycontactid": {  
                "contactid": "72bf4d48-34cb-ed11-b596-0022481d68cd",  
                "fullname": "Nancy Anderson (sample)"  
            },  
        }  
    ]  
}
```

```

    "createdby": {
        "fullname": "System Administrator",
        "systemuserid": "4026be43-6b69-e111-8f65-78e7d1620f5e",
        "ownerid": "4026be43-6b69-e111-8f65-78e7d1620f5e"
    }
},
"@odata.nextLink": "[Organization URI]/api/data/v9.2/accounts?
$select=name%0A&$expand=primarycontactid($select=contactid,fullname),created
by($select=fullname)&$skiptoken=%3Ccookie%20pagenumber=%222%22%20pagingcooki
e=%22%253ccookie%2520page%253d%25221%2522%253e%253caccountid%25201last%253d%2
522%257b7A914942-34CB-ED11-B596-
0022481D68CD%257d%2522%2520first%253d%2522%257b78914942-34CB-ED11-B596-
0022481D68CD%257d%2522%2520%252f%253e%253c%252fcookie%253e%22%20istracking=%
22False%22%20/%3E"
}

```

La propiedad de navegación de valor único `createdby` devuelve una instancia de [systemuser EntityType](#). Se devuelven las propiedades `systemuserid` y `ownerid`. Esto se debe a que `systemuser` hereda de [EntityType principal](#) y comparte la clave principal `ownerid` con [EntityType de equipo](#) a través de esta herencia.

Sin embargo, la [tabla Usuario \(SystemUser\)](#) tiene la clave principal de [SystemUserId](#). Ambas propiedades `systemuserid` y `ownerid` tienen el mismo valor. Más información: [Herencia EntityType](#)

Devolver referencias

En lugar de devolver datos, también puede devolver referencias o vínculos a los registros relacionados expandiendo la propiedad de navegación de un solo valor con la opción `/$ref`. El siguiente ejemplo devuelve objetos JSON con una propiedad `@odata.id` que tiene una URL para cada contacto principal.

Solicitud:

HTTP

```

GET [Organization URI]/api/data/v9.2/accounts?$select=name
&$expand=primarycontactid/$ref
Accept: application/json
OData-MaxVersion: 4.0
OData-Version: 4.0

```

Respuesta:

HTTP

```

HTTP/1.1 200 OK
Content-Type: application/json; odata.metadata=minimal
Preference-Applied: odata.maxpagesize=2
OData-Version: 4.0

{
    "@odata.context": "[Organization
URI]/api/data/v9.2/$metadata#accounts(name,primarycontactid,primarycontactid
/$ref())",
    "value": [
        {
            "@odata.etag": "W/\"80649578\"",
            "name": "Litware, Inc. (sample)",
            "_primarycontactid_value": "70bf4d48-34cb-ed11-b596-
0022481d68cd",
            "accountid": "78914942-34cb-ed11-b596-0022481d68cd",
            "primarycontactid": {
                "@odata.id": "[Organization
URI]/api/data/v9.2/contacts(70bf4d48-34cb-ed11-b596-0022481d68cd)"
            }
        },
        {
            "@odata.etag": "W/\"80649580\"",
            "name": "Adventure Works (sample)",
            "_primarycontactid_value": "72bf4d48-34cb-ed11-b596-
0022481d68cd",
            "accountid": "7a914942-34cb-ed11-b596-0022481d68cd",
            "primarycontactid": {
                "@odata.id": "[Organization
URI]/api/data/v9.2/contacts(72bf4d48-34cb-ed11-b596-0022481d68cd)"
            }
        }
    ],
    "@odata.nextLink": "[Organization URI]/api/data/v9.2/accounts?
$select=name%0A&$expand=primarycontactid/$ref&$skiptoken=%3Ccookie%20pagenum
ber=%22%22%20pagingcookie=%22%253ccookie%2520page%253d%25221%2522%253e%253c
accountid%2520last%253d%2522%257b7A914942-34CB-ED11-B596-
0022481D68CD%257d%2522%2520first%253d%2522%257b78914942-34CB-ED11-B596-
0022481D68CD%257d%2522%2520%252f%253e%253c%252fcookie%253e%22%20istracking=%
22False%22%20/%3E"
}

```

Solo puede usar la opción `/$ref` con propiedades de navegación de un único valor. Si la usa con una propiedad de navegación con valor de colección, obtiene el error siguiente:

JSON

```
{
    "error": {
        "code": "0x80060888",
        "message": "Expand with $ref is only supported on lookup type
navigation property."
}
```

```
    }  
}
```

Expansión anidada de propiedades de navegación de un único valor

Puede expandir las propiedades de navegación de un único valor a varios niveles, anidando una opción `$expand` en otra opción `$expand`.

La siguiente consulta devuelve registros de `task` y expande el `contact` relacionado, la `account` relacionada con el `contact` y el `systemuser` que creó el registro `account`:

Solicitud:

HTTP

```
GET [Organization URI]/api/data/v9.2/tasks?$select=subject  
&$expand=regardingobjectid_contact_task($select=fullname;  
$expand=parentcustomerid_account($select=name;  
$expand=createdby($select=fullname)))  
Accept: application/json  
OData-MaxVersion: 4.0  
OData-Version: 4.0
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK  
Content-Type: application/json; odata.metadata=minimal  
Preference-Applied: odata.maxpagesize=2  
OData-Version: 4.0  
  
{  
    "@odata.context": "[Organization  
URI]/api/data/v9.2/$metadata#tasks(subject,regardingobjectid_contact_task(fu  
llname,parentcustomerid_account(name,createdby(fullname))))",  
    "value": [  
        {  
            "@odata.etag": "W/\\"80730855\\\"",  
            "subject": "Task 1 for Susanna Stubberod",  
            "activityid": "e9a8c72c-dbcc-ed11-b597-000d3a993550",  
            "regardingobjectid_contact_task": {  
                "fullname": "Susanna Stubberod (sample)",  
                "contactid": "70bf4d48-34cb-ed11-b596-0022481d68cd",  
                "parentcustomerid_account": {  
                    "name": "Litware, Inc. (sample)",  
                    "accountid": "78914942-34cb-ed11-b596-0022481d68cd",  
                }  
            }  
        }  
    ]  
}
```

```

        "createdby": {
            "fullname": "System Administrator",
            "systemuserid": "4026be43-6b69-e111-8f65-
78e7d1620f5e",
            "ownerid": "4026be43-6b69-e111-8f65-78e7d1620f5e"
        }
    }
},
{
    "@odata.etag": "W/\\"80730861\\\"",
    "subject": "Task 2 for Susanna Stubberod",
    "activityid": "c206f534-dbcc-ed11-b597-000d3a993550",
    "regardingobjectid_contact_task": {
        "fullname": "Susanna Stubberod (sample)",
        "contactid": "70bf4d48-34cb-ed11-b596-0022481d68cd",
        "parentcustomerid_account": {
            "name": "Litware, Inc. (sample)",
            "accountid": "78914942-34cb-ed11-b596-0022481d68cd",
            "createdby": {
                "fullname": "System Administrator",
                "systemuserid": "4026be43-6b69-e111-8f65-
78e7d1620f5e",
                "ownerid": "4026be43-6b69-e111-8f65-78e7d1620f5e"
            }
        }
    }
},
],
"@odata.nextLink": "[Organization URI]/api/data/v9.2/tasks?
$select=subject&$expand=regardingobjectid_contact_task($select=fullname;$exp
and=parentcustomerid_account($select=name;$expand=createdby($select=fullname
)))&$skiptoken=%3Ccookie%20pagenumber=%222%22%20pagingcookie=%22%253ccookie%
2520page%253d%25221%2522%253e%253cactivityid%2520last%253d%2522%257bC206F534
-DBCC-ED11-B597-000D3A993550%257d%2522%2520first%253d%2522%257bE9A8C72C-
DBCC-ED11-B597-
000D3A993550%257d%2522%2520%252f%253e%253c%252fcookie%253e%22%20istracking=%
22False%22%20/%3E"
}

```

Expandir propiedades de navegación valoradas como colección

Hay algunas diferencias importantes en la respuesta que dependen de si usa anidado `$expand` con una propiedad de navegación con valor de colección en cualquier parte de su consulta.

	\$expand anidado	\$expand único
Paging	Paginación en filas expandidas.	Paginación solo en recurso de EntitySet . Las direcciones URL <code><property name>@odata.nextLink</code> de las filas expandidas no incluyen información de paginación.
Se admiten	No \$top o \$orderby	Sí

\$expand único en propiedades de navegación valoradas como colección

Si usa solo un nivel `$expand`, no se aplica paginación a las filas expandidas. Si incluye el encabezado de solicitud `Prefer: odata.maxpagesize`, la paginación solo se aplica al recurso de EntitySet de la consulta.

Cada propiedad de navegación expandida con valor de colección devuelve una URL `<property>@odata.nextLink` que no incluye información de paginación. Es una URL que representa la [colección filtrada](#) para la relación con las opciones de consulta adjuntas. Puede usar esa URL para enviar una solicitud `GET` por separado y devolver las mismas filas que se devolvieron en su solicitud original. Puede aplicar paginación a esa solicitud.

Debido a que no se aplica paginación a los registros expandidos, se pueden devolver hasta 5000 registros relacionados por cada propiedad de navegación con valor de colección expandida. Dependiendo de sus datos y la consulta, podría ser una gran cantidad de datos. Devolver esa cantidad de datos podría afectar el rendimiento y posiblemente provocar que se agote el tiempo de espera de su solicitud. Tenga cuidado con las consultas que redacta. Puede usar las opciones `$top`, `$filter` y `$orderby` para controlar el número total de registros devueltos.

El siguiente ejemplo incluye una sola expansión de `Account_Tasks` y `contact_customer_accounts` mientras recupera registros de cuenta. El encabezado de solicitud `Prefer: odata.maxpagesize=1` garantiza que solo se devuelva un registro de cuenta en la primera página.

Solicitud:

```
HTTP
GET [Organization URI]/api/data/v9.2/accounts?$select=name,accountid
&$expand=Account_Tasks($select=subject),contact_customer_accounts($select=fu
llname)
```

```
Prefer: odata.maxpagesize=1
Accept: application/json
OData-MaxVersion: 4.0
OData-Version: 4.0
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
Content-Type: application/json; odata.metadata=minimal
Preference-Applied: odata.maxpagesize=1
OData-Version: 4.0

{
    "@odata.context": "[Organization
URI]/api/data/v9.2/$metadata#accounts(name,accountid,Account_Tasks(subject),
contact_customer_accounts(fullname))",
    "value": [
        {
            "@odata.etag": "W/\"80649578\"",
            "name": "Litware, Inc. (sample)",
            "accountid": "78914942-34cb-ed11-b596-0022481d68cd",
            "Account_Tasks": [
                {
                    "@odata.etag": "W/\"80730894\"",
                    "subject": "Task 1 for Litware",
                    "_regardingobjectid_value": "78914942-34cb-ed11-b596-
0022481d68cd",
                    "activityid": "be9f6557-e2cc-ed11-b597-000d3a993550"
                },
                {
                    "@odata.etag": "W/\"80730903\"",
                    "subject": "Task 2 for Litware",
                    "_regardingobjectid_value": "78914942-34cb-ed11-b596-
0022481d68cd",
                    "activityid": "605dbd65-e2cc-ed11-b597-000d3a993550"
                },
                {
                    "@odata.etag": "W/\"80730909\"",
                    "subject": "Task 3 for Litware",
                    "_regardingobjectid_value": "78914942-34cb-ed11-b596-
0022481d68cd",
                    "activityid": "a718856c-e2cc-ed11-b597-000d3a993550"
                }
            ],
            "Account_Tasks@odata.nextLink": "[Organization
URI]/api/data/v9.2/accounts(78914942-34cb-ed11-b596-
0022481d68cd)/Account_Tasks?$select=subject",
            "contact_customer_accounts": [
                {
                    "@odata.etag": "W/\"80648695\"",
                    "fullname": "Susanna Stubberod (sample)"
                }
            ]
        }
    ]
}
```

```
        "_parentcustomerid_value": "78914942-34cb-ed11-b596-  
0022481d68cd",  
        "contactid": "70bf4d48-34cb-ed11-b596-0022481d68cd"  
    }  
],  
    "contact_customer_accounts@odata.nextLink": "[Organization  
URI]/api/data/v9.2/accounts(78914942-34cb-ed11-b596-  
0022481d68cd)/contact_customer_accounts?$select=fullname"  
}  
],  
    "@odata.nextLink": "[Organization URI]/api/data/v9.2/accounts?  
$select=name,accountid&$expand=Account_Tasks($select=subject),contact_custom  
er_accounts($select=fullname)&$skiptoken=%3Ccookie%20pagenumber=%22%22%20pa  
gingcookie=%22%253ccookie%2520page%253d%25221%2522%253e%253caccountid%2520la  
st%253d%2522%257b7A914942-34CB-ED11-B596-  
0022481D68CD%257d%2522%2520first%253d%2522%257b78914942-34CB-ED11-B596-  
0022481D68CD%257d%2522%2520%252f%253e%253c%252fc(cookie%253e%22%20istracking=%  
22False%22%20/%3E"  
}
```

Compare esta respuesta con el siguiente ejemplo, que incluye un anidado `$expand`. Desplace la respuesta de ejemplo horizontalmente para ver que solo la URL `@odata.nextLink` del resultado de la cuenta contiene información de paginación.

\$expand anidado en propiedades de navegación valoradas como colección

Si usa un `$expand` anidado en cualquier parte de su consulta y ha incluido el encabezado de solicitud `Prefer: odata.maxpagesize`, la paginación se aplica a cada una de las colecciones expandidas.

Cada propiedad de navegación expandida con valor de colección devuelve una URL `<property>@odata.nextLink` que incluye información de paginación. Puede usar esa URL para enviar una solicitud `GET` por separado y devolverá el siguiente conjunto de registros que no se incluyeron en su solicitud original.

No puede usar las opciones `$top` o `$orderby` para limitar la cantidad total de registros devueltos con un `$expand` anidado. Se devuelve el siguiente error si utiliza estas opciones:

JSON

```
{  
    "error": {  
        "code": "0x80060888",  
        "message": "Only $select and $filter clause can be provided while  
doing $expand on many-to-one relationship or nested one-to-many
```

```
relationship."  
}  
}
```

El siguiente ejemplo se basa en el ejemplo anterior y utiliza los mismos datos. La única diferencia es la adición en la URL de este `$expand` anidado en una propiedad de navegación de un solo valor para devolver el usuario propietario del contacto:

```
; $expand=owninguser($select=fullname).
```

Solicitud:

HTTP

```
GET [Organization URI]/api/data/v9.2/accounts?$select=name,accountid  
&$expand=Account_Tasks($select=subject),contact_customer_accounts($select=fu  
llname;  
$expand=owninguser($select=fullname))  
Prefer: odata.maxpagesize=1  
Accept: application/json  
OData-MaxVersion: 4.0  
OData-Version: 4.0
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK  
Content-Type: application/json; odata.metadata=minimal  
Preference-Applied: odata.maxpagesize=1  
OData-Version: 4.0  
  
{  
    "@odata.context": "[Organization  
URI]/api/data/v9.2/$metadata#accounts(name,accountid,Account_Tasks(subject),  
contact_customer_accounts(fullname,owninguser(fullname)))",  
    "value": [  
        {  
            "@odata.etag": "W/\\"80649578\\\"",  
            "name": "Litware, Inc. (sample)",  
            "accountid": "78914942-34cb-ed11-b596-0022481d68cd",  
            "Account_Tasks": [  
                {  
                    "subject": "Task 1 for Litware",  
                    "activityid": "be9f6557-e2cc-ed11-b597-000d3a993550"  
                }  
            ],  
            "Account_Tasks@odata.nextLink": "[Organization  
URI]/api/data/v9.2/accounts(78914942-34cb-ed11-b596-  
0022481d68cd)/Account_Tasks?  
$select=subject,description&$skiptoken=%3Ccookie%20pagenumber=%222%22%20pagi
```

```

ngcookie=%22%253ccookie%2520page%253d%25221%2522%2520countOfRecords%253d%252
21%2522%253e%253cactivityid%2520last%253d%2522%257bbe9f6557-e2cc-ed11-b597-
000d3a993550%257d%2522%2520first%253d%2522%257bbe9f6557-e2cc-ed11-b597-
000d3a993550%257d%2522%2520%252f%253e%253c%252fcookie%253e%22%20istracking=%
22False%22%20/%3E",
    "contact_customer_accounts": [
        {
            "fullname": "Susanna Stubberod (sample)",
            "contactid": "70bf4d48-34cb-ed11-b596-0022481d68cd",
            "owninguser": {
                "fullname": "System Administrator",
                "systemuserid": "4026be43-6b69-e111-8f65-
78e7d1620f5e",
                "ownerid": "4026be43-6b69-e111-8f65-78e7d1620f5e"
            }
        }
    ],
    "contact_customer_accounts@odata.nextLink": "[Organization
URI]/api/data/v9.2/accounts(78914942-34cb-ed11-b596-
0022481d68cd)/contact_customer_accounts?
$select=fullname&$expand=owninguser($select=fullname)&$skiptoken=%3Ccookie%2
0pagenumber=%22%22%20pagingcookie=%22%253ccookie%2520page%253d%25221%2522%2
520countOfRecords%253d%25221%2522%253e%253ccontactid%2520last%253d%2522%257b
70bf4d48-34cb-ed11-b596-
0022481d68cd%2522%2520first%253d%2522%257b70bf4d48-34cb-ed11-b596-
0022481d68cd%257d%2522%2520%252f%253e%253c%252fcookie%253e%22%20istracking=%
22False%22%20/%3E"
    }
],
"@odata.nextLink": "[Organization URI]/api/data/v9.2/accounts?
$select=name,accountid&$expand=Account_Tasks($select=subject,description),co
ntact_customer_accounts($select=fullname;$expand=owninguser($select=fullname
))&$skiptoken=%3Ccookie%20pagenumber=%22%22%20pagingcookie=%22%253ccookie%2
520page%253d%25221%2522%2520countOfRecords%253d%25221%2522%253e%253caccounti
d%2520last%253d%2522%257b78914942-34cb-ed11-b596-
0022481d68cd%257d%2522%2520first%253d%2522%257b78914942-34cb-ed11-b596-
0022481d68cd%257d%2522%2520%252f%253e%253c%252fcookie%253e%22%20istracking=%
22False%22%20/%3E"
}

```

Compare esta respuesta con el ejemplo anterior, que no incluye un anidado `$expand`. En esta respuesta, el encabezado de solicitud `Prefer: odata.maxpagesize=1` se aplica a los registros `task` devueltos con `Account_Tasks`. Solo se devuelve una tarea en lugar de tres. La URL `Account_Tasks@odata.nextLink` devuelve las siguientes dos tareas. Desplace la respuesta de ejemplo horizontalmente para ver que las direcciones URL `Account_Tasks@odata.nextLink`, `contact_customer_accounts@odata.nextLink` y `@odata.nextLink` contienen información de paginación.

Ordenar filas

Use la [opción de consulta](#) `$orderby` para especificar el orden en el que se devuelven los elementos. Use el sufijo `asc` o `desc` para especificar orden ascendente o descendente, respectivamente. El predeterminado es ascendente. El ejemplo siguiente recupera las propiedades `name` y `revenue` de las cuentas, ordenadas por orden de `revenue` ascendente y por `name` descendente:

```
HTTP  
  
GET [Organization URI]/api/data/v9.2/accounts?$select=name,revenue  
&$orderby=revenue asc,name desc  
&$filter=revenue ne null
```

Filtrar filas

Utilice la [opción de consulta](#) `$filter` para filtrar una colección de recursos.

Dataverse evalúa cada recurso de la colección mediante el conjunto de expresiones para `$filter`. Solo se devuelven en la respuesta los registros en los que la expresión se evalúa como `true`. Los registros no se devuelven si la expresión se evalúa como `false` o `null`, o si el usuario no tiene acceso de lectura al registro.

La siguiente tabla describe los operadores y funciones que puede usar en expresiones `$filter`.

	Descripción	Más información
Operadores de comparación	Use los operadores <code>eq</code> , <code>ne</code> , <code>gt</code> , <code>ge</code> , <code>lt</code> , y <code>le</code> para comparar una propiedad y un valor.	Operadores de comparación
Operadores lógicos	Use <code>and</code> , <code>or</code> y <code>not</code> para crear expresiones más complejas.	Operadores lógicos
Agrupación de operadores	Use paréntesis: <code>()</code> , para especificar la precedencia para evaluar una expresión compleja.	Agrupación de operadores
Funciones de consulta de OData	Evaluar valores de cadena usando las funciones <code>contains</code> , <code>endswith</code> y <code>startswith</code>	Usar funciones de consulta de OData
Funciones de consulta de Dataverse	Utilice más de 60 funciones especializadas diseñadas para aplicaciones comerciales.	Funciones de consulta de Dataverse

	Descripción	Más información
Expresiones Lambda	Cree expresiones basadas en valores de colecciones relacionadas.	Filtrar usando valores de colecciones relacionadas

Si usa una [propiedad de búsqueda](#) en un `$filter`, también puede usar una [colección filtrada](#) con la propiedad de navegación con valor de colección correspondiente. Por ejemplo, estas dos consultas devuelven los mismos resultados:

```
accounts?$filter=_owninguser_value eq '<systemuserid value>'&$select=name
```

```
systemusers(<systemuserid value>)/user_accounts?$select=name
```

Operadores de comparación

La siguiente tabla describe los operadores que puede usar para comparar una propiedad y un valor.

Operator	Descripción	Ejemplo
<code>eq</code>	Es igual a	<code>\$filter=revenue eq 100000</code>
<code>ne</code>	No es igual a	<code>\$filter=revenue ne 100000</code>
<code>gt</code>	Mayor que	<code>\$filter=revenue gt 100000</code>
<code>ge</code>	Mayor o igual que	<code>\$filter=revenue ge 100000</code>
<code>lt</code>	Menor que	<code>\$filter=revenue lt 100000</code>
<code>le</code>	Menor o igual que	<code>\$filter=revenue le 100000</code>

Comparación de columnas

Puede usar operadores de comparación para comparar valores de propiedad en la misma fila; es decir, para [comparar columnas](#). Solo se pueden usar operadores de comparación para comparar valores en la misma fila, y los tipos de columna deben coincidir. Por ejemplo, la siguiente consulta devuelve todos los contactos donde `firstname` es igual a `lastname`:

HTTP
<pre>GET [Organization URI]/api/data/v9.2/contacts? \$select=fullname&\$filter=firstname eq lastname</pre>

Operadores lógicos

La siguiente tabla describe los operadores lógicos que puede usar para crear expresiones más complejas.

Operator	Descripción	Ejemplo
and	Lógico y	<code>\$filter=revenue lt 100000 and revenue gt 2000</code>
or	Disyunción lógica	<code>\$filter=contains(name,'(sample)') or contains(name,'test')</code>
not	Negación lógica	<code>\$filter=not contains(name,'sample')</code>

Agrupación de operadores

Use paréntesis: () con operadores lógicos para especificar la precedencia para evaluar una expresión compleja; por ejemplo:

```
$filter=(contains(name,'sample') or contains(name,'test')) and revenue gt 5000
```

Funciones de consulta de Dataverse

Utilice más de 60 funciones especializadas diseñadas para aplicaciones comerciales. Estas funciones proporcionan capacidades especiales, como se describe en la siguiente tabla.

Grupo	Funciones
Fechas	InFiscalPeriod , InFiscalPeriodAndYear , InFiscalYear , InOrAfterFiscalPeriodAndYear , InOrBeforeFiscalPeriodAndYear , Last7Days , LastFiscalPeriod , LastFiscalYear , LastMonth , LastWeek , LastXDays , LastXFiscalPeriods , LastXFiscalYears , LastXHours , LastXMonths , LastXWeeks , LastXYears , LastYear , Next7Days , NextFiscalPeriod , NextFiscalYear , NextMonth , NextWeek , NextXDays , NextXFiscalPeriods , NextXFiscalYears , NextXHours , NextXMonths , NextXWeeks , NextXYears , NextYear , OlderThanXDays , OlderThanXHours , OlderThanXMinutes , OlderThanXMonths , OlderThanXWeeks , OlderThanXYears , On , OnOrAfter , OnOrBefore , ThisFiscalPeriod , ThisFiscalYear , ThisMonth , ThisWeek , ThisYear , Today , Tomorrow , Yesterday
Valores de id.	EqualBusinessId , EqualUserId , NotEqualBusinessId , NotEqualUserId
Jerarquía	Above , AboveOrEqual , EqualUserOrUserHierarchy , EqualUserOrUserHierarchyAndTeams , EqualUserOrUserTeams ,

Grupo	Funciones
	EqualUserTeams , NotUnder , Under , UnderOrEqual Más información: Consultar datos jerárquicos
Columnas de elecciones	ContainValues , DoesNotContainValues Más información: Consultar datos a partir de opciones
Entre	Between , NotBetween
En	In , NotIn
Lenguaje	EqualUserLanguage

(!) Nota

La función **Contains** es para usar con columnas que tienen indexación de texto completo. Solo la tabla Dynamics 365 KBArticle (artículo) tiene columnas que tienen indexación de texto completo. Use en su lugar la función `contains` de OData.

[Web API Query Function Reference](#) tiene la lista completa. Cada artículo proporciona un ejemplo de sintaxis que puede copiar.

Debe usar el *nombre completo de la función* y agregar el [espacio de nombres del servicio](#) (`Microsoft.Dynamics.CRM`) al nombre de la función.

Cada función tiene un parámetro `PropertyName` que especifica la propiedad a evaluar. La función puede tener más parámetros, como `PropertyValue`, `PropertyValues` o `PropertyValue1` y `PropertyValue2`. Cuando existen estos parámetros, debe proporcionar un valor o valores para comparar con el parámetro `PropertyName`.

El siguiente ejemplo muestra los usos de la función [Between](#) para buscar cuentas con entre 5 y 2000 empleados.

HTTP

```
GET [Organization URI]/api/data/v9.2/accounts?$select=name,numberofemployees
&$filter=Microsoft.Dynamics.CRM.Between(PropertyName='numberofemployees',Pro
pertyValues=[ "5", "2000" ])
```

Filtrar usando valores de cadena

Tenga en cuenta los siguientes puntos cuando filtre valores de cadena:

- Todos los filtros que usen valores de cadena distinguen mayúsculas de minúsculas.

- Debe codificar como URL los caracteres especiales en los criterios de filtro. Más información: [codificar caracteres especiales de URL](#)
- Puede utilizar caracteres comodín, pero evite usarlos al final. Más información: [Uso de caracteres comodín](#)
- Puede utilizar las funciones de consulta de OData: `contains`, `startswith` y `endswith`. Más información: [Usar las funciones de consulta de OData](#)
- Debe administrar comillas simples cuando use filtros que acepten una matriz de valores de cadena. Más información: [Administrar comillas simples](#)

URL codifica los caracteres especiales

Si la cadena que está utilizando como valor en una función de filtro incluye un carácter especial, debe codificarlo como URL. Por ejemplo, si usa esta función:

`contains(name, '+123')`, no funcionará porque `+` es un carácter que no se puede incluir en una URL. Si codifica la cadena como URL, se convertirá en `contains(name, '%2B123')` y obtendrá resultados donde el valor de la columna contiene `+123`.

La siguiente tabla muestra los valores codificados de URL para caracteres especiales comunes.

Especial carácter	URL de codificación carácter
<code>\$</code>	<code>%24</code>
<code>&</code>	<code>%26</code>
<code>+</code>	<code>%2B</code>
<code>,</code>	<code>%2C</code>
<code>/</code>	<code>%2F</code>
<code>:</code>	<code>%3A</code>
<code>;</code>	<code>%3B</code>
<code>=</code>	<code>%3D</code>
<code>?</code>	<code>%3F</code>
<code>@</code>	<code>%40</code>

Usar caracteres comodín

Al componer filtros usando cadenas, puede aplicar los siguientes caracteres comodín:

Caracteres	Descripción	Documentación y ejemplos de T-SQL
%	Coincide con cualquier cadena de cero o más caracteres. Este carácter comodín se puede utilizar como prefijo o como sufijo.	Carácter de porcentaje (Comodín - Caracteres que deben coincidir) (Transact-SQL)
_	Utilice el carácter de subrayado para hacer coincidir cualquier carácter único en una operación de comparación de cadenas que implique la coincidencia de patrones.	_ (Comodín - Coincidir con un carácter) (Transact-SQL)
[]	Hace que coincida con cualquier carácter único dentro del rango o conjunto especificado que se especifica entre paréntesis.	[] (Comodín - Coincidir con caracteres) (Transact-SQL)
[^]	Hace que coincida con cualquier carácter único que no esté dentro del rango o conjunto especificado que se especifica entre paréntesis cuadrados.	[^] (Comodín - Los caracteres no coinciden) (Transact-SQL)

Más información: [Usar caracteres comodín en condiciones para valores de cadena](#)

No se admiten comodines finales

Es importante no utilizar comodines finales porque no se admiten. Las consultas que utilizan estos antipatrones presentan problemas de rendimiento porque las consultas no se pueden optimizar. Estos son algunos ejemplos de comodines finales:

```
startswith(name, '%value')
endswith(name, 'value%')
```

Usar funciones de consulta de OData

La siguiente tabla describe las funciones de consulta de OData que puede usar para filtrar valores de cadena:

Function	Ejemplo
contains	\$filter=contains(name, '(sample)')
endswith	\$filter=endswith(name, 'Inc.')

Function	Ejemplo
startswith	\$filter=startswith(name, 'a')

Puede usar estas funciones con el operador lógico `not` para negar el resultado.

Administrar comillas simples

Algunos filtros aceptan una matriz de valores de cadena, como la función [In Query](#). Al especificar valores en estos filtros que contienen una comilla o apóstrofo, como `O'Brian` o `Men's clothes`, debe usar comillas dobles alrededor de los valores, por ejemplo:

HTTP

```
GET [Organization URI]/api/data/v9.2/contacts?$select=fullname
&$filter=Microsoft.Dynamics.CRM.In(PropertyName=@p1,PropertyValues=@p2)
&@p1='lastname'
&@p2=["OBrian","OBryan","O'Brian","O'Bryan"]
```

Si no lo hace, obtiene el siguiente error: `Invalid JSON. A comma character ',' was expected in scope 'Array'. Every two elements in an array and properties of an object must be separated by commas.`

Si el filtro es para un solo valor, reemplace el carácter de comilla simple con dos caracteres de comilla simple consecutivos; por ejemplo:

HTTP

```
GET [Organization URI]/api/data/v9.2/contacts?$select=fullname
&$filter=lastname eq 'O''Bryan'
```

Si no lo hace, obtiene un error como este: `There is an unterminated literal at position 21 in 'lastname eq 'O'Bryan''.`

Filtrar en función de los valores de datos relacionados

Puede filtrar las filas devueltas en función de los valores de las tablas relacionadas. La forma de filtrar depende del tipo de relación.

Filtrar usando valores de propiedad de columna de búsqueda

Puede filtrar en función de los valores en las propiedades de navegación de un solo valor que representan las columnas de búsqueda. Use este patrón:

```
<single-valued navigation property>/<property name>
```

El siguiente ejemplo devuelve registros de cuentas en función del valor de la columna `primarycontactid/fullname`:

Solicitud:

HTTP

```
GET [Organization URI]/api/data/v9.2/accounts?
$filter=primarycontactid/fullname eq 'Susanna Stubberod (sample)'
&$select=name,_primarycontactid_value
Accept: application/json
OData-MaxVersion: 4.0
OData-Version: 4.0
Prefer: odata.include-
annotations="OData.Community.Display.V1.FormattedValue"
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0
Preference-Applied: odata.include-
annotations="OData.Community.Display.V1.FormattedValue"

{
    "@odata.context": "[Organization
URI]/api/data/v9.2/$metadata#accounts(name,_primarycontactid_value)",
    "value": [
        {
            "@odata.etag": "W/\\"81359849\\",
            "name": "Litware, Inc. (sample)",

            "_primarycontactid_value@OData.Community.Display.V1.FormattedValue":
            "Susanna Stubberod (sample)",
            "_primarycontactid_value": "70bf4d48-34cb-ed11-b596-
0022481d68cd",
            "accountid": "78914942-34cb-ed11-b596-0022481d68cd"
        }
    ]
}
```

También puede comparar valores más arriba en la jerarquía de propiedades de navegación de un solo valor.

El siguiente ejemplo devuelve la primera cuenta donde el registro de contacto representa `primarycontactid`, donde 'System Administrador' creó el registro, usando `primarycontactid/createdby/fullname` en el `$filter`.

Solicitud:

HTTP

```
GET [Organization URI]/api/data/v9.2/accounts?
$filter=primarycontactid/createdby/fullname eq 'System Administrator'
&$select=name,_primarycontactid_value
&$expand=primarycontactid(
    $select=fullname,_createdby_value;
    $expand=createdby($select=fullname))
&$top=1
Accept: application/json
OData-MaxVersion: 4.0
OData-Version: 4.0
Prefer: odata.include-
annotations="OData.Community.Display.V1.FormattedValue"
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0
Preference-Applied: odata.include-
annotations="OData.Community.Display.V1.FormattedValue"

{
    "@odata.context": "[Organization
URI]/api/data/v9.2/$metadata#accounts(name,_primarycontactid_value,primaryco
ntactid(fullname,_createdby_value,createdby(fullname)))",
    "value": [
        {
            "@odata.etag": "W/\\"81359849\\\"",
            "name": "Litware, Inc. (sample)",

            "_primarycontactid_value@OData.Community.Display.V1.FormattedValue":
            "Susanna Stubberod (sample)",
            "_primarycontactid_value": "70bf4d48-34cb-ed11-b596-
0022481d68cd",
            "accountid": "78914942-34cb-ed11-b596-0022481d68cd",
            "primarycontactid": {
                "fullname": "Susanna Stubberod (sample)",
```

```
"_createdby_value@odata.Community.Display.V1.FormattedValue": "System Administrator",
    "_createdby_value": "4026be43-6b69-e111-8f65-78e7d1620f5e",
    "contactid": "70bf4d48-34cb-ed11-b596-0022481d68cd",
    "createdby": {
        "fullname": "System Administrator",
        "systemuserid": "4026be43-6b69-e111-8f65-78e7d1620f5e",
        "ownerid": "4026be43-6b69-e111-8f65-78e7d1620f5e"
    }
}
]
}
```

Filtrar usando valores de colecciones relacionadas

Utilice los *operadores Lambda* `any` y `all` para evaluar valores en una colección para filtrar los resultados.

- `any`: devuelve `true` si la expresión booleana aplicada es true para cualquier miembro de la colección; si no, devuelve false. El operador `any` sin un argumento devuelve `true` si la colección no está vacía.
- `all`: devuelve true se la expresión booleana aplicada es true todos los miembros de la colección; si no, devuelve false.

La sintaxis tiene el siguiente aspecto:

```
<collection>/[any | all](o:<expression to evaluate>)
```

En este caso, `o` es la variable que representa los elementos de la colección. La convención es usar la primera letra del tipo. En la expresión, utilice `o/<property or collection name>` para hacer referencia a una propiedad o colección de un elemento determinado.

Puede incluir condiciones en varias propiedades de navegación con valores de colección y colecciones anidadas. No puede incluir condiciones en las propiedades de navegación con valores de colección que están anidadas en una propiedad de navegación de búsqueda. Por ejemplo,

```
$filter=primarycontactid/new_contact_account/any(a:a/accountid eq '{GUID}')
```

 no se admite.

Más información: [Usar operadores Lambda en odata.org](#)

Ejemplos de operadores Lambda

El ejemplo siguiente recupera todos los registros de cuenta de cuenta que tienen al menos un correo electrónico con "sometext" en el asunto:

HTTP

```
GET [Organization URI]/api/data/v9.2/accounts?$select=name  
&$filter=Account_Emails/any(e:contains(e/subject, 'sometext'))  
Accept: application/json  
OData-MaxVersion: 4.0  
OData-Version: 4.0
```

El ejemplo a continuación recupera todos los registros de entidad de cuenta que tienen todas las tareas asociadas cerradas:

HTTP

```
GET [Organization URI]/api/data/v9.2/accounts?$select=name  
&$filter=Account_Tasks/all(t:t/statecode eq 1)  
Accept: application/json  
OData-MaxVersion: 4.0  
OData-Version: 4.0
```

El ejemplo a continuación recupera todos los registros de entidad de cuenta que tienen al menos un correo electrónico con "sometext" en el asunto y cuyo código de estado es activo:

HTTP

```
GET [Organization URI]/api/data/v9.2/accounts?$select=name  
&$filter=Account_Emails/any(e:contains(e/subject, 'sometext') and  
e/statecode eq 0)  
Accept: application/json  
OData-MaxVersion: 4.0  
OData-Version: 4.0
```

El siguiente ejemplo crea una consulta anidada usando los operadores `any` y `all`:

HTTP

```
GET [Organization URI]/api/data/v9.2/accounts?$select=name  
&$filter=(contact_customer_accounts/any(c:c/jobtitle eq 'jobtitle' and  
c/opportunity_customer_contacts/any(o:o/description ne 'N/A'))) and  
endswith(name, 'Inc.')  
Accept: application/json
```

```
OData-MaxVersion: 4.0  
OData-Version: 4.0
```

Resultados de página

Utilice el encabezado de solicitud `Prefer: odata.maxpagesize` para controlar la cantidad de registros devueltos. Si no especifica un número, se pueden devolver hasta 5000 registros por cada solicitud. No puede solicitar un tamaño de página superior a 5000.

ⓘ Nota

Dataverse no admite la opción de consulta `$skip`, por lo que no puede usar la combinación de `$top` y `$skip` para la paginación. Para obtener más información:
[Usar la opción de consulta \\$top](#)

El siguiente ejemplo devuelve solo los dos primeros registros de contacto:

Solicitud:

```
HTTP  
  
GET [Organization URI]/api/data/v9.2/contacts?$select=fullname  
Accept: application/json  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
Prefer: odata.maxpagesize=2
```

Respuesta:

```
HTTP  
  
HTTP/1.1 200 OK  
OData-Version: 4.0  
Preference-Applied: odata.maxpagesize=2  
  
{  
    "@odata.context": "[Organization  
URI]/api/data/v9.2/$metadata#contacts(fullname)",  
    "value": [  
        {  
            "@odata.etag": "W/\\"72201545\\\"",  
            "fullname": "Yvonne McKay (sample)",  
            "contactid": "49b0be2e-d01c-ed11-b83e-000d3a572421"  
        },  
        {  
            "@odata.etag": "W/\\"80648695\\\"",  
            "fullname": "John Doe (sample)",  
            "contactid": "5f3a2a2e-000d-4a00-8000-000d3a572421"  
        }  
    ]  
}
```

```

        "fullname": "Susanna Stubberod (sample)",
        "contactid": "70bf4d48-34cb-ed11-b596-0022481d68cd"
    }
],
"@odata.nextLink": "[Organization URI]/api/data/v9.2/contacts?
$select=fullname&$skiptoken=%3Ccookie%20pagenumber=%222%22%20pagingcookie=%2
2%253ccookie%2520page%253d%25221%2522%253e%253ccontactid%2520last%253d%2522%
257bD5026A4D-D01C-ED11-B83E-
000D3A572421%257d%2522%2520first%253d%2522%257b49B0BE2E-D01C-ED11-B83E-
000D3A572421%257d%2522%2520%252f%253e%253c%252fcookie%253e%22%20istracking=%
22False%22%20/%3E"
}

```

Cuando hay más registros de los solicitados, la anotación `@odata.nextLink` proporciona una URL que puede usar con `GET` para devolver la siguiente página de datos, como se muestra en el siguiente ejemplo:

Solicitud:

HTTP

```

GET [Organization URI]/api/data/v9.2/contacts?
$select=fullname&$skiptoken=%3Ccookie%20pagenumber=%222%22%20pagingcookie=%2
2%253ccookie%2520page%253d%25221%2522%253e%253ccontactid%2520last%253d%2522%
257bD5026A4D-D01C-ED11-B83E-
000D3A572421%257d%2522%2520first%253d%2522%257b49B0BE2E-D01C-ED11-B83E-
000D3A572421%257d%2522%2520%252f%253e%253c%252fcookie%253e%22%20istracking=%
22False%22%20/%3E
Accept: application/json
OData-MaxVersion: 4.0
OData-Version: 4.0
Prefer: odata.maxpagesize=2

```

Respuesta:

HTTP

```

HTTP/1.1 200 OK
OData-Version: 4.0
Preference-Applied: odata.maxpagesize=2

{
    "@odata.context": "[Organization
URI]/api/data/v9.2/$metadata#contacts(fullname)",
    "value": [
        {
            "@odata.etag": "W/\\"80648710\\\"",
            "fullname": "Nancy Anderson (sample)",
            "contactid": "72bf4d48-34cb-ed11-b596-0022481d68cd"
        },
        {

```

```
        "@odata.etag": "W/\\"80648724\\",
        "fullname": "Maria Campbell (sample)",
        "contactid": "74bf4d48-34cb-ed11-b596-0022481d68cd"
    }
],
"@odata.nextLink": "[Organization URI]/api/data/v9.2/contacts?
$select=fullname&$skiptoken=%3Ccookie%20pagenumber=%223%22%20pagingcookie=%2
2%253ccookie%2520page%253d%25222%2522%253e%253ccontactid%2520last%253d%2522%
257bF2318099-171F-ED11-B83E-
000D3A572421%257d%2522%2520first%253d%2522%257bBB55F942-161F-ED11-B83E-
000D3A572421%257d%2522%2520%252f%253e%253c%252fcookie%253e%22%20istracking=%
22False%22%20/%3E"
}
```

Debe almacenar en caché los resultados devueltos o el valor de URL `@odata.nextLink` y utilizarlos para volver a las páginas anteriores.

No cambie el valor de la URL `@odata.nextLink` ni le anexe opciones de consulta. Para cada solicitud posterior de páginas adicionales, debe usar el mismo valor de preferencia de `odata.maxpagesize` que usó en la solicitud original. Puede continuar paginando los datos hasta que no se incluya ninguna anotación `@odata.nextLink` en los resultados.

En los ejemplos anteriores, la información codificada se estableció como el valor del parámetro `$skiptoken` en el valor de URL `@odata.nextLink`. El servidor establece esta información codificada para controlar la paginación. No debe modificar la información codificada ni codificarla más. Utilice el valor de URL proporcionado para recuperar la página siguiente.

Usar la opción de consulta `$top`

Use la opción de consulta `$top` para limitar el número de resultados devueltos. No use `$top` con el encabezado de solicitud `Prefer: odata.maxpagesize`. Si incluye ambos, `$top` se ignora.

El siguiente ejemplo solo devuelve las primeras tres filas de cuenta:

HTTP

```
GET [Organization URI]/api/data/v9.2/accounts?$select=name,revenue&$top=3
```

Agregar datos

Use la opción `$apply` para agregar y agrupar los datos dinámicamente.

Las funciones agregadas están limitados a una colección de 50.000 registros. La información adicional acerca de usar la funcionalidad agregada con Dataverse se puede encontrar aquí: [Uso de FetchXML para agregación](#).

Puede encontrar más información sobre la agregación de datos de OData aquí: [Extensión de OData para la versión 4.0 de agregación de datos](#). Dataverse solo admite un subconjunto de estos métodos agregados.

ⓘ Nota

No se admite `groupby` con los valores `datetime`.

Estos son algunos ejemplos:

- Lista de estados únicos en la consulta
- Recuento por valores de estado
- Suma agregada de ingresos
- Ingresos medios basados en el estado
- Suma de los ingresos basados en el estado
- Recuento total de ingresos por nombre de contacto principal
- Nombres de contacto primario para cuentas de "WA"
- Fecha y hora de registro creado por última vez
- Fecha y hora de registro creado por primera vez

Estos ejemplos no muestran la solicitud y la respuesta completas por motivos de brevedad.

Lista de estados únicos en la consulta

HTTP

```
GET accounts?$apply=groupby((statuscode))
Prefer: odata.include-
annotations="OData.Community.Display.V1.FormattedValue"
```

Cuerpo de respuesta

JSON

```
{
  "@odata.context": "[Organization URI]/api/data/v9.2/$metadata#accounts",
  "value": [
    {
      "statuscode@OData.Community.Display.V1.FormattedValue":
```

```
"Active",
    "statuscode": 1
},
{
    "statuscode@OData.Community.Display.V1.FormattedValue":
"Inactive",
    "statuscode": 2
}
]
```

Recuento por valores de estado

HTTP

```
GET accounts?$apply=groupby((statuscode),aggregate($count as count))
Prefer: odata.include-
annotations="OData.Community.Display.V1.FormattedValue"
```

Cuerpo de respuesta

JSON

```
{
    "@odata.context": "[Organization URI]/api/data/v9.2/$metadata#accounts",
    "value": [
        {
            "statuscode@OData.Community.Display.V1.FormattedValue":
"Active",
            "statuscode": 1,
            "count@OData.Community.Display.V1.FormattedValue": "8",
            "count": 8
        },
        {
            "statuscode@OData.Community.Display.V1.FormattedValue":
"Inactive",
            "statuscode": 2,
            "count@OData.Community.Display.V1.FormattedValue": "1",
            "count": 1
        }
    ]
}
```

Suma agregada de ingresos

HTTP

```
GET accounts?$apply=aggregate(revenue with sum as total)
Prefer: odata.include-
annotations="OData.Community.Display.V1.FormattedValue"
```

Cuerpo de respuesta

JSON

```
{
    "@odata.context": "[Organization URI]/api/data/v9.2/$metadata#accounts",
    "value": [
        {
            "total@OData.Community.Display.V1.FormattedValue": "$440,000.00",
            "total": 440000.000000000
        }
    ]
}
```

Ingresos medios basados en el estado

HTTP

```
GET accounts?$apply=groupby((statuscode),aggregate(revenue with average as
averagevalue))
Prefer: odata.include-
annotations="OData.Community.Display.V1.FormattedValue"
```

Cuerpo de respuesta

JSON

```
{
    "@odata.context": "[Organization URI]/api/data/v9.2/$metadata#accounts",
    "value": [
        {
            "statuscode@OData.Community.Display.V1.FormattedValue": "Active",
            "statuscode": 1,
            "averagevalue@OData.Community.Display.V1.FormattedValue": "$53,750.00",
            "averagevalue": 53750.000000000
        },
        {
            "statuscode@OData.Community.Display.V1.FormattedValue": "Inactive",
            "statuscode": 2,
            "averagevalue@OData.Community.Display.V1.FormattedValue": "
```

```
        "$10,000.00",
        "averagevalue": 10000.00000000
    }
]
}
```

Suma de los ingresos basados en el estado

HTTP

```
GET accounts?$apply=groupby((statuscode),aggregate(revenue with sum as total))
Prefer: odata.include-
annotations="OData.Community.Display.V1.FormattedValue"
```

Cuerpo de respuesta

JSON

```
{
    "@odata.context": "[Organization URI]/api/data/v9.2/$metadata#accounts",
    "value": [
        {
            "statuscode@OData.Community.Display.V1.FormattedValue": "Active",
            "statuscode": 1,
            "total@OData.Community.Display.V1.FormattedValue": "$430,000.00",
            "total": 430000.00000000
        },
        {
            "statuscode@OData.Community.Display.V1.FormattedValue": "Inactive",
            "statuscode": 2,
            "total@OData.Community.Display.V1.FormattedValue": "$10,000.00",
            "total": 10000.00000000
        }
    ]
}
```

Recuento total de ingresos por nombre de contacto principal

HTTP

```
GET accounts?$apply=groupby((primarycontactid/fullname),aggregate(revenue with sum as total))
```

```
Prefer: odata.include-  
annotations="OData.Community.Display.V1.FormattedValue"
```

Cuerpo de respuesta

JSON

```
{  
    "@odata.context": "[Organization URI]/api/data/v9.2/$metadata#accounts",  
    "value": [  
        {  
            "total@OData.Community.Display.V1.FormattedValue": "$10,000.00",  
            "total": 10000.00000000,  
            "contact_fullname": "Jim Glynn (sample)"  
        },  
        {  
            "total@OData.Community.Display.V1.FormattedValue": "$80,000.00",  
            "total": 80000.00000000,  
            "contact_fullname": "Maria Campbell (sample)"  
        },  
        ... <truncated for brevity>  
    ]  
}
```

Nombres de contacto primario para cuentas de "WA"

HTTP

```
GET accounts?$apply=filter(address1_stateorprovince eq  
'WA')/groupby((primarycontactid/fullname))
```

Cuerpo de respuesta

JSON

```
{  
    "@odata.context": "[Organization URI]/api/data/v9.2/$metadata#accounts",  
    "value": [  
        {  
            "contact_fullname": "Rene Valdes (sample)"  
        },  
        {  
            "contact_fullname": "Robert Lyon (sample)"  
        },  
        {  
            "contact_fullname": "Scott Konersmann (sample)"  
        }  
    ]  
}
```

```
    ]  
}
```

Fecha y hora de registro creado por última vez

HTTP

```
GET accounts??$apply=aggregate(createdon with max as lastCreate)  
Prefer: odata.include-  
annotations="OData.Community.Display.V1.FormattedValue"
```

Cuerpo de respuesta

JSON

```
{  
  "@odata.context": "[Organization URI]/api/data/v9.2/$metadata#accounts",  
  "value": [  
    {  
      "lastCreate@OData.Community.Display.V1.FormattedValue":  
        "3/25/2023 10:42 AM",  
      "lastCreate": "2023-03-25T17:42:47Z"  
    }  
  ]  
}
```

Fecha y hora de registro creado por primera vez

HTTP

```
GET accounts?$apply=aggregate(createdon with min as firstCreate)  
Prefer: odata.include-  
annotations="OData.Community.Display.V1.FormattedValue"
```

Cuerpo de respuesta

JSON

```
{  
  "@odata.context": "[Organization URI]/api/data/v9.2/$metadata#accounts",  
  "value": [  
    {  
      "firstCreate@OData.Community.Display.V1.FormattedValue":  
        "3/25/2023 10:42 AM",  
      "firstCreate": "2023-03-25T17:42:46Z"  
    }  
  ]  
}
```

```
        ]  
    }
```

Recuento del número de filas

Use la opción de consulta `$count=true` para incluir un recuento de entidades que coincidan con los criterios de filtro, hasta 5000.

Solicitud:

HTTP

```
GET [Organization URI]/api/data/v9.2/accounts?$select=accountid&$count=true  
Accept: application/json  
OData-MaxVersion: 4.0  
OData-Version: 4.0
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK  
OData-Version: 4.0  
  
{  
    "@odata.context": "[Organization  
URI]/api/data/v9.2/$metadata#accounts(accountid)",  
    "@odata.count": 9,  
    "value": [  
        {  
            "@odata.etag": "W/\\"81359849\\\"",  
            "accountid": "78914942-34cb-ed11-b596-0022481d68cd"  
        },  
        ... <Truncated for brevity>  
    ]  
}
```

La anotación de respuesta `@odata.count` contiene el número de filas, hasta 5000, que coinciden con los criterios de filtro con independencia del tamaño de página seleccionado.

ⓘ Nota

Si desea recuperar una instantánea de las últimas 24 horas del número total de filas de una tabla más allá de 5000, use la función `RetrieveTotalRecordCount`.

Si el valor del recuento es 5000 y quiere saber si el recuento es exactamente 5000 o mayor que 5000, puede agregar el siguiente encabezado:

```
Prefer: odata.include-  
annotations="Microsoft.Dynamics.CRM.totalrecordcount,Microsoft.Dynamics.CRM.  
totalrecordcountlimitexceeded"
```

Este encabezado agrega las siguientes anotaciones al resultado:

- `@Microsoft.Dynamics.CRM.totalrecordcount`
- `@Microsoft.Dynamics.CRM.totalrecordcountlimitexceeded`

Cuando se use con la opción de consulta `$count=true` y haya más de 5000 registros, se devuelven los siguientes valores:

```
"@odata.count": 5000,  
"@Microsoft.Dynamics.CRM.totalrecordcount": 5000,  
"@Microsoft.Dynamics.CRM.totalrecordcountlimitexceeded": true,
```

Si hay menos de 5000 registros, se devuelve el recuento real.

```
"@odata.count": 58,  
"@Microsoft.Dynamics.CRM.totalrecordcount": 58,  
"@Microsoft.Dynamics.CRM.totalrecordcountlimitexceeded": false,
```

Si no incluye la opción `$count=true`, el valor `@Microsoft.Dynamics.CRM.totalrecordcount` total será `-1`.

El siguiente ejemplo muestra que hay 10 cuentas que coinciden con `$filter`, pero solo se devuelven las tres primeras cuentas:

Solicitud:

HTTP

```
GET [Organization URI]/api/data/v9.2/accounts?$select=name?  
&$filter=contains(name,'sample')  
&$count=true  
Accept: application/json  
OData-MaxVersion: 4.0
```

```
OData-Version: 4.0
Prefer: odata.maxpagesize=3
Prefer: odata.include-annotations="Microsoft.Dynamics.CRM.*"
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0
Preference-Applied: odata.maxpagesize=3
Preference-Applied: odata.include-annotations="Microsoft.Dynamics.CRM.*"

{
    "@odata.context": "[Organization
URI]/api/data/v9.2/$metadata#accounts(name)",
    "@odata.count": 10,
    "@Microsoft.Dynamics.CRM.totalrecordcount": 5000,
    "@Microsoft.Dynamics.CRM.totalrecordcountlimitexceeded": true,
    "value": [
        {
            "@odata.etag": "W/\"502482\"",
            "name": "Fourth Coffee (sample)",
            "accountid": "655eaf89-f083-e511-80d3-00155d2a68d3"
        },
        {
            "@odata.etag": "W/\"502483\"",
            "name": "Litware, Inc. (sample)",
            "accountid": "675eaf89-f083-e511-80d3-00155d2a68d3"
        },
        {
            "@odata.etag": "W/\"502484\"",
            "name": "Adventure Works (sample)",
            "accountid": "695eaf89-f083-e511-80d3-00155d2a68d3"
        }
    ],
    "@odata.nextLink": "[Organization URI]/api/data/v9.2/accounts?
$select=name&$filter=contains(name, 'sample')&$skiptoken=%3Ccookie%20pagenumber=%222%22%20pagingcookie=%22%253ccookie%2520page%253d%25221%2522%253e%253ccountid%2520last%253d%2522%257b695EAF89-F083-E511-80D3-00155D2A68D3%257d%2522%2520first%253d%2522%257b655EAF89-F083-E511-80D3-00155D2A68D3%257d%2522%2520%252f%253e%253c%252fcookie%253e%22%20istracking=%22False%22%20/%3E"
}
```

Para obtener solo un número que represente el conteo de una colección, agregue `/$count`, como en el siguiente ejemplo:

Solicitud:

HTTP

```
GET [Organization URI]/api/data/v9.2/accounts/$count
Accept: application/json
OData-MaxVersion: 4.0
OData-Version: 4.0
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
Content-Type: text/plain
OData-Version: 4.0

10
```

Consulte también

[Buscar en los datos de la tabla usando la búsqueda de Dataverse](#)

[Trabajar con el límite de elementos de búsqueda de Búsqueda rápida](#)

[Ejemplo de datos de consulta de la API web \(C#\)](#)

[Ejemplo de datos de consulta de la API web \(JavaScript del lado del cliente\)](#)

[Realizar operaciones mediante la API web](#)

[Componer solicitudes HTTP y administrar errores](#)

Usar FetchXML con API web

Artículo • 25/08/2023

FetchXML es un lenguaje de consulta patentado que puede recuperar y agregar datos mediante la API web y el SDK de .NET. Más información: [Usar FetchXML para consultar datos](#).

ⓘ Nota

A diferencia de las consultas que utilizan la sintaxis de OData, las consultas de FetchXML no devuelven propiedades con valores `null`.

Puede redactar una consulta FetchXML para una tabla específica. Luego, codifique en URL el XML y use el parámetro `fetchXml` de cadena de consulta para pasarlo al conjunto de entidades.

Por ejemplo, el siguiente FetchXML tiene `account` como la entidad:

XML

```
<fetch mapping='logical'>
  <entity name='account'>
    <attribute name='accountid'/>
    <attribute name='name' />
    <attribute name='accountnumber' />
  </entity>
</fetch>
```

Este FetchXML tiene el siguiente valor codificado en URL:

text

```
%3Cfetch%20mapping%3D%27logical%27%3E%3Centity%20name%3D%27account%27%3E%3Ca
ttribute%20name%3D%27accountid%27%2F%3E%3Cattribute%20name%3D%27name%27%2F%3
E%3Cattribute%20name%3D%27accountnumber%27%2F%3E%3C%2Fentity%3E%3C%2Ffetch%3
E
```

La mayoría de los lenguajes de programación incluyen una función para codificar como dirección URL una cadena.

- En JavaScript, usa la función [encodeURI](#).
- En .NET, puede usar el método [System.NET.WebUtility.UrlEncode\(String\)](#)

Debe codificar como dirección URL cualquier solicitud que envíe a cualquier servicio web RESTful. Si pega una dirección URL en la barra de direcciones del explorador, debe codificar como dirección URL la dirección automáticamente.

El siguiente ejemplo muestra una solicitud `GET` mediante el FetchXML anterior con la ruta del conjunto de entidades para `accounts`. Pasa el XML codificado usando este parámetro: `?fetchXml=`

Solicitud:

HTTP

```
GET [Organization URI]/api/data/v9.2/accounts?
fetchXml=%3Cfetch%20mapping%3D%27logical%27%3E%3Centity%20name%3D%27account%
27%3E%3Cattribute%20name%3D%27accountid%27%2F%3E%3Cattribute%20name%3D%27nam%
e%27%2F%3E%3Cattribute%20name%3D%27accountnumber%27%2F%3E%3C%2Fentity%3E%3C%
2Ffetch%3E HTTP/1.1
Accept: application/json
OData-MaxVersion: 4.0
OData-Version: 4.0
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0

{
  "@odata.context": "[Organization
URI]/api/data/v9.2/$metadata#accounts(accountid,name)", "value": [
    {
      "@odata.etag": "W/\"506678\"",
      "accountid": "89390c24-9c72-e511-80d4-00155d2a68d1",
      "name": "Fourth Coffee (sample)",
      "accountnumber": "1234",
    },
    {
      "@odata.etag": "W/\"502172\"",
      "accountid": "8b390c24-9c72-e511-80d4-00155d2a68d1",
      "name": "Litware, Inc. (sample)"
    },
    {
      "@odata.etag": "W/\"502174\"",
      "accountid": "8d390c24-9c72-e511-80d4-00155d2a68d1",
      "name": "Adventure Works (sample)"
    },
    {
      "@odata.etag": "W/\"506705\"",
      "accountid": "8f390c24-9c72-e511-80d4-00155d2a68d1",
      "name": "Fabrikam, Inc. (sample)"
    }
  ]
}
```

```
]  
}
```

Recuerde que las propiedades con valores nulos no se incluyen en los resultados devueltos mediante FetchXML. En este ejemplo, solo el primer registro devuelto tiene un valor `accountnumber`.

Paginación con FetchXML

Con FetchXML, puede aplicar paginación simple estableciendo los atributos `page` y `count` del elemento `fetch`. Por ejemplo, las siguientes cuentas de consultas fetchXML, limita el número de entidades a 2 y devuelve solamente la primera página:

XML

```
<fetch mapping="logical"  
    page="1"  
    count="2">  
    <entity name="account">  
        <attribute name="accountid" />  
        <attribute name="name" />  
        <attribute name="industrycode" />  
        <order attribute="name" />  
    </entity>  
</fetch>
```

Paginar grandes conjuntos de resultados

Cuando trabaja con grandes conjuntos de resultados que alcanzan el límite de paginación de 5000, el uso de cookies de paginación con la consulta ayuda a mejorar el rendimiento. Solicite una cookie de paginación como anotación. Utilice el encabezado de solicitud `prefer: odata.include-annotations` para usar o incluir `Microsoft.Dynamics.CRM.fetchxmlpagingcookie` y se devuelve una anotación `@Microsoft.Dynamics.CRM.fetchxmlpagingcookie` con el resultado.

La siguiente serie de solicitudes de FetchXML muestra el uso de cookies de paginación. Este ejemplo utiliza un pequeño valor `count` (3) por brevedad. Normalmente no usaría cookies de paginación para tamaños de página tan pequeños.

XML

```
<fetch page='1'  
      count='3'  
      paging-cookie=''
```

```

mapping='logical'
output-format='xml-platform'
version='1.0'
distinct='false'
<entity name ='contact'>
  <attribute name ='fullname' />
  <attribute name ='jobtitle' />
  <attribute name ='annualincome' />
  <order descending ='true'
    attribute='fullname' />
  <filter type ='and'>
    <condition value ='%(sample)%'
      attribute='fullname'
      operator='like' />
    <condition value ='18717e9c-643f-ed11-9db0-002248225e95'
      attribute='parentcustomerid'
      operator='eq' />
  </filter>
</entity>
</fetch>

```

Primera página

Envíe la primera página con el conjunto de valores `page` en `'1'`. Utilice el encabezado de solicitud `Prefer: odata.include-annotations="*"` para asegurarse de que se devuelvan las anotaciones necesarias en la respuesta.

Solicitud:

HTTP

```

GET [Organization Uri]/api/data/v9.2/contacts?
fetchXml=%3Cfetch+page%3D%221%22+count%3D%223%22+mapping%3D%22logical%22+out
put-format%3D%22xml-
platform%22+version%3D%221.0%22+distinct%3D%22false%22%3E%0D%0A++%3Centity+n
ame%3D%22contact%22%3E%0D%0A++++%3Cattribute+name%3D%22fullname%22+%2F%3E%0D
%0A++++%3Cattribute+name%3D%22jobtitle%22+%2F%3E%0D%0A++++%3Cattribute+name%3D%
22annualincome%22+%2F%3E%0D%0A++++%3Corder+descending%3D%22true%22+attrib
ute%3D%22fullname%22+%2F%3E%0D%0A++++%3Cfilter+type%3D%22and%22%3E%0D%0A++++
++%3Ccondition+value%3D%2225(sample)%25%22+attribute%3D%22fullname%22+opera
tor%3D%22like%22+%2F%3E%0D%0A++++++%3Ccondition+value%3D%2218717e9c-643f-
ed11-9db0-
002248225e95%22+attribute%3D%22parentcustomerid%22+operator%3D%22eq%22+%2F%3
E%0D%0A++++%3C%2Ffilter%3E%0D%0A++%3C%2Fentity%3E%0D%0A%3C%2Ffetch%3E&$count
=true HTTP/1.1
Prefer: odata.include-annotations="*"
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0
Preference-Applied: odata.include-annotations="*"

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#contacts(fullname,jobtitle,annualincome,_transa
ctioncurrencyid_value,transactioncurrencyid,contactid,transactioncurrencyid(
))",
    "@odata.count": 8,
    "@Microsoft.Dynamics.CRM.totalrecordcount": 8,
    "@Microsoft.Dynamics.CRM.totalrecordcountlimitexceeded": false,
    "@Microsoft.Dynamics.CRM.globalmetadataversion": "74343461",
    "@Microsoft.Dynamics.CRM.fetchxmlpagingcookie": "<cookie pagename=\"2\""
pagingcookie=\"%253ccookie%2520page%253d%25221%2522%253e%253cfullname%2520la
st%253d%2522Robert%2520Lyon%2520%2528sample%2529%2522%2520first%253d%2522Sus
anna%2520Stubberod%2520%2528sample%2529%2522%2520%252f%253e%253ccontactid%25
20last%253d%2522%257b30717E9C-643F-ED11-9DB0-
002248225E95%257d%2522%2520first%253d%2522%257b20717E9C-643F-ED11-9DB0-
002248225E95%257d%2522%2520%252f%253e%253c%252fcookie%253e\""
istracking=\\"False\\" />",
    "@Microsoft.Dynamics.CRM.morerecords": true,
    "value": [
        {
            "@odata.etag": "W/\"74359676\"",
            "fullname": "Susanna Stubberod (sample)",
            "jobtitle": "Senior Purchaser",
            "annualincome@OData.Community.Display.V1.FormattedValue":
"$52,000.00",
            "annualincome": 52000.0,
            "contactid": "20717e9c-643f-ed11-9db0-002248225e95"
        },
        {
            "@odata.etag": "W/\"74359706\"",
            "fullname": "Scott Konersmann (sample)",
            "jobtitle": "Accounts Manager",
            "annualincome@OData.Community.Display.V1.FormattedValue":
"$38,000.00",
            "annualincome": 38000.0,
            "contactid": "2c717e9c-643f-ed11-9db0-002248225e95"
        },
        {
            "@odata.etag": "W/\"74359716\"",
            "fullname": "Robert Lyon (sample)",
            "jobtitle": "Senior Technician",
            "annualincome@OData.Community.Display.V1.FormattedValue":
"$78,000.00",
            "annualincome": 78000.0,
            "contactid": "30717e9c-643f-ed11-9db0-002248225e95"
        }
    ]
}
```

```
]  
}
```

En la respuesta, el valor de anotación `@Microsoft.Dynamics.CRM.morerecords` indica que existen más registros que coinciden con los criterios.

El valor de anotación `@Microsoft.Dynamics.CRM.fetchxmlpagingcookie` proporciona la información de paginación sobre el registro devuelto. El valor `@Microsoft.Dynamics.CRM.fetchxmlpagingcookie` es un documento XML. Debe usar el valor de atributo `pagingcookie` de ese documento en la siguiente solicitud.

El valor del atributo `pagingcookie` se codifica dos veces mediante URL. El valor descodificado tiene este aspecto:

XML

```
<cookie page="1"><fullname last="Robert Lyon (sample)" first="Susanna  
Stubberod (sample)" /><contactid last="{30717E9C-643F-ED11-9DB0-  
002248225E95}" first="{20717E9C-643F-ED11-9DB0-002248225E95}" /></cookie>
```

Siguientes páginas

En todas las solicitudes posteriores donde el valor de anotación de la página anterior `@Microsoft.Dynamics.CRM.morerecords` indica que existen más registros, debe:

1. Incrementar el valor de atributo `page` del elemento `fetch`.
2. Decodificar la URL del valor de atributo `pagingcookie` dos veces.
3. Codificar en XML el valor de atributo `pagingcookie` decodificado y establecerlo como el valor de un atributo `paging-cookie` en el elemento `fetch`.

Si debe codificar XML explícitamente el valor puede depender de la tecnología que utilice. En .NET, se puede hacer por usted cuando establece el valor XML en un atributo de otro elemento XML.

4. URL codifica todo el valor de FetchXML como lo hizo en la primera solicitud.

En la siguiente solicitud, el FetchXML tiene este aspecto antes de codificarse como URL:

XML

```
<fetch page="2" count="3" mapping="logical" output-format="xml-platform"  
version="1.0" distinct="false" paging-cookie="<cookie  
page="1">&lt;fullname last="Robert Lyon (sample)"&gt;&lt;/fullna
```

```

first="Susanna Stubberod (sample)" &gt;&lt;contactid last="
{30717E9C-643F-ED11-9DB0-002248225E95}" first="{20717E9C-643F-
ED11-9DB0-002248225E95}" &gt;&lt;/cookie&gt;>
<entity name="contact">
<attribute name="fullname" />
<attribute name="jobtitle" />
<attribute name="annualincome" />
<order descending="true" attribute="fullname" />
<filter type="and">
<condition value="%{(sample)}" attribute="fullname" operator="like" />
<condition value="18717e9c-643f-ed11-9db0-002248225e95"
attribute="parentcustomerid" operator="eq" />
</filter>
</entity>
</fetch>

```

Solicitud:

HTTP

```

GET [Organization Uri]/api/data/v9.2/contacts?
fetchXml=%3Cfetch+page%3D%22%22+count%3D%223%22+mapping%3D%22logical%22+out
put-format%3D%22xml-
platform%22+version%3D%221.0%22+distinct%3D%22false%22+paging-
cookie%3D%22%26lt%3Bcookie+page%3D%26quot%3B1%26quot%3B%26gt%3B%26lt%3Bfulln
ame+last%3D%26quot%3BRobert+Lyon+
(sample)%26quot%3B+first%3D%26quot%3BSusanna+Stubberod+
(sample)%26quot%3B+%2F%26gt%3B%26lt%3Bcontactid+last%3D%26quot%3B%7B30717E9C
-643F-ED11-9DB0-002248225E95%7D%26quot%3B+first%3D%26quot%3B%7B20717E9C-
643F-ED11-9DB0-
002248225E95%7D%26quot%3B+%2F%26gt%3B%26lt%3B%2Fcookie%26gt%3B%22%3E%0D%0A+++
%3Centity+name%3D%22contact%22%3E%0D%0A++++%3Cattribute+name%3D%22fullname%2
2%2F%3E%0D%0A++++%3Cattribute+name%3D%22jobtitle%22%2F%3E%0D%0A++++%3Cattr
ibute+name%3D%22annualincome%22%2F%3E%0D%0A++++%3Corder+descending%3D%22tru
e%22+attribute%3D%22fullname%22%2F%3E%0D%0A++++%3Cfilter+type%3D%22and%22%3
E%0D%0A++++++%3Ccondition+value%3D%22%25(sample)%25%22+attribute%3D%22fullna
me%22+operator%3D%22like%22%2F%3E%0D%0A++++++%3Ccondition+value%3D%2218717e
9c-643f-ed11-9db0-
002248225e95%22+attribute%3D%22parentcustomerid%22+operator%3D%22eq%22%2F%3
E%0D%0A++++%3C%2Ffilter%3E%0D%0A++%3C%2Fentity%3E%0D%0A%3C%2Ffetch%3E&$count
=true HTTP/1.1
Prefer: odata.include-annotations="*"
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

```

Respuesta:

HTTP

HTTP/1.1 200 OK
OData-Version: 4.0
Preference-Applied: odata.include-annotations="*"

```
{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#contacts(fullname,jobtitle,annualincome,_transa
ctioncurrencyid_value,transactioncurrencyid,contactid,transactioncurrencyid(
))",
    "@odata.count": 8,
    "@Microsoft.Dynamics.CRM.totalrecordcount": 8,
    "@Microsoft.Dynamics.CRM.totalrecordcountlimitexceeded": false,
    "@Microsoft.Dynamics.CRM.globalmetadataversion": "74343461",
    "@Microsoft.Dynamics.CRM.fetchxmlpagingcookie": "<cookie pagename=\"2\""
pagingcookie=\"%253ccookie%2520page%253d%2522%2522%253e%253cfullname%2520la
st%253d%2522Nancy%2520Anderson%2520%2528sample%2529%2522%2520first%253d%2522
Rene%2520Valdes%2520%2528sample%2529%2522%2520%252f%253e%253ccontactid%25201
ast%253d%2522%257b24717E9C-643F-ED11-9DB0-
002248225E95%257d%2522%2520first%253d%2522%257b38717E9C-643F-ED11-9DB0-
002248225E95%257d%2522%2520%252f%253e%253c%252fcookie%253e\""
istracking=\\"False\\" />",
    "@Microsoft.Dynamics.CRM.morerecords": true,
    "value": [
        {
            "@odata.etag": "W/\"74359736\"",
            "fullname": "Rene Valdes (sample)",
            "jobtitle": "Data Analyst III",
            "annualincome@OData.Community.Display.V1.FormattedValue":
"$86,000.00",
            "annualincome": 86000.0,
            "contactid": "38717e9c-643f-ed11-9db0-002248225e95"
        },
        {
            "@odata.etag": "W/\"74359726\"",
            "fullname": "Paul Cannon (sample)",
            "jobtitle": "Ski Instructor",
            "annualincome@OData.Community.Display.V1.FormattedValue":
"$68,500.00",
            "annualincome": 68500.0,
            "contactid": "34717e9c-643f-ed11-9db0-002248225e95"
        },
        {
            "@odata.etag": "W/\"74359686\"",
            "fullname": "Nancy Anderson (sample)",
            "jobtitle": "Activities Manager",
            "annualincome@OData.Community.Display.V1.FormattedValue":
"$55,500.00",
            "annualincome": 55500.0,
            "contactid": "24717e9c-643f-ed11-9db0-002248225e95"
        }
    ]
}
```

Última página

En la página final, las anotaciones `@Microsoft.Dynamics.CRM.morerecords` y `@Microsoft.Dynamics.CRM.fetchxmlpagingcookie` no se incluyen en la respuesta.

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/contacts?
fetchXml=%3Cfetch+page%3D%223%22+count%3D%223%22+mapping%3D%22logical%22+out-
put-format%3D%22xml-
platform%22+version%3D%221.0%22+distinct%3D%22false%22+paging-
cookie%3D%22%26lt%3Bcookie+page%3D%26quot%3B%26quot%3B%26gt%3B%26lt%3Bfulln-
ame+last%3D%26quot%3BNancy+Anderson+
(sample)%26quot%3B+first%3D%26quot%3BRene+Valdes+
(sample)%26quot%3B+%2F%26gt%3B%26lt%3Bcontactid+last%3D%26quot%3B%7B24717E9C-
-643F-ED11-9DB0-002248225E95%7D%26quot%3B+first%3D%26quot%3B%7B38717E9C-
643F-ED11-9DB0-
002248225E95%7D%26quot%3B+%2F%26gt%3B%26lt%3B%2Fcookie%26gt%3B%22%3E%0D%0A++-
%3Centity+name%3D%22contact%22%3E%0D%0A++++%3Cattribute+name%3D%22fullname%2
2%2F%3E%0D%0A++++%3Cattribute+name%3D%22jobtitle%22%2F%3E%0D%0A++++%3Cattri-
bute+name%3D%22annualincome%22%2F%3E%0D%0A++++%3Corder+descending%3D%22tru-
e%22+attribute%3D%22fullname%22%2F%3E%0D%0A++++%3Cfilter+type%3D%22and%22%3
E%0D%0A++++%3Ccondition+value%3D%22%25(sample)%25%22+attribute%3D%22fullna-
me%22+operator%3D%22like%22%2F%3E%0D%0A++++%3Ccondition+value%3D%2218717e
9c-643f-ed11-9db0-
002248225e95%22+attribute%3D%22parentcustomerid%22+operator%3D%22eq%22%2F%3
E%0D%0A++++%3C%2Ffilter%3E%0D%0A++%3C%2Fentity%3E%0D%0A%3C%2Ffetch%3E&$count
=true HTTP/1.1
Prefer: odata.include-annotations="*"
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0
Preference-Applied: odata.include-annotations="*"

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#contacts(fullname,jobtitle,annualincome,_transa-
ctioncurrencyid_value,transactioncurrencyid,contactid,transactioncurrencyid(
))",
  "@odata.count": 8,
  "@Microsoft.Dynamics.CRM.totalrecordcount": 8,
```

```

"@Microsoft.Dynamics.CRM.totalrecordcountlimitexceeded": false,
"@Microsoft.Dynamics.CRM.globalmetadataversion": "74343461",
"value": [
{
    "@odata.etag": "W/\"74359696\"",
    "fullname": "Maria Cambell (sample)",
    "jobtitle": "Accounts Manager",
    "annualincome@odata.Community.Display.V1.FormattedValue":
"$31,000.00",
    "annualincome": 31000.0,
    "contactid": "28717e9c-643f-ed11-9db0-002248225e95"
},
{
    "@odata.etag": "W/\"74359746\"",
    "fullname": "Jim Glynn (sample)",
    "jobtitle": "Senior International Sales Manager",
    "annualincome@odata.Community.Display.V1.FormattedValue":
"$81,400.00",
    "annualincome": 81400.0,
    "contactid": "3c717e9c-643f-ed11-9db0-002248225e95"
}
]
}

```

Use FetchXML en una solicitud de lote

La longitud de una dirección URL en una solicitud `GET` está limitada. Incluir FetchXML como parámetro en la dirección URL pueden alcanzar el límite. Puede ejecutar una operación `$batch` con una solicitud `POST` como método para mover el FetchXML de la dirección URL al cuerpo de la solicitud donde no aplica el límite. Enviar una solicitud de `GET` dentro de un `$batch` permite que las tengan hasta 64 KB (65 536 caracteres). Más que con una solicitud de `GET` normal, pero no es ilimitada. Más información:[Ejecute las operaciones por lotes mediante API web](#)

Ejemplo

Solicitud:

HTTP

```

POST [Organization URI]/api/data/v9.2/$batch HTTP/1.1
Content-Type:multipart/mixed;boundary=batch_AAA123
Accept:application/json
OData-MaxVersion:4.0
OData-Version:4.0

--batch_AAA123
Content-Type: application/http

```

```
Content-Transfer-Encoding: binary
```

```
GET [Organization URI]/api/data/v9.2/accounts?  
fetchXml=%3Cfetch%20mapping='logical'%3E%3Centity%20name='account'%3E%3Cattr  
ibute%20name='accountid'/%3E%3Cattribute%20name='name'/%3E%3Cattribute%20nam  
e='telephone1'/%3E%3Cattribute%20name='accountid'/%3E%3Cattribute%20name='cr  
editonhold'/%3E%3Centity%3E%3C/fetch%3E HTTP/1.1  
Content-Type: application/json  
OData-Version: 4.0  
OData-MaxVersion: 4.0  
  
--batch_AAA123--
```

Respuesta:

JSON

```
--batchresponse_cbfd44cd-a322-484e-913b-49e18af44e34  
Content-Type: application/http  
Content-Transfer-Encoding: binary  
  
HTTP/1.1 200 OK  
Content-Type: application/json; odata.metadata=minimal  
OData-Version: 4.0  
  
{  
  "@odata.context": "[Organization  
URI]/api/data/v9.2/$metadata#accounts(accountid,name,telephone1,creditonhold  
)",  
  "value": [  
    {  
      "@odata.etag": "W/\"563737\"",  
      "accountid": "1f55c679-485e-e811-8151-000d3aa3c22a",  
      "name": "Fourth Coffee (sample)",  
      "telephone1": "+1-425-555-0121",  
      "creditonhold": false  
    },  
    {  
      "@odata.etag": "W/\"563739\"",  
      "accountid": "2555c679-485e-e811-8151-000d3aa3c22a",  
      "name": "Litware, Inc. (sample)",  
      "telephone1": "+1-425-555-0120",  
      "creditonhold": false  
    }  
  ]  
}  
--batchresponse_cbfd44cd-a322-484e-913b-49e18af44e34--
```

Consulte también

[Usar FetchXML para crear una consulta](#)

[Consultar datos utilizando la API web](#)

[Recuperar y ejecutar consultas predefinidas](#)

 **Nota**

¿Puede indicarnos sus preferencias de idioma de documentación? [Realice una breve encuesta.](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad](#)).

Recuperar y ejecutar consultas predefinidas

Artículo • 09/04/2023

Microsoft Dataverse proporciona una forma de que los administradores creen vistas del sistema que están disponibles para todos los usuarios. Los usuarios individuales pueden guardar consultas de búsqueda avanzada para su reutilización en la aplicación. Ambas representan consultas predefinidas que puede recuperar y ejecutar utilizando la API web.

ⓘ Nota

A diferencia de las consultas que utilizan la sintaxis de OData, los datos devueltos de consultas predefinidas o fetchXml no devolverán propiedades con valores `null`. Cuando el valor es `null`, la propiedad no se incluirá en los resultados.

Cuando se devuelve una consulta utilizando la sintaxis de OData, un registro incluirá una propiedad con un valor `null` así:

JSON

```
{  
    "@odata.etag": "W/\"46849433\"",  
    "name": "Contoso, Ltd. (sample)",  
    "accountnumber": null,  
    "accountid": "7a4814f9-b0b8-ea11-a812-000d3a122b89"  
}
```

Cuando se recupera mediante una consulta predefinida o con FetchXml, el mismo registro no incluirá la propiedad `accountnumber` porque es `null`, al igual que:

JSON

```
{  
    "@odata.etag": "W/\"46849433\"",  
    "name": "Contoso, Ltd. (sample)",  
    "accountid": "7a4814f9-b0b8-ea11-a812-000d3a122b89"  
}
```

Consultas predefinidas

Dataverse permite definir, guardar, y ejecutar dos tipos de consultas que se muestran aquí.

Tipo de consulta	Descripción
Consulta guardada	Vistas definidas por el sistema para una tabla (entidad). Estas vistas se almacenan en <code>savedquery EntityType</code> . Más información: Personalizar vistas de tabla
Consulta de usuario	Búsqueda avanzada búsquedas guardadas por los usuarios para una tabla (entidad). Estas vistas se almacenan en <code>userquery EntityType</code> . Más información: Tabla UserQuery (vista guardada)

Los registros para ambos tipos de entidades contienen la definición FetchXML para que se devuelvan los datos. Puede buscar el tipo de entidad respectivo para recuperar el valor de clave principal. Con el valor de clave principal, puede ejecutar la consulta pasando el valor de clave principal. Por ejemplo, para ejecutar la consulta guardada **Cuentas activas**, primero debe obtener la clave principal mediante una consulta como ésta.

HTTP

```
GET [Organization URI]/api/data/v9.0/savedqueries?  
$select=name,savedqueryid&$filter=name eq 'Active Accounts'
```

Luego puede usar el valor de `savedqueryid` y pasarlo como el valor al parámetro `savedQuery` al conjunto de entidades de cuentas.

HTTP

```
GET [Organization URI]/api/data/v9.0/accounts?savedQuery=00000000-0000-0000-  
00aa-000010001002
```

Utilice el mismo enfoque para obtener el `userqueryid` y pasarlo como valor al parámetro `userQuery` al conjunto de entidades que coincide con el correspondiente `returnedtypecode` de la consulta guardada.

HTTP

```
GET [Organization URI]/api/data/v9.0/accounts?userQuery=121c6fd8-1975-e511-  
80d4-00155d2a68d1
```

Aplicar una consulta a cualquier colección del tipo adecuado

Además de simplemente aplicar la consulta guardada a la colección principal del conjunto de entidades, también puede usar una consulta guardada o consulta de usuario para aplicar el mismo filtrado a cualquier colección del tipo adecuado de entidades. Por ejemplo, si desea aplicar una consulta solo a entidades relacionadas con una entidad específica, puede aplicar el mismo patrón. Por ejemplo, la siguiente URL aplicará la consulta **Oportunidades abiertas** a las oportunidades relacionadas con una cuenta específica a través de la propiedad de navegación valorada como colección `opportunity_parent_account`.

HTTP

```
GET [Organization URI]/api/data/v9.0/accounts(8f390c24-9c72-e511-80d4-00155d2a68d1)/opportunity_parent_account/?savedQuery=00000000-0000-0000-00aa-000010003001
```

Consulte también

[Ejemplo de datos de consulta API \(C#\)](#)

[Ejemplo de datos de consulta de la API web \(JavaScript del lado del cliente\)](#)

[Realizar operaciones mediante la API web](#)

[Componer solicitudes HTTP y administrar errores](#)

[Consultar datos utilizando la API web](#)

[Crear una fila de tabla usando la API web](#)

[Recuperar una fila de tabla usando la API web](#)

[Actualizar y eliminar filas de tablas usando la API web](#)

[Asociar y anular la asociación de filas de tabla mediante la API web](#)

[Usar funciones de la API web](#)

[Usar acciones de la API web](#)

[Ejecutar las operaciones por lotes mediante API web](#)

[Suplantar a otro usuario utilizando la API web](#)

[Realizar operaciones condicionales mediante la API web](#)

ⓘ Nota

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta. ↗ (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales
[\(declaración de privacidad ↗\)](#).

Crear una fila de tabla usando la API web

Artículo • 07/12/2023

Use una solicitud `POST` para enviar datos para crear una fila de tabla (registro de entidad). Puede crear varias filas de tabla relacionadas en una sola operación usando *inserción profunda*. También necesita saber cómo establecer valores para asociar una nueva fila de tabla a tablas existentes usando la anotación `@odata.bind`.

ⓘ Nota

Para obtener información sobre cómo crear y actualizar las definiciones de tabla (entidad) mediante la API web, consulte [Crear y actualizar definiciones de tablas mediante la API web](#).

Crear básico

En este ejemplo crea un nuevo registro de entidad de cuenta. `accounts` es el nombre de entidad establecido para [account EntityType](#). El encabezado `OData-EntityId` de respuesta contiene el URI de la entidad creada.

Solicitud:

HTTP

```
POST [Organization URI]/api/data/v9.0/accounts
Content-Type: application/json; charset=utf-8
OData-MaxVersion: 4.0
OData-Version: 4.0
Accept: application/json

{
    "name": "Sample Account",
    "creditonhold": false,
    "address1_latitude": 47.639583,
    "description": "This is the description of the sample account",
    "revenue": 5000000,
    "accountcategorycode": 1
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 No Content
OData-Version: 4.0
OData-EntityId: [Organization URI]/api/data/v9.0/accounts(7eb682f1-ca75-e511-80d4-00155d2a68d1)
```

Para crear un nuevo registro de entidad debe identificar el [nombre del conjunto de entidades](#), y los nombres y los tipos de propiedad válidos. Para todas las tablas y atributos del sistema (columnas de la tabla), puede encontrar esta información en el artículo para esa entidad en [Referencia del tipo de entidad de la API web](#). Para tablas o columnas personalizadas, consulte la definición de esa tabla en [\\$Metadata Document CSDL](#). Más información: [EntityTypes de la API web](#)

Crear con datos devueltos

Puede crear su solicitud `POST` de forma que los datos del registro creado sean devueltos con un estado de `201 (Created)`. Para obtener este resultado, debe usar la preferencia `return=representation` en los encabezados de solicitud.

Para controlar qué propiedades se devuelven, anexe la opción de consulta `$select` a la dirección URL del conjunto de entidades. También puede utilizar `$expand` para devolver entidades relacionadas.

Las propiedades de navegación de `$expand` anidado con valor de colección no devolverán datos cuando se usan con la preferencia `return=representation`. Más información: [\\$expand anidado en propiedades de navegación valorada en una colección](#)

Cuando se crea una entidad de esta forma, el encabezado `OData-EntityId` que contiene el URI al registro creado no se vuelve.

Este ejemplo crea una nueva entidad de cuenta y devuelve los datos solicitados en la respuesta.

Solicitud:

HTTP

```
POST [Organization URI]/api/data/v9.0/accounts?
$select=name,creditonhold,address1_latitude,description,revenue,accountcategorycode,createdon
```

```

OData-MaxVersion: 4.0
OData-Version: 4.0
Accept: application/json
Content-Type: application/json; charset=utf-8
Prefer: return=representation

{
    "name": "Sample Account",
    "creditonhold": false,
    "address1_latitude": 47.639583,
    "description": "This is the description of the sample account",
    "revenue": 5000000
}

```

Respuesta:

HTTP

```

HTTP/1.1 201 Created
Content-Type: application/json; odata.metadata=minimal
Preference-Applied: return=representation
OData-Version: 4.0

{
    "@odata.context": "[Organization
URI]/api/data/v9.0/$metadata#accounts/$entity",
    "@odata.etag": "W/\"536530\"",
    "accountid": "d6f193fc-ce85-e611-80d8-00155d2a68de",
    "accountcategorycode": 1,
    "description": "This is the description of the sample account",
    "address1_latitude": 47.63958,
    "creditonhold": false,
    "name": "Sample Account",
    "createdon": "2016-09-28T22:57:53Z",
    "revenue": 5000000.0000,
    "_transactioncurrencyid_value": "048dddaa-6f7f-e611-80d3-00155db5e0b6"
}

```

Crear múltiples registros en una sola solicitud

La forma más rápida de crear varios registros del mismo tipo en una sola solicitud es utilizar la acción [CreateMultiple](#). En el momento de escribir este artículo, la acción [CreateMultiple](#) es una función de vista previa. No todas las tablas estándar admiten esta acción, pero todas las tablas elásticas sí.

Más información:

- Mensajes de operación en masa
- Muestra: SDK para .NET, Usar operaciones masivas
- Use CreateMultiple con tablas elásticas

Crea filas de tablas relacionadas en una sola operación

Puede crear entidades relacionadas entre sí definiéndolas como valores de propiedades de navegación. Este patrón se conoce como *inserción en profundidad*. Este método tiene dos ventajas. Es más eficiente, porque reemplaza múltiples operaciones más sencillas de creación y asociación con una operación *atómica*. Una operación atómica tiene éxito o falla por completo.

Como con un Crear básico, el encabezado `OData-EntityId` de respuesta contiene el URI de la entidad creada. Los URI de las entidades relacionadas creadas no se devuelven. Puede obtener los valores de clave principal de los registros si utiliza el encabezado `Prefer: return=representation` para que devuelva los valores del registro creado. Más información: [Crear con datos devueltos](#)

Por ejemplo, el siguiente cuerpo de solicitud enviado al conjunto de entidades `accounts` crea un total de cuatro registros en el contexto de creación de una cuenta.

- Se crea un contacto porque se define como una propiedad de objeto de la propiedad de navegación de valor único `primarycontactid`.
- Se crea una oportunidad porque se define como un objeto dentro de una matriz que se establece en el valor de una propiedad de navegación valorada como colección `opportunity_customer_accounts`.
- Se crea una tarea porque se define como un objeto en una matriz que se establece en el valor de una propiedad de navegación valorada como colección `Opportunity_Tasks`.

ⓘ Nota

Cuando crea una nueva fila de tabla, no puede insertar una imagen no primaria al mismo tiempo. Para agregar una imagen no principal, la fila ya debe existir.

Solicitud:

HTTP

```

POST [Organization URI]/api/data/v9.0/accounts
Content-Type: application/json; charset=utf-8
OData-MaxVersion: 4.0
OData-Version: 4.0
Accept: application/json

{
  "name": "Sample Account",
  "primarycontactid": {
    "firstname": "John",
    "lastname": "Smith"
  },
  "opportunity_customer_accounts": [
    {
      "name": "Opportunity associated to Sample Account",
      "Opportunity_Tasks": [
        { "subject": "Task associated to opportunity" }
      ]
    }
  ]
}

```

Respuesta:

HTTP

```

HTTP/1.1 204 No Content
OData-Version: 4.0
OData-EntityId: [Organization URI]/api/data/v9.0/accounts(3c6e4b5f-86f6-e411-80dd-00155d2a68cb)

```

Asociar filas de tabla en la creación

Para asociar nuevos registros nuevos con registros existentes al crearlos, use la anotación `@odata.bind` para establecer el valor de las propiedades de navegación.

El siguiente cuerpo de solicitud enviado al conjunto de entidades de `accounts` crea una nueva cuenta asociada a un contacto existente con el valor `contactid` de `00000000-0000-0000-0000-000000000001` y dos tareas existentes con valores `activityid` de `00000000-0000-0000-0000-000000000002` y `00000000-0000-0000-0000-000000000003`.

Esta solicitud utiliza el encabezado `Prefer: return=representation` para que devuelva los valores del registro creado. Más información: [Crear con datos devueltos](#)

Solicitud:

HTTP

```
POST [Organization URI]/api/data/v9.0/accounts?
$select=name&$expand=primarycontactid($select=fullname),Account_Tasks($select=subject)
Content-Type: application/json; charset=utf-8
OData-MaxVersion: 4.0
OData-Version: 4.0
Accept: application/json
Prefer: return=representation

{
    "name": "Sample Account",
    "primarycontactid@odata.bind": "/contacts(00000000-0000-0000-0000-000000000001)",
    "Account_Tasks@odata.bind": [
        "/tasks(00000000-0000-0000-0000-000000000002)",
        "/tasks(00000000-0000-0000-0000-000000000003)"
    ]
}
```

Respuesta:

HTTP

```
HTTP/1.1 201 Created
OData-Version: 4.0
Preference-Applied: return=representation

{
    "@odata.context": "[Organization
URI]/api/data/v9.1/$metadata#accounts(name,primarycontactid(fullname),Accoun
t_Tasks(subject))/$entity",
    "@odata.etag": "W/\"36236432\"",
    "name": "Sample Account",
    "accountid": "00000000-0000-0000-000000000004",
    "primarycontactid": {
        "@odata.etag": "W/\"28877094\"",
        "fullname": "Yvonne McKay (sample)",
        "contactid": "00000000-0000-0000-000000000001"
    },
    "Account_Tasks": [
        {
            "@odata.etag": "W/\"36236437\""
    }
]
```

```
        "subject": "Task 1",
        "activityid": "00000000-0000-0000-0000-000000000002"
    },
    {
        "@odata.etag": "W/\\"36236440\\",
        "subject": "Task 2",
        "activityid": "00000000-0000-0000-0000-000000000003"
    }
]
```

Comprobar registros duplicados

De forma predeterminada, se suprime la detección de duplicados cuando se crean registros mediante la API web. Para habilitar la detección de duplicados, incluya el encabezado `MSCRM.SuppressDuplicateDetection: false` con la solicitud POST para habilitar la detección de duplicados. Detección de duplicados solo se aplica cuando se cumplen las siguientes condiciones:

- La organización ha habilitado la detección de duplicados.
- La entidad tiene habilitada la detección de duplicados.
- Se aplican las reglas de detección de duplicados activas.

Más información:

- [Detección de datos duplicados con código](#)
- [Detección de datos duplicados con la API web](#)

Crear un registro a partir de otro registro

Utilice la [función InitializeFrom](#) para crear un nuevo registro en el contexto de un registro existente en el que se asigna una relación entre tablas. Para obtener información sobre cómo crear estas asignaciones, consulte:

- [Asignar columnas de tabla](#)
- [Personalizar las asignaciones de tablas y columnas](#)

Para determinar si se pueden asignar dos entidades, utilice la consulta siguiente:

```
GET [Organization URI]/api/data/v9.1/entitymaps?
$select=sourceentityname,targetentityname&$orderby=sourceentityname
```

La creación de un nuevo registro a partir de otro registro es un proceso de dos pasos. Primero, use la función `InitializeFrom` para devolver valores de propiedad asignados desde el registro original. Luego, combine los datos de respuesta devueltos en `InitializeFrom` function con los cambios que desee realizar y luego publique los datos mediante `POST` para crear el registro.

El siguiente ejemplo muestra cómo crear un registro de cuenta utilizando los valores de un registro de cuenta existente con el valor `accountid` igual a `00000000-0000-0000-0000-000000000001`.

Paso 1: Obtenga los datos con `InitializeFrom`

Solicitud:

HTTP

```
GET [Organization  
URI]/api/data/v9.0/InitializeFrom(EntityMoniker=@p1,TargetEntityName=@p2,Tar  
getFieldType=@p3)?@p1={'@odata.id':'accounts(00000000-0000-0000-0000-  
000000000001)'}&@p2='account'&@p3=Microsoft.Dynamics.CRM.TargetFieldType'Val  
idForCreate'  
If-None-Match: null  
OData-Version: 4.0  
OData-MaxVersion: 4.0  
Content-Type: application/json  
Accept: application/json
```

Respuesta:

JSON

```
{  
    "@odata.context": "[Organization  
URI]/api/data/v9.0/$metadata#accounts/$entity",  
    "@odata.type": "#Microsoft.Dynamics.CRM.account",  
    "parentaccountid@odata.bind": "accounts(00000000-0000-0000-0000-  
000000000001)",  
    "transactioncurrencyid@odata.bind": "transactioncurrencies(732e87e1-  
1d96-e711-80e4-00155db75426)",  
    "address1_line1": "123 Maple St.",  
    "address1_city": "Seattle",  
    "address1_country": "United States of America"  
}
```

Paso 2: Crear el nuevo registro

La respuesta recibida de `InitializeFrom` function consta de valores de columnas asignadas entre la tabla de origen y la tabla de destino y el GUID del registro primario. La asignación de columnas entre las tablas que tienen una relación de entidad es diferente para tablas diferentes y se puede personalizar, por lo que la respuesta de la solicitud de `InitializeFrom` function puede variar para organizaciones diferentes.

También pueden establecerse y/o modificarse otros valores de propiedades para el nuevo registro agregándolos en el cuerpo de la solicitud JSON, como se muestra en el siguiente ejemplo:

HTTP

```
POST [Organization URI]/api/data/v9.0/accounts
Content-Type: application/json; charset=utf-8
OData-MaxVersion: 4.0
OData-Version: 4.0
Accept: application/json

{
    "@odata.context": "[Organization
URI]/api/data/v9.0/$metadata#accounts/$entity",
    "@odata.type": "#Microsoft.Dynamics.CRM.account",
    "parentaccountid@odata.bind": "accounts(00000000-0000-0000-0000-
000000000001)",
    "transactioncurrencyid@odata.bind": "transactioncurrencies(732e87e1-
1d96-e711-80e4-00155db75426)",
    "name": "Contoso Ltd",
    "numberofemployees": "200",
    "address1_line1": "100 Maple St.",
    "address1_city": "Seattle",
    "address1_country": "United States of America",
    "fax": "73737"
}
}
```

Crear documentos en particiones de almacenamiento

Si está creando una gran cantidad de registros para tablas elásticas, puede crear las entidades en particiones de almacenamiento para acelerar el acceso a esos registros de entidad.

Más información: [Crear un registro en una tabla elástica](#)

Consulte también

[Ejemplo de operaciones básicas de la API web \(C#\)](#)

[Ejemplo de operaciones básicas de la API web \(JavaScript del lado del cliente\)](#)

[Función InitializeFrom](#)

[Realizar operaciones mediante la API web](#)

[Componer solicitudes HTTP y administrar errores](#)

[Consultar datos utilizando la API web](#)

[Recuperar una fila de tabla usando la API web](#)

[Actualizar y eliminar filas de tablas usando la API web](#)

[Asociar y anular la asociación de filas de tabla mediante la API web](#)

[Usar funciones de la API web](#)

[Usar acciones de la API web](#)

[Ejecutar las operaciones por lotes mediante API web](#)

[Suplantar a otro usuario utilizando la API web](#)

[Realizar operaciones condicionales mediante la API web](#)

 **Nota**

¿Puede indicarnos sus preferencias de idioma de documentación? [Realice una breve encuesta.](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad](#)).

Recuperar una fila de tabla usando la API web

Artículo • 25/08/2023

Use una solicitud `GET` para recuperar datos de un registro especificado como el recurso con un identificador único. Al recuperar una fila de tabla (registro de entidad), también puede solicitar propiedades específicas y expandir las propiedades de navegación para devolver propiedades desde registros relacionados en diferentes tablas.

ⓘ Nota

Para obtener información sobre cómo recuperar definiciones de tablas, consulte [Consultar definiciones de tablas mediante la API web](#).

Recuperación básica

El ejemplo siguiente devuelve datos para un registro de entidad de cuenta con el valor de clave primaria igual a 00000000-0000-0000-0000-000000000001.

HTTP

```
GET [Organization URI]/api/data/v9.2/accounts(00000000-0000-0000-0000-000000000001)
```

Este ejemplo devuelve todas las propiedades del registro de la cuenta, lo que no es una práctica recomendada de rendimiento. Debe usar siempre la opción de consulta del sistema `$select` para limitar las propiedades devueltas al recuperar datos. Esto es especialmente importante cuando se recuperan varias filas de datos. Para obtener más información, consulte [Consultar datos mediante la API web](#).

Recuperar propiedades específicas

Para limitar las propiedades devueltas al recuperar dtos con la solicitud GET, use la opción de consulta del sistema `$select` con una lista separada por comas de nombres de propiedad. Solicitar solo las propiedades que necesita es una práctica recomendada importante de rendimiento. Si no especifica propiedades para devolver, se devolverán todas las propiedades.

El ejemplo siguiente recupera propiedades `name` y `revenue` para la entidad de cuenta con el valor de clave principal igual a 00000000-0000-0000-0000-000000000001

Solicitud:

HTTP

```
GET [Organization URI]/api/data/v9.2/accounts(00000000-0000-0000-0000-000000000001)?$select=name,revenue HTTP/1.1
Accept: application/json
Content-Type: application/json; charset=utf-8
OData-MaxVersion: 4.0
OData-Version: 4.0
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0

{
  "@odata.context": "[Organization
URI]/api/data/v9.2/$metadata#accounts(name,revenue)/$entity",
  "@odata.etag": "W/\"502186\",
  "name": "A. Datum Corporation (sample)",
  "revenue": 10000,
  "accountid": "00000000-0000-0000-0000-000000000001",
  "_transactioncurrencyid_value": "b2a6b689-9a39-e611-80d2-00155db44581"
}
```

Al solicitar determinados tipos de propiedades puede esperar que se devuelvan automáticamente más propiedades de solo lectura.

Si solicita un valor monetario, se devolverá la [propiedad de búsqueda](#) `_transactioncurrencyid_value`. Esta propiedad solo contiene el valor GUID de la divisa de transacción para que pueda usar este valor para recuperar información acerca de la divisa con [transactioncurrency EntityType](#). De forma alternativa, al solicitar anotaciones puede obtener más datos en la misma solicitud.

Si solicita una propiedad que forma parte de un atributo compuesto para una dirección, obtiene también la propiedad compuesta. Por ejemplo, si la consulta solicita la propiedad `address1_line1` para un contacto, se devuelve además la propiedad `address1_composite`.

Recuperar el registro con una clave alternativa

Si una entidad tiene una clave alternativa definida, también puede usar la clave alternativa para recuperar, actualizra o eliminar la entidad. De forma predeterminada, no hay claves alternativas configuradas para entidades. Las claves alternativas sólo están disponibles si su organización o una solución las agrega.

Actualmente no se admiten los valores de clave alternativos con los siguientes caracteres: /, <, >, *, %, &, :, \\, ?, +.

Por ejemplo, si la entidad `contact` tiene una definición de clave alternativa que incluye las propiedades `firstname` y `emailaddress1`, puede recuperar el contacto mediante el uso de una consulta con los datos proporcionados para esas claves:

HTTP

```
GET [Organization  
URI]/api/data/v9.2/contacts(firstname='Joe',emailaddress1='abc@example.com')
```

Si la definición clave alternativa contiene un campo de tipo de búsqueda (por ejemplo, la propiedad `primarycontactid` para la entidad `account`), puede recuperar `account` utilizando las [propiedades de búsqueda](#), como en el ejemplo siguiente:

HTTP

```
GET [Organization  
URI]/api/data/v9.2/accounts(_primarycontactid_value=00000000-0000-0000-0000-  
000000000001)
```

Recuperar documentos en particiones de almacenamiento

Cuando [recupera un registro en una tabla elástica](#) que está almacenado en una partición, asegúrese de especificar la clave de partición.

Recuperar un solo valor de propiedad

Cuando solo necesita recuperar el valor de una sola propiedad, puede agregar el nombre de la propiedad al URI de la entidad. Reducir la cantidad de datos devueltos es una práctica recomendada de rendimiento.

Este ejemplo siguiente devuelve solo el valor de la propiedad `name` para una entidad `account`:

Solicitud:

HTTP

```
GET [Organization URI]/api/data/v9.2/accounts(00000000-0000-0000-0000-000000000001)/name HTTP/1.1
Accept: application/json
OData-MaxVersion: 4.0
OData-Version: 4.0
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0

{
"@odata.context": "[Organization
URI]/api/data/v9.2/$metadata#accounts(00000000-0000-0000-0000-
000000000001)/name",
"value": "Adventure Works (sample)"
}
```

Recuperar el valor sin procesar de una propiedad

Para recuperar el valor bruto de una propiedad primitiva, en lugar de JSON, agregue `/$value` a la URL; Por ejemplo:

Solicitud:

HTTP

```
GET [Organization URI]/api/data/v9.2/accounts(00000000-0000-0000-0000-000000000001)/name/$value HTTP/1.1
Accept: application/json
OData-MaxVersion: 4.0
OData-Version: 4.0
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
Content-Type: text/plain
OData-Version: 4.0

Adventure Works (sample)
```

Usar el valor bruto no es común, a menos que esté trabajando con datos de archivos o imágenes. Más información: [Descargue un archivo en una sola solicitud utilizando la API web](#).

Recuperar valores de propiedad de navegación

Puede obtener acceso a los valores de propiedades de navegación o campos de búsqueda anexando el nombre de la propiedad de navegación a la URI que hace referencia a una entidad individual.

El siguiente ejemplo devuelve el `fullname` del `contact` primario de una `account` usando la propiedad de navegación de un solo valor `primarycontactid`:

Solicitud:

```
HTTP

GET [Organization URI]/api/data/v9.2/accounts(00000000-0000-0000-0000-
000000000001)/primarycontactid?$select=fullname HTTP/1.1
Accept: application/json
OData-MaxVersion: 4.0
OData-Version: 4.0
```

Respuesta:

```
HTTP

HTTP/1.1 200 OK
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0

{
  "@odata.context": "[Organization
URI]/api/data/v9.2/$metadata#contacts(fullname)/$entity",
  "@odata.etag": "W/\\"500128\\\"",
  "fullname": "Rene Valdes (sample)",
  "contactid": "ff390c24-9c72-e511-80d4-00155d2a68d1"
}
```

Para propiedades de navegación valoradas como colección, puede solicitar la devolución solo de referencias a las entidades relacionadas o simplemente un recuento de entidades relacionadas.

El siguiente ejemplo devolverá referencias a las tareas relacionadas con una cuenta específica agregando `/$ref` a la solicitud.

Solicitud:

HTTP

```
GET [Organization URI]/api/data/v9.2/accounts(00000000-0000-0000-0000-000000000001)/AccountTasks/$ref HTTP/1.1
Accept: application/json
OData-MaxVersion: 4.0
OData-Version: 4.0
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0

{
    "@odata.context": "[Organization
URI]/api/data/v9.2/$metadata#Collection($ref)",
    "value":
        [
            { "@odata.id": "[Organization URI]/api/data/v9.2/tasks(6b5941dd-d175-
e511-80d4-00155d2a68d1)" },
            { "@odata.id": "[Organization URI]/api/data/v9.2/tasks(fcbb60ed-d175-
e511-80d4-00155d2a68d1)" }
        ]
}
```

El siguiente ejemplo devuelve el número de tareas relacionadas con una cuenta específica mediante la propiedad de navegación valorada como colección

`Account_Tasks` con `/$count` anexado:

Solicitud:

HTTP

```
GET [Organization URI]/api/data/v9.2/accounts(00000000-0000-0000-0000-000000000001)/Account_Tasks/$count HTTP/1.1
Accept: application/json
```

```
OData-MaxVersion: 4.0  
OData-Version: 4.0
```

Respuesta:

HTTP

ÿ»¿2

ⓘ Nota

El valor devuelto incluye caracteres de marca de orden de bytes (BOM) UTF-8 (ÿ»¿), que indica que es un documento UTF-8.

Recuperar registros relacionados ampliando las propiedades de navegación

Use la opción de consulta del sistema `$expand` para controlar qué datos de entidades relacionadas se devuelven. Más información: [Unir tablas](#)

Detectar si un registro ha cambiado desde que se recuperó

Como recomendación de rendimiento solo debe solicitar los datos que necesite. Si ha recuperado anteriormente un registro de entidad, puede usar la *ETag* asociada a un registro anteriormente recuperado para realizar recuperaciones condicionales en dicho registro. Más información: [Recuperaciones condicionales](#).

Recuperar valores con formato

Solicite [valores con formato](#) para recuperaciones de registros individuales se hace la misma forma que cuando consulta conjuntos de entidades.

Consulte también

[Realizar operaciones mediante la API web](#)
[Componer solicitudes HTTP y administrar errores](#)

Nota

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta. [↗](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad ↗](#)).

Actualizar y eliminar filas de tablas usando la API web

Artículo • 18/12/2023

Las operaciones para modificar datos son una parte básica de la API web. Además de una simple actualización y eliminación de operaciones, puede realizar operaciones en columnas de una sola tabla (atributos de entidad) y componer solicitudes *upsert* que actualizarán o insertarán datos dependiendo de si existen.

Actualización básica

Las operaciones de actualización usan el verbo HTTP `PATCH`. Pase un objeto JSON que contenga las propiedades que desee actualizar a la URL que representa al registro. Se devolverá una respuesta con un estado de `204 No Content` si la actualización es correcta.

El encabezado `If-Match: *` garantiza que no crea un nuevo registro al realizar accidentalmente una operación *upsert*. Más información: [Evitar realizar operaciones de creación en upsert](#).

Importante

Al actualizar una entidad, incluya únicamente las propiedades que está modificando en el cuerpo de la solicitud. Actualizando simplemente las propiedades de una entidad que recuperó anteriormente, e incluyendo ese JSON en su solicitud, se actualizará cada propiedad aunque el valor sea el mismo. Esto puede provocar eventos del sistema que pueden desencadenar la lógica de negocios que espera que los valores hayan cambiado. Esto puede hacer que parezca que las propiedades se han actualizado en auditoría de datos cuando de hecho no han cambiado realmente.

Cuando actualice la propiedad `statecode`, es importante establecer siempre el `statuscode` deseado. `statecode` y `statuscode` tienen valores dependientes. Puede haber varios valores `statuscode` válidos para un valor `statecode` dado, pero cada columna de `statecode` tiene un único valor **DefaultStatus** configurado. Cuando actualice `statecode` sin especificar un `statuscode`, el sistema establecerá el valor de estado predeterminado. Además, cuando la auditoría está habilitada en la tabla y la columna `statuscode`, el valor modificado para la columna `statuscode` no se

capturará en los datos de auditoría a menos que se especifique en la operación de actualización.

ⓘ Nota

La definición de atributos incluye una propiedad `RequiredLevel`. Cuando esta opción se establece en `SystemRequired`, no puede establecer estos atributos a un valor nulo. Más información: [Nivel de requisito de atributo](#)

Este ejemplo actualiza un registro de cuenta existente con el valor `accountid` de 00000000-0000-0000-000000000001.

Solicitud:

HTTP

```
PATCH [Organization URI]/api/data/v9.0/accounts(00000000-0000-0000-0000-000000000001) HTTP/1.1
Content-Type: application/json
OData-MaxVersion: 4.0
OData-Version: 4.0
If-Match: *

{
    "name": "Updated Sample Account",
    "creditonhold": true,
    "address1_latitude": 47.639583,
    "description": "This is the updated description of the sample account",
    "revenue": 6000000,
    "accountcategorycode": 2
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 No Content
OData-Version: 4.0
```

ⓘ Nota

Consulte [Uso de propiedades de navegación de un solo valor](#) para obtener información sobre cómo asociar y desasociar entidades en la actualización.

Actualizar con datos devueltos

Para recuperar datos de una entidad que está actualizando puede crear la solicitud `PATCH` de forma que los datos del registro creado sean devueltos con un estado de 200 (OK). Para obtener este resultado, debe usar el encabezado de solicitud `Prefer: return=representation`.

Para controlar qué propiedades se devuelven, anexe la opción de consulta `$select` a la dirección URL del conjunto de entidades. Se ignorará la opción de consulta `$expand` si se utiliza.

Este ejemplo actualiza una entidad de cuenta y devuelve los datos solicitados en la respuesta.

Solicitud:

HTTP

```
PATCH [Organization URI]/api/data/v9.0/accounts(00000000-0000-0000-0000-000000000001)?
$select=name,creditonhold,address1_latitude,description,revenue,accountcategorycode,createdon HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
Accept: application/json
Content-Type: application/json; charset=utf-8
Prefer: return=representation
If-Match: *

{"name": "Updated Sample Account"}
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
Content-Type: application/json; odata.metadata=minimal
Preference-Applied: return=representation
OData-Version: 4.0

{
    "@odata.context": "[Organization
URI]/api/data/v9.0/$metadata#accounts/$entity",
    "@odata.etag": "W/\"536537\"",
    "accountid": "00000000-0000-0000-000000000001",
    "accountcategorycode": 1,
    "description": "This is the description of the sample account",
    "address1_latitude": 47.63958,
```

```
    "creditonhold": false,  
    "name": "Updated Sample Account",  
    "createdon": "2016-09-28T23:14:00Z",  
    "revenue": 5000000.000,  
    "_transactioncurrencyid_value": "048ddaa-6f7f-e611-80d3-00155db5e0b6"  
}
```

Actualizar múltiples registros en una sola solicitud

La forma más rápida de actualizar varios registros del mismo tipo en una sola solicitud es utilizar la acción [UpdateMultiple](#). En el momento de escribir este artículo, la acción [UpdateMultiple](#). No todas las tablas estándar admiten esta acción, pero todas las tablas elásticas sí.

Más información:

- [Mensajes de operación en masa](#)
- [Muestra: API web, Usar operaciones masivas](#)
- [Use UpdateMultiple con tablas elásticas](#)

Actualizar un solo valor de propiedad

Cuando desee actualizar únicamente un solo valor de propiedad, utilice una solicitud [PUT](#) con el nombre de propiedad anexado a la Uri de la entidad.

El siguiente ejemplo actualiza la propiedad de `name` de una fila `account` existente con el valor `accountid` de 00000000-0000-0000-000000000001.

Solicitud:

HTTP

```
PUT [Organization URI]/api/data/v9.0/accounts(00000000-0000-0000-0000-  
000000000001)/name HTTP/1.1  
Content-Type: application/json  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
  
{"value": "Updated Sample Account Name"}
```

Respuesta:

HTTP

```
HTTP/1.1 204 No Content  
OData-Version: 4.0
```

Eliminar un solo valor de propiedad

Para eliminar el valor de una sola propiedad, utilice una solicitud `DELETE` con el nombre de propiedad anexado a la Uri de la entidad.

El siguiente ejemplo elimina el valor de la propiedad `description` de una entidad de cuenta existente con el valor `accountid` de `00000000-0000-0000-0000-000000000001`.

Solicitud:

HTTP

```
DELETE [Organization URI]/api/data/v9.0/accounts(00000000-0000-0000-0000-  
000000000001)/description HTTP/1.1  
Content-Type: application/json  
OData-MaxVersion: 4.0  
OData-Version: 4.0
```

Respuesta:

HTTP

```
HTTP/1.1 204 No Content  
OData-Version: 4.0
```

ⓘ Nota

Esto no se puede usar con una propiedad de navegación de un solo valor para anular la asociación de dos entidades. Para un enfoque alternativo, consulte [Desasociar con una propiedad de navegación de un solo valor](#).

Upsert la fila de una tabla

Una operación *upsert* es similar a una actualización. Usa una solicitud `PATCH` y usa una URI para hacer referencia a un registro específico. La diferencia es que si el registro no

existe, se creará. Si ya existe, se actualiza.

Upsert es valioso al sincronizar datos entre sistemas externos. El sistema externo no puede contener una referencia a la clave principal de la tabla Dataverse, por lo que puede configurar claves alternativas para la tabla Dataverse utilizando valores del sistema externo que identifiquen de forma única el registro en ambos sistemas. Más información: [Definir claves alternativas para hacer referencia a filas](#)

Puede ver las claves alternativas definidas para una tabla en las anotaciones para el tipo de entidad en el documento \$Metadata Service. Más información: [Claves alternativas](#).

En el siguiente ejemplo, hay una tabla con el nombre `sample_thing` que tiene un clave alternativa que hace referencia a dos columnas: `sample_key1` y `sample_key2`, ambas definidas para almacenar valores enteros.

Solicitud:

HTTP

```
PATCH [Organization
URI]/api/data/v9.2/sample_things(sample_key1=1,sample_key2=1) HTTP/1.1
Accept: application/json
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Content-Type: application/json

{
    "sample_name": "1:1"
}
```

Tanto para las operaciones de creación como para las de actualización, obtiene la misma respuesta. El encabezado de respuesta `OData-EntityId` usa los valores clave, y no el identificador de clave principal GUID para el registro.

Respuesta:

HTTP

```
HTTP/1.1 204 No Content
OData-Version: 4.0
OData-EntityId: [Organization
URI]/api/data/v9.2/sample_things(sample_key1=1,sample_key2=1)
```

Debido a que la respuesta es la misma, no puede saber si la operación representó una operación `Create` o `Update`.

Si necesita saberlo, puede usar el encabezado de solicitud `Prefer: return=representation`. Con este encabezado, obtiene una respuesta `201 Created` cuando se crea un registro y una respuesta `200 OK` cuando se actualiza el registro. Esta opción agrega una operación `Retrieve`, que tiene un impacto en el rendimiento. Si usa el encabezado de solicitud `Prefer: return=representation`, asegúrese de que su `$select` incluya la cantidad mínima de datos, preferiblemente solo la columna de clave principal. Más información: [Actualizar con datos devueltos](#) y [Crear con datos devueltos](#).

Cuando utilice claves alternativas, no debe incluir los valores clave alternativa en el cuerpo de la solicitud.

- Cuando un upsert representa un `Update`, estos valores clave alternativa se ignoran. No puede actualizar los valores de clave alternativa mientras los usa para identificar el registro.
- Cuando un upsert representa a `Create`, los valores clave en la URL se establecen para el registro si no están presentes en el cuerpo. Por lo tanto, no es necesario incluirlos en el cuerpo de la solicitud.

Más información: [Uso de Upset para crear o actualizar un registro](#)

① Nota

Normalmente, al crear un nuevo registro, usted deja que el sistema asigne un valor GUID para la clave principal. Este es un método recomendado porque el sistema genera claves que están optimizadas para el índice, y esto mejora el rendimiento. Pero, si necesita crear un registro con un valor de clave principal específico, como cuando un sistema externo genera el valor GUID de clave, la operación `upsert` proporciona una manera de hacer esto.

Impedir crear o actualizar con upsert

A veces, hay situaciones donde desea realizar `upsert`, pero desea evitar una de las operaciones potenciales: crear o actualizar. Puede hacerlo mediante encabezados `If-Match` o `If-None-Match`. Para obtener más información, consulte [Limitar operaciones de upsert](#).

Eliminación básica

Una operación de eliminación es sencilla. Utilice el verbo `DELETE` con el URI de la entidad que desea borrar. Este mensaje de ejemplo elimina una entidad de cuenta con

el valor de clave principal `accountid` igual a 00000000-0000-0000-0000-000000000001.

Solicitud:

HTTP

```
DELETE [Organization URI]/api/data/v9.0/accounts(00000000-0000-0000-0000-000000000001) HTTP/1.1
Content-Type: application/json
OData-MaxVersion: 4.0
OData-Version: 4.0
```

Respuesta:

Si existe la entidad, recibirá una respuesta con estado 204 para indicar que la eliminación fue correcta. Si no se encuentra la entidad, recibirá una respuesta con estado 404.

HTTP

```
HTTP/1.1 204 No Content
OData-Version: 4.0
```

Comprobar registros duplicados

Consulte [Detectar duplicados durante la operación de actualización mediante la API web](#) para obtener más información sobre cómo buscar registros duplicados durante la operación de actualización.

Eliminar múltiples registros en una sola solicitud

La forma más rápida de eliminar varios registros del mismo tipo en una sola solicitud es utilizar la acción `DeleteMultiple`. En el momento de escribir este artículo, la acción `DeleteMultiple` es una función de vista previa. Las tablas estándar no admiten esta acción, pero todas las tablas elásticas sí.

ⓘ Nota

Para las tablas estándar, recomendamos usar la acción **BulkDelete**, que permite la eliminación asíncrona de registros que coinciden con una consulta. Más información: [Eliminar datos en masa](#)

Más información:

- [Mensajes de operación en masa](#)
- [Código de ejemplo de tablas elásticas](#)
- [Use DeleteMultiple con tablas elásticas](#)

Actualizar y eliminar documentos en particiones de almacenamiento

Si está actualizando o eliminando datos de tabla elástica almacenados en particiones, asegúrese de especificar la clave de partición cuando acceda a esos datos.

Más información: [Elegir un valor de PartitionId](#)

Consulte también

[Ejemplo de operaciones básicas de la API web \(C#\)](#)

[Ejemplo de operaciones básicas de la API web \(JavaScript del lado del cliente\)](#)

[Realizar operaciones mediante la API web](#)

[Componer solicitudes HTTP y administrar errores](#)

[Consultar datos utilizando la API web](#)

[Crear una fila de tabla usando la API web](#)

[Recuperar una fila de tabla usando la API web](#)

[Asociar y anular la asociación de filas de tabla mediante la API web](#)

[Usar funciones de la API web](#)

[Usar acciones de la API web](#)

[Ejecutar las operaciones por lotes mediante API web](#)

[Suplantar a otro usuario utilizando la API web](#)

[Realizar operaciones condicionales mediante la API web](#)

Nota

¿Puede indicarnos sus preferencias de idioma de documentación? [Realice una breve encuesta.](#) ↗ (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales
[\(declaración de privacidad ↗\)](#).

Asociar y anular la asociación de filas de tabla mediante la API web

Artículo • 25/08/2023

Puede asociar registros individuales en las filas de la tabla con otros registros usando las relaciones existentes entre las definiciones de la tabla. En OData, las relaciones se expresan como propiedades de navegación.

Puede descubrir qué propiedades de navegación existen en el documento \$Metadata Service. Consulte [Propiedades de navegación de la API web](#). Para las tablas Dataverse existentes, consulte [Web API EntityType Reference](#); para cada tipo de entidad, consulte las propiedades de navegación valoradas como colección y de un solo valor.

La siguiente tabla describe los tres tipos de relaciones entre tablas en Dataverse.

Type	Descripción	Ejemplo
Uno a varios	Un registro puede tener muchos registros asociados.	Un registro account puede tener muchos registros contact en la <i>propiedad de navegación valorada como colección</i> <code>contact_customer_accounts</code> .
Varios a uno	Muchos registros se pueden asociar a un registro. Varios a uno es una imagen reflejada de una relación Uno a varios. Solo hay una relación.	Muchos registros contact se pueden asociar a un solo registro account usando la <i>propiedad de navegación valorada como colección</i> <code>parentcustomerid_account</code> .
Varios a varios	Muchos registros se pueden asociar a muchos registros.	Cada security role (role) puede incluir referencias a la definición de un systemuser . Ambas tablas tienen una <i>propiedad de navegación valorada como colección</i> <code>systemuserroles_association</code> .

Uso de propiedades de navegación de un solo valor

Para los registros existentes en el lado *varios* de una relación Uno a varios o Varios a uno, puede asociar el registro estableciendo una referencia Uri al otro registro. La forma más fácil y común de hacer esto es agregando la anotación `@odata.bind` al nombre de la propiedad de navegación de un solo valor y, luego, estableciendo el valor como el Uri para el otro registro en una solicitud [PATCH](#).

Asociar con una propiedad de navegación de un solo valor

Por ejemplo, para asociar un registro `contact` a un `account` utilizando la propiedad de navegación de un solo valor `parentcustomerid_account`:

Solicitud:

HTTP

```
PATCH [Organization Uri]/api/data/v9.2/contacts(cf9eaaef-f718-ed11-b83e-00224837179f) HTTP/1.1
If-Match: *
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
  "parentcustomerid_account@odata.bind": "accounts(ce9eaaef-f718-ed11-b83e-00224837179f)"
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization Uri]/api/data/v9.2/contacts(cf9eaaef-f718-ed11-b83e-00224837179f)
```

Como se describe en [Asociar filas de tabla en la creación](#), los nuevos registros también se pueden asociar con registros existentes de la misma manera.

Desasociar con una propiedad de navegación de un solo valor

Si desea desasociar, simplemente puede establecer el valor en nulo.

Solicitud:

HTTP

```
PATCH [Organization Uri]/api/data/v9.2/contacts(cf9eaaef-f718-ed11-b83e-00224837179f) HTTP/1.1
```

```
If-Match: *
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
  "parentcustomerid_account@odata.bind": null
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization Uri]/api/data/v9.2/contacts(cf9eaaef-f718-
ed11-b83e-00224837179f)
```

Al desasociar de esta manera, no necesita incluir la anotación `@odata.bind`. Puede simplemente usar el nombre de la propiedad de navegación de un solo valor:

Solicitud:

HTTP

```
PATCH [Organization Uri]/api/data/v9.2/contacts(cf9eaaef-f718-ed11-b83e-
00224837179f) HTTP/1.1
If-Match: *
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
  "parentcustomerid_account": null
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization Uri]/api/data/v9.2/contacts(cf9eaaef-f718-
ed11-b83e-00224837179f)
```

Más información: [Actualización básica](#)

Otros métodos

Hay otras formas de lograr los mismos resultados descritos anteriormente con propiedades de navegación de un solo valor.

Puedes usar la siguiente solicitud `PUT` para fijar el valor de la propiedad de navegación de un solo valor `parentcustomerid_account`:

Solicitud:

HTTP

```
PUT [Organization Uri]/api/data/v9.2/contacts(cf9eaaef-f718-ed11-b83e-00224837179f)/parentcustomerid_account/$ref HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
    "@odata.id": "[Organization URI]/api/data/v9.2/accounts(ce9eaaef-f718-ed11-b83e-00224837179f)"
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
```

① Nota

Nota: Debe usar una URL absoluta al establecer el valor para `@odata.id`.

Para eliminar la referencia, también puede utilizar esta solicitud `DELETE`:

Solicitud:

HTTP

```
DELETE [Organization Uri]/api/data/v9.2/contacts(cf9eaaef-f718-ed11-b83e-00224837179f)/parentcustomerid_account/$ref HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
```

```
If-None-Match: null  
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent  
OData-Version: 4.0
```

Uso de propiedades de navegación valoradas como colección

Con OData, ambos lados de una relación Varios a varios tendrán propiedades de navegación valoradas como colección. En relaciones Uno a varios y Varios a uno, la tabla uno del lado 'Uno' tendrá una propiedad de navegación valorada como colección. No hay diferencia en cómo trabaja con cualquiera de estos tipos de relaciones mientras usa propiedades de navegación valoradas como colección. Esta sección describirá cómo trabajar con propiedades de navegación valoradas como colección con cualquier tipo de relación.

Agregar un registro a una colección

El siguiente ejemplo muestra cómo agregar un registro `contact` a la colección `account` `contact_customer_accounts` que es parte de una relación Uno a varios.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/accounts(ce9eaaef-f718-ed11-b83e-  
00224837179f)/contact_customer_accounts/$ref HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json  
  
{  
    "@odata.id": "[Organization URI]/api/data/v9.2/contacts(cf9eaaef-f718-  
ed11-b83e-00224837179f)"  
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent  
OData-Version: 4.0
```

El siguiente ejemplo muestra cómo agregar un registro **role** a la colección **systemuser** **systemuserroles_association** que es parte de una relación Varios a varios.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/systemusers(34dcba5-f718-ed11-b83e-  
00224837179f)/systemuserroles_association/$ref HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json  
  
{  
    "@odata.id": "[Organization URI]/api/data/v9.2/roles(886b280c-6396-4d56-  
a0a3-2c1b0a50ceb0)"  
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent  
OData-Version: 4.0
```

Quitar un registro de una colección

El siguiente ejemplo muestra cómo eliminar un registro **contact** a la colección **account** **contact_customer_accounts** donde el valor **contactid** del contacto es **cf9eaaef-f718-ed11-b83e-00224837179f**.

Solicitud:

HTTP

```
DELETE [Organization Uri]/api/data/v9.2/accounts(ce9eaaef-f718-ed11-b83e-  
00224837179f)/contact_customer_accounts(cf9eaaef-f718-ed11-b83e-  
00224837179f)/*ref HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0
```

```
If-None-Match: null  
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent  
OData-Version: 4.0
```

Lo siguiente también funciona:

Solicitud:

HTTP

```
DELETE [Organization Uri]/api/data/v9.2/accounts(ce9eaaef-f718-ed11-b83e-  
00224837179f)/contact_customer_accounts/$ref?$id=[Organization  
URI]/api/data/v9.2/contacts(cf9eaaef-f718-ed11-b83e-00224837179f) HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent  
OData-Version: 4.0
```

Consulte también

[Ejemplo de operaciones básicas de la API web \(C#\)](#)

[Ejemplo de operaciones básicas de la API web \(JavaScript del lado del cliente\)](#)

[Realizar operaciones mediante la API web](#)

[Componer solicitudes HTTP y administrar errores](#)

[Consultar datos utilizando la API web](#)

[Crear una fila de tabla usando la API web](#)

[Recuperar una fila de tabla usando la API web](#)

[Actualizar y eliminar filas de tablas usando la API web](#)

[Usar funciones de la API web](#)

[Usar acciones de la API web](#)

[Ejecute las operaciones por lotes mediante API web](#)

[Suplantar a otro usuario utilizando la API web](#)

[Realizar operaciones condicionales mediante la API web](#)

 **Nota**

¿Puede indicarnos sus preferencias de idioma de documentación? **Realice una breve encuesta.**  (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales (**declaración de privacidad** ).

Fusionar filas de tablas con la API web

Artículo • 24/08/2023

Cuando encuentre registros duplicados, puede combinarlos en uno usando [Merge Action](#).

ⓘ Nota

Solo se pueden combinar los siguientes tipos de entidad:

- `account`
- `contact`
- `lead`: Disponibles con [Dynamics 365 for Sales](#)
- `incident`: Disponible con [Dynamics 365 para servicio](#)

Vea [Fusionar comportamiento para incidente](#)

Acción Combinar

Combinar es una acción independiente que acepta cuatro parámetros:

Nombre	Tipo	Descripción	Opcional
<code>Target</code>	<code>crmbaseentity</code>	El destino de la operación de combinar.	No
<code>Subordinate</code>	<code>crmbaseentity</code>	El registro de entidad desde el que se combinarán datos.	No
<code>UpdateContent</code>	<code>crmbaseentity</code>	Atributos de entidad adicionales que se establecerán durante la operación de combinación.	Sí
<code>PerformParentingChecks</code>	Booleano	Indica si se debe comprobar si la información principal es diferente para los dos registros de entidad.	No

La fusión moverá todos los datos útiles del registro `Subordinate` a `Target`. Cualquier dato existente en el registro `Target` no se sobrescribirá. Después se desactiva el registro `Subordinate`. Para llevar a cabo esta operación, el autor de la llamada debe tener

privilegios y derechos de acceso a los dos registros identificados como `Target` y `Subordinate`.

Use una solicitud POST para enviar datos con el fin de fusionar registros. Este ejemplo combina dos registros de entidad de cuenta mientras se actualiza la propiedad `accountnumber` del registro que permanecerá después de la combinación.

Solicitud:

```
HTTP

POST [Organization URI]/api/data/v9.0/Merge HTTP/1.1
Content-Type: application/json; charset=utf-8
OData-MaxVersion: 4.0
OData-Version: 4.0
Accept: application/json

{
  "Target": {
    "name": "Account 1",
    "accountid": "bb8055c0-aea6-ea11-a812-000d3a55d474",
    "@odata.type": "Microsoft.Dynamics.CRM.account"
  },
  "Subordinate": {
    "name": "Account 2",
    "accountid": "c38055c0-aea6-ea11-a812-000d3a55d474",
    "@odata.type": "Microsoft.Dynamics.CRM.account"
  },
  "UpdateContent": {
    "accountnumber": "1234",
    "@odata.type": "Microsoft.Dynamics.CRM.account"
  },
  "PerformParentingChecks": false
}
```

ⓘ Importante

Dado que los tipos de propiedad `Target`, `Subordinate` y `UpdateContent` no se definen explícitamente por el parámetro, debe incluir la anotación `@odata.type` para especificar el tipo.

Respuesta:

```
HTTP

HTTP/1.1 204 No Content
OData-Version: 4.0
```

Fusionar comportamiento para incident

La tabla fusionar comportamiento para `incident` (caso) es diferente de las tablas `account`, `contact`, o `lead`.

- Los datos del parámetro `UpdateContent` no se utilizan.
- La fusión se realiza en el contexto de seguridad del usuario.

Las operaciones de fusión para otras tablas se realizan con un contexto de seguridad de usuario del sistema. Debido a que las operaciones de combinación de incidentes se realizan en el contexto de seguridad del usuario, este debe tener los privilegios de seguridad para realizar cualquiera de las acciones, como la reorganización dinámica de relación jerárquica para los registros relacionados, que realiza la operación de combinación.

Si el usuario que fusiona registros no tiene privilegios para todas las acciones incluidas en la operación de fusión, la operación de fusión no se realizará correctamente y volverá al estado original.

Consulte también

[Usar acciones de la API web](#)

[Combinar registros duplicados](#)

[MergeRequest Class](#)

[Guía de administración: fusionar datos](#)

[Dynamics 365 para servicio: fusionar casos](#)

ⓘ Nota

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta. ↗ (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales (declaración de privacidad ↗).

Usar funciones de la API web

Artículo • 24/08/2023

Las funciones son operaciones reutilizables que puede realizar utilizando la API web. Existen dos tipos de funciones en la API web:

- **Funciones:** Use una solicitud `GET` con las funciones que aparecen en [Web API Function Reference](#) para realizar operaciones que tienen efectos secundarios. Estas funciones generalmente recuperan datos, ya sea una colección o un tipo complejo. Cada función tiene un mensaje correspondiente en el servicio de la organización.
- **Funciones de consulta:** Use las funciones enumeradas en [Web API Query Function Reference](#) para evaluar propiedades y valores en la creación de una consulta. Cada función de consulta tiene un valor [ConditionOperator](#) correspondiente.

Paso de parámetros a una función

Para las funciones que requieren parámetros, se recomienda pasar los valores mediante parámetros.

Por ejemplo, cuando utiliza la función [GetTimeZoneCodeByLocalizedName](#), debe incluir los valores de los parámetros `LocalizedStandardName` y `LocaleId`. Podría usar la siguiente sintaxis en línea:

HTTP

```
GET [Organization  
URI]/api/data/v9.0/GetTimeZoneCodeByLocalizedName(LocalizedStandardName='Pac  
ific Standard Time', LocaleId=1033)
```

Sin embargo, habrá un problema cuando use los valores `DateTimeOffset` con la sintaxis en línea, como se explica en el siguiente artículo: [DateTimeOffset como parámetro de consulta #204](#).

Evite el problema `DateTimeOffset` pasando los valores como parámetros, como se muestra en el siguiente código de ejemplo:

HTTP

```
GET [Organization  
URI]/api/data/v9.0/GetTimeZoneCodeByLocalizedName(LocalizedStandardName=@p1,  
LocaleId=@p2)?@p1='Pacific Standard Time'&@p2=1033
```

Cuando un valor de parámetro se usa varias veces, los [alias de parámetro](#) permiten reusarlo para reducir la longitud total de la URL.

Pasar referencia de registro a una función

Determinadas funciones requieren el pase de una referencia a un registro existente. Por ejemplo, las siguientes funciones tienen un parámetro que requiere un tipo de entidad `crmbaseentity`:

Funciones		
CalculateRollupField	IncrementKnowledgeArticleViewCount	InitializeFrom
IsValidStateTransition	RetrieveDuplicates	RetrieveLocLabels
RetrievePrincipalAccess	RetrieveRecordWall	ValidateRecurrenceRule

Al pasar una referencia a un registro existente, use la anotación `@odata.id` en el URI para el registro. Por ejemplo, si está utilizando la función [RetrievePrincipalAccess](#), puede usar el siguiente Uri para especificar la recuperación del acceso a un registro de contacto específico:

HTTP

```
GET [Organization URI]/api/data/v9.0/systemusers(af9b3cf6-f654-4cd9-97a6-cf9526662797)/Microsoft.Dynamics.CRM.RetrievePrincipalAccess(Target=@tid)?@tid={'@odata.id':'contacts(9f3162f6-804a-e611-80d1-00155d4333fa)'}
```

La anotación `@odata.id` puede ser el URI completo o un URI relativo.

Funciones enlazadas y sin enlazar

Solo se pueden vincular las funciones que se encuentran en [Web API Function Reference](#) o creadas como una [API personalizada](#). Las funciones de consulta nunca están enlazadas.

Funciones enlazadas

En el [Documento \\$metadata de CSDL](#), cuando un elemento `Function` representa una función enlazada, tiene un atributo `IsBound` con el valor `true`. El primer elemento `Parameter` definido en la función representa la entidad a la que está enlazada la función.

Cuando el atributo `Type` del parámetro es una colección, la función está enlazada a una colección de entidades.

El siguiente ejemplo es la definición de la función `RetrieveUserPrivileges` y del tipo complejo `RetrieveUserPrivilegesResponse` en CSDL.

XML

```
<ComplexType Name="RetrieveUserPrivilegesResponse">
    <Property Name="RolePrivileges" Type="Collection(mscrm.RolePrivilege)" />
</ComplexType>
<Function Name="RetrieveUserPrivileges" IsBound="true">
    <Parameter Name="entity" Type="mscrm.systemuser" Nullable="false" />
    <ReturnType Type="mscrm.RetrieveUserPrivilegesResponse" Nullable="false" />
</Function>
```

Esta función vinculada a la entidad es equivalente a la clase `RetrieveUserPrivilegesRequest` utilizada por el SDK para .NET. En la API web, esta función está vinculada al tipo de entidad `systemuser` que representa la propiedad `RetrieveUserPrivilegesRequest.UserId`. En lugar de devolver una instancia de la clase `RetrieveUserPrivilegesResponse`, esta función devuelve un tipo complejo `RetrieveUserPrivilegesResponse`. Cuando una función devuelve un tipo complejo, su definición suele aparecer directamente encima de la definición de la función en el CSDL.

Para invocar a una función sin enlazar, anexe el nombre completo de la función a la URL e incluya cualquier parámetro con nombre entre paréntesis tras el nombre de la función. El nombre completo de la función incluye el espacio de nombres `Microsoft.Dynamics.CRM`. Las funciones que no están enlazadas no deben usar el nombre completo.

ⓘ Importante

Invoque una función enlazada mediante un URI para establecer el primer valor de parámetro. No puede establecerlo como un valor de parámetro con nombre.

El siguiente ejemplo usa la función `RetrieveUserPrivileges`, que está ligada a la tabla `systemuser`.

Solicitud:

HTTP

```
GET [Organization URI]/api/data/v9.2/systemusers(da455fec-68b7-ec11-9840-000d3a13d713)/Microsoft.Dynamics.CRM.RetrieveUserPrivileges HTTP/1.1
Accept: application/json
OData-MaxVersion: 4.0
OData-Version: 4.0
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0

{
  "@odata.context": "[Organization
URI]/api/data/v8.2/$metadata#Microsoft.Dynamics.CRM.RetrieveUserPrivilegesRe
sponse",
  "RolePrivileges": [
    {
      "Depth": "Global",
      "PrivilegeId": "20db4bf7-60c4-4eb9-ab95-0949766fef1a",
      "BusinessUnitId": "dfe37870-c8ac-ec11-9841-0022482088be",
      "PrivilegeName": "prvCreateflowsession"
    },
    {
      "Depth": "Global",
      "PrivilegeId": "d8db8e4c-5b76-48eb-b5ec-171b8c661917",
      "BusinessUnitId": "dfe37870-c8ac-ec11-9841-0022482088be",
      "PrivilegeName": "prvWriteworkflowbinary"
    },
    ... <full list of privileges removed for brevity>
    {
      "Depth": "Global",
      "PrivilegeId": "b234db9f-27a2-4d12-8b51-fc34fbef9d87",
      "BusinessUnitId": "dfe37870-c8ac-ec11-9841-0022482088be",
      "PrivilegeName": "prvWriteflowsession"
    }
  ]
}
```

Funciones sin enlazar

La función [WhoAmI](#) no está vinculada a una entidad. Se define en CSDL sin un atributo `IsBound`.

XML

```
<ComplexType Name="WhoAmIResponse">
  <Property Name="BusinessUnitId" Type="Edm.Guid" Nullable="false" />
  <Property Name="UserId" Type="Edm.Guid" Nullable="false" />
  <Property Name="OrganizationId" Type="Edm.Guid" Nullable="false" />
</ComplexType>
<Function Name="WhoAmI">
  <ReturnType Type="mscrm.WhoAmIResponse" Nullable="false" />
</Function>
```

Esta función corresponde a la clase [WhoAmIRequest](#) y devuelve un tipo complejo [WhoAmIResponse](#) que corresponde a la clase [WhoAmIResponse](#) utilizada por el SDK para .NET. Esta función no tiene ningún parámetro.

Al invocar una función sin enlazar, use únicamente el nombre de función, como se muestra en el siguiente ejemplo:

Solicitud:

HTTP

```
GET [Organization URI]/api/data/v9.0/WhoAmI() HTTP/1.1
Accept: application/json
OData-MaxVersion: 4.0
OData-Version: 4.0
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0
{
  "@odata.context": "[Organization
URI]/api/data/v9.0/$metadata#Microsoft.Dynamics.CRM.WhoAmIResponse",
  "BusinessUnitId": "ded5a64f-f06d-e511-80d0-00155db07cb1",
  "UserId": "d96e9f55-f06d-e511-80d0-00155db07cb1",
  "OrganizationId": "4faf1f34-f06d-e511-80d0-00155db07cb1"
}
```

Crear una consulta con funciones

Hay dos formas de utilizar las funciones para controlar los datos devueltos con consultas. Algunas funciones permiten el control de las columnas o condiciones que

devuelven y se pueden usar funciones de consulta para evaluar condiciones en una consulta.

Funciones que admiten composición

Algunas funciones que figuran en [Web API Function Reference](#) devuelven una colección de entidades. Un subconjunto de estas funciones *admite composición*, lo que significa que puede incluir una opción de consulta del sistema `$select` o `$filter` para controlar qué columnas se devuelven en los resultados. Estas funciones tienen un atributo `IsComposable` en CSDL. Cada una de estas funciones tiene un mensaje de acompañamiento manual en el SDK que acepta un parámetro de tipo `ColumnSet` o `QueryBase`. Las opciones de consulta del sistema OData ofrecen la misma funcionalidad, por lo que estas funciones no tienen los mismos parámetros que sus mensajes de acompañamiento en el SDK. La siguiente tabla muestra una lista de esas funciones que admiten composición de esta versión.

Funciones		
RetrieveAllChildUsersSystemUser	RetrieveBusinessHierarchyBusinessUnit	RetrieveUnpublishedMultiple
SearchByBodyKbArticle	SearchByKeywordsKbArticle	SearchByTitleKbArticle

Funciones de consulta

Las funciones enumeradas en [Web API Query Function Reference](#) se proporcionan para ser utilizadas para crear una consulta. Puede usarlas de forma similar a las [Funciones de consulta OData](#), aunque hay algunas diferencias importantes. Debe usar el nombre completo de la función e incluir los nombres de los parámetros.

El siguiente ejemplo usa la función de consulta `LastXHours` para devolver todas las entidades de cuenta modificadas en las últimas 12 horas.

HTTP

```
GET [Organization URI]/api/data/v9.0/accounts?  
$select=name,accountnumber&$filter=Microsoft.Dynamics.CRM.LastXHours(Property  
yName=@p1,PropertyValue=@p2)&@p1='modifiedon'&@p2=12
```

Limitaciones de las funciones de consulta

Una de las limitaciones de funciones de consulta es que no puede usar el operador `not` para negar funciones de consulta.

Por ejemplo, la consulta siguiente, que usa `EqualUserId`, produce el error: `Not operator along with the Custom Named Condition operators is not allowed.`

HTTP

```
GET [Organization URI]/api/data/v9.1/systemusers?
$select=fullname,systemuserid&$filter=not
Microsoft.Dynamics.CRM.EqualUserId(PropertyName=@p1)&@p1='systemuserid'
```

Varias funciones de consulta tienen una función de consulta negada complementaria. Por ejemplo, `NotEqualUserId` niega `EqualUserId`, por lo que la siguiente consulta devuelve los resultados esperados:

HTTP

```
GET [Organization URI]/api/data/v9.1/systemusers?
$select=fullname,systemuserid&$filter=Microsoft.Dynamics.CRM.NotEqualUserId(
PropertyName=@p1)&@p1='systemuserid'
```

Otras funciones de consulta se pueden negar de distintas formas. Por ejemplo, en lugar de intentar negar una función de consulta `Last7Days` como esta (que produce un error con el mismo mensaje anteriormente indicado):

HTTP

```
GET [Organization URI]/api/data/v9.1/accounts?$select=name&$filter=not
Microsoft.Dynamics.CRM.Last7Days(PropertyName=@p1)&@p1='createdon'
```

Utilizar una función de consulta `OlderThanXDays` como esta:

HTTP

```
GET [Organization URI]/api/data/v9.1/accounts?
$select=name&$filter=Microsoft.Dynamics.CRM.OlderThanXDays(PropertyName=@p1,
PropertyValue=@p2)&@p1='createdon'&@p2=7
```

Consulte también

[Ejemplo de funciones y acciones de la API web \(C#\)](#)

[Ejemplo de funciones y acciones de la API web \(JavaScript del lado del cliente\)](#)

[Realizar operaciones mediante la API web](#)

[Componer solicitudes HTTP y administrar errores](#)

[Consultar datos utilizando la API web](#)

[Crear una fila de tabla usando la API web](#)
[Recuperar una fila de tabla usando la API web](#)
[Actualizar y eliminar filas de tablas usando la API web](#)
[Asociar y anular la asociación de filas de tabla mediante la API web](#)
[Usar acciones de la API web](#)
[Ejecutar las operaciones por lotes mediante API web](#)
[Suplantar a otro usuario utilizando la API web](#)
[Realizar operaciones condicionales mediante la API web](#)

 **Nota**

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta. [↗](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad ↗](#)).

Usar acciones de la API web

Artículo • 29/09/2023

Las acciones y funciones representan operaciones reutilizables que puede realizar mediante la API web. Use una solicitud `POST` con las acciones que aparecen en [Web API Action Reference](#) para realizar operaciones que tienen efectos secundarios. También puede definir acciones personalizadas. Más información: [Crear sus propios mensajes](#).

Las acciones se definen en el [Documento de \\$metadatos CSDL](#). Consulte [Acciones API web](#) para obtener más información.

Acciones sin enlazar

El XML siguiente es la definición de la acción `Merge` representada en el documento de servicio `$metadata`.

XML

```
<Action Name="Merge">
  <Parameter Name="Target"
    Type="mscrm.crmbaseentity"
    Nullable="false" />
  <Parameter Name="Subordinate"
    Type="mscrm.crmbaseentity"
    Nullable="false" />
  <Parameter Name="UpdateContent"
    Type="mscrm.crmbaseentity" />
  <Parameter Name="PerformParentingChecks"
    Type="Edm.Boolean"
    Nullable="false" />
</Action>
```

La acción `Merge` corresponde a `MergeRequest` utilizando el SDK para .NET. Utilice esta acción para fusionar un par de registros duplicados. Esta acción no incluye un valor de devolución. Si se realiza correctamente, la operación se habrá completado.

El siguiente ejemplo es la solicitud HTTP y la respuesta para llamar a la acción `Merge` para dos registros de cuenta.

Solicitud:

HTTP

```
POST [Organization URI]/api/data/v9.2/Merge HTTP/1.1
Accept: application/json
```

```
Content-Type: application/json; charset=utf-8
OData-MaxVersion: 4.0
OData-Version: 4.0

{
  "Target": {
    "@odata.type": "Microsoft.Dynamics.CRM.account",
    "accountid": "cc1e2c4a-e577-ec11-8d21-000d3a554dc"
  },
  "Subordinate": {
    "@odata.type": "Microsoft.Dynamics.CRM.account",
    "accountid": "e408fa45-3a70-ec11-8943-00224823561e"
  },
  "PerformParentingChecks": false
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 No Content
OData-Version: 4.0
```

Más información: [Fusionar filas de tabla usando la API web](#).

Acciones enlazadas

Hay dos formas de vincular una acción. La forma más común es que la acción esté vinculada por una entidad. Con menos frecuencia, también se puede vincular a una colección de entidades.

En el [Documento de \\$metadatos CSDL](#), cuando un elemento `Action` representa una acción enlazada, tiene un atributo `IsBound` con el valor `true`. El primer elemento `Parameter` definido en la acción representa la entidad a la que está enlazada la operación. Cuando el atributo `Type` del parámetro es una colección, la operación está enlazada a una colección de entidades.

Cuando invoca una función enlazada, debe incluir el nombre completo de la función incluido el espacio de nombres `Microsoft.Dynamics.CRM`. Si no incluye el nombre completo, recibirá el siguiente error: `Status Code:400 Request message has unresolved parameters.`

Acciones vinculadas a una tabla

Como ejemplo de una acción ligada a una entidad, a continuación se define la acción [AddToQueue](#) representada en el CSDL:

```
XML

<ComplexType Name="AddToQueueResponse">
    <Property Name="QueueItemId"
        Type="Edm.Guid"
        Nullable="false" />
</ComplexType>
<Action Name="AddToQueue">
    IsBound="true">
    <Parameter Name="entity"
        Type="mscrm.queue"
        Nullable="false" />
    <Parameter Name="Target"
        Type="mscrm.crmbaseentity"
        Nullable="false" />
    <Parameter Name="SourceQueue"
        Type="mscrm.queue" />
    <Parameter Name="QueueItemProperties"
        Type="mscrm.queueitem" />
    <ReturnType Type="mscrm.AddToQueueResponse"
        Nullable="false" />
</Action>
```

Esta acción vinculada a la entidad es equivalente a la [AddToQueueRequest](#) utilizada por SDK para .NET. En la API web, esta acción está vinculada al tipo de entidad [queue](#) que representa la propiedad [AddToQueueRequest.DestinationQueueId](#). Esta acción acepta varios parámetros adicionales y devuelve un tipo complejo [AddToQueueResponse](#) que se corresponde a la [AddToQueueResponse](#) que devuelve SDK para .NET. Cuando una acción devuelve un tipo complejo, la definición del tipo complejo aparecerá directamente sobre la acción en el CSDL.

Una acción vinculada a una entidad debe invocarse mediante un URI para establecer el primer valor de parámetro. No puede establecerlo como un valor de parámetro con nombre.

El siguiente ejemplo muestra el uso de la acción [AddToQueue](#) para agregar una carta a una cola. Dado que el tipo de tipo de parámetro [Target](#) no es específico ([mscrm.crmbaseentity](#)), debe declarar explícitamente el tipo de objeto usando el valor de propiedad [@odata.type](#) del nombre completo de la entidad, incluido el espacio de nombres de [Microsoft.Dynamics.CRM](#). En este caso, [Microsoft.Dynamics.CRM.letter](#). Más información:[Especificar tipo de parámetro de la entidad](#)

Solicitud:

HTTP

```
POST [Organization URI]/api/data/v9.0/queues(56ae8258-4878-e511-80d4-00155d2a68d1)/Microsoft.Dynamics.CRM.AddToQueue HTTP/1.1
Accept: application/json
Content-Type: application/json; charset=utf-8
OData-MaxVersion: 4.0
OData-Version: 4.0

{
  "Target": {
    "activityid": "59ae8258-4878-e511-80d4-00155d2a68d1",
    "@odata.type": "Microsoft.Dynamics.CRM.letter"
  }
}
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0

{
  "@odata.context": "[Organization
URI]/api/data/v9.0/$metadata#Microsoft.Dynamics.CRM.AddToQueueResponse",
  "QueueItemId": "5aae8258-4878-e511-80d4-00155d2a68d1"
}
```

Acciones vinculadas a una colección de tabla

Es menos común encontrar acciones vinculadas a una colección de entidades. Los siguientes son algunos que puede encontrar:

[CreateException](#)

[DeliverIncomingEmail](#)

[ExportTranslation](#)

[ValidateSavedQuery](#)

[FulfillSalesOrder](#) en [Dynamics 365 for Sales](#)

[CreateMultiple](#)

UpdateMultiple

Como ejemplo de una acción ligada a una colección de entidades, a continuación se define la acción [ExportTranslation](#) representada en \$metadata de CSDL:

XML

```
<ComplexType Name="ExportTranslationResponse">
  <Property Name="ExportTranslationFile"
    Type="Edm.Binary" />
</ComplexType>
<Action Name="ExportTranslation"
  IsBound="true">
  <Parameter Name="entityset"
    Type="Collection(mscrm.solution)"
    Nullable="false" />
  <Parameter Name="SolutionName"
    Type="Edm.String"
    Nullable="false"
    Unicode="false" />
  <ReturnType Type="mscrm.ExportTranslationResponse"
    Nullable="false" />
</Action>
```

Esta acción vinculada a la colección de entidades es equivalente a la [ExportTranslationRequest](#) utilizada por el SDK para .NET. En la API web, esta acción está vinculada al tipo de entidad [solution](#). Pero en lugar de pasar un valor a la solicitud, el enlace de la colección de entidades simplemente aplica la restricción de que el URI de la solicitud debe incluir la ruta al conjunto de entidades especificado.

El siguiente ejemplo muestra el uso de la acción [ExportTranslation](#) que exporta un archivo binario que contiene datos sobre valores de cadena localizables que se pueden actualizar para modificar o agregar valores localizables. Tenga en cuenta cómo la acción vinculada a la colección de entidades tras el nombre del conjunto de entidades para la entidad de la solución: `solutions` .

Solicitud:

HTTP

```
POST [Organization
URI]/api/data/v9.1/solutions/Microsoft.Dynamics.CRM.ExportTranslation
HTTP/1.1
Accept: application/json
Content-Type: application/json
OData-MaxVersion: 4.0
OData-Version: 4.0
```

```
{  
    "SolutionName": "MySolution"  
}
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK  
Content-Type: application/json; odata.metadata=minimal  
OData-Version: 4.0  
  
{  
    "@odata.context": "[Organization  
URI]/api/data/v9.1/$metadata#Microsoft.Dynamics.CRM.ExportTranslationRespons  
e",  
    "ExportTranslationFile": "[Binary data Removed for brevity]"  
}
```

Utilizar una acción personalizada

Una acción personalizada puede ser una API personalizada o una acción de proceso personalizada. De cualquier forma que se cree, habrá una operación correspondiente que puede utilizar. Con la API personalizada, la operación puede ser una función. Más información: [Crear sus propios mensajes](#)

El siguiente ejemplo es para una acción de proceso personalizada.

Ejemplo de acción personalizada: Agregar una nota a un contacto

Digamos que desea crear una acción personalizada que agregue una nueva nota a un contacto específico. Puede crear una acción personalizada enlazada a la entidad de contacto con las siguientes propiedades.

UI Label	valor
Nombre del proceso	AddNoteToContact
Nombre único	new_AddNoteToContact
Entidad	Contacto
Categoría	Acción

Argumentos de procesos

Nombre	Tipo	Requerido	Dirección
NoteTitle	Cadena	Requerido	Entrada
NoteText	Cadena	Requerido	Entrada
NoteReference	EntityReference	Requerido	Salida

Pasos

Nombre	Tipo de paso	Descripción
Crear la nota	Crear registro	Title = {NoteTitle(Arguments)} Note Body = {NoteText(Arguments)} referente = {Contacto{Contact}}
Devuelve una referencia a la nota	Asignar valor	NoteReference Value = {Note(Create the note (Note))}

Tras publicar y activar la acción personalizada, cuando descargue el CSDL encontrará esta nueva acción especificada.

XML

```
<Action Name="new_AddNoteToContact"
    IsBound="true">
    <Parameter Name="entity"
        Type="mscrm.contact"
        Nullable="false" />
    <Parameter Name="NoteTitle"
        Type="Edm.String"
        Nullable="false"
        Unicode="false" />
    <Parameter Name="NoteText"
        Type="Edm.String"
        Nullable="false"
        Unicode="false" />
    <ReturnType Type="mscrm.annotation"
        Nullable="false" />
</Action>
```

La solicitud y respuesta HTTP siguientes muestran cómo llamar a la acción personalizada y la respuesta que devuelve si es correcta.

Solicitud:

HTTP

```
POST [Organization URI]/api/data/v9.2/contacts(94d8c461-a27a-e511-80d2-00155d2a68d2)/Microsoft.Dynamics.CRM.new_AddNoteToContact HTTP/1.1
Accept: application/json
Content-Type: application/json; charset=utf-8
OData-MaxVersion: 4.0
OData-Version: 4.0

{
  "NoteTitle": "New Note Title",
  "NoteText": "This is the text of the note"
}
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0

{
  "@odata.context": "[Organization
URI]/api/data/v9.2/$metadata#annotations/$entity",
  "annotationid": "9ad8c461-a27a-e511-80d2-00155d2a68d2"
}
```

Permite especificar el parámetro de tipo de tabla

Cuando una acción requiere una entidad como parámetro y el tipo de entidad es ambiguo, debe usar la propiedad `@odata.type` para especificar el tipo de entidad. El valor de esta propiedad es el nombre completo de la entidad, que sigue este patrón:

`Microsoft.Dynamics.CRM. +<entity logical name>.`

Como se muestra en la sección [Acciones enlazadas](#) anterior, el parámetro `Target` para la acción [AddToQueue](#) es una actividad. Pero como todas las actividades heredan del tipo de entidad [activitypointer](#), debe incluir la siguiente propiedad en el JSON de la entidad para especificar que el tipo de entidad es una letra: `"@odata.type":`

`"Microsoft.Dynamics.CRM.letter".`

Otros dos ejemplos son las acciones `AddMembersTeam` y `RemoveMembersTeam` porque el parámetro `Members` es una colección de tipos de entidad `systemuser`, que hereda su clave primaria `ownerid` del tipo de entidad `principal`. Si pasa el siguiente JSON para representar un único `systemuser` en la colección, está claro que la entidad es un `systemuser` y no un tipo de entidad `team`, que también hereda del tipo de entidad `principal`.

JSON

```
{  
  "Members": [{  
    "@odata.type": "Microsoft.Dynamics.CRM.systemuser",  
    "ownerid": "5dbf5efc-4507-e611-80de-5065f38a7b01"  
  }]  
}
```

Si no especifica el tipo de entidad en esta situación, puede obtener el siguiente error:

`"EdmEntityObject passed should have the key property value set."`

Consulte también

[Acciones de la API web](#)

[Ejemplo de funciones y acciones de la API web \(C#\)](#)

[Ejemplo de funciones y acciones de la API web \(JavaScript del lado del cliente\)](#)

[Realizar operaciones mediante la API web](#)

[Componer solicitudes HTTP y administrar errores](#)

[Consultar datos utilizando la API web](#)

[Crear una fila de tabla usando la API web](#)

[Recuperar una fila de tabla usando la API web](#)

[Actualizar y eliminar filas de tablas usando la API web](#)

[Asociar y anular la asociación de filas de tabla mediante la API web](#)

[Usar funciones de la API web](#)

[Ejecutar las operaciones por lotes mediante API web](#)

[Suplantar a otro usuario utilizando la API web](#)

[Realizar operaciones condicionales mediante la API web](#)

ⓘ Nota

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta. ↗ (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales
[\(declaración de privacidad ↗\)](#).

Ejecute las operaciones por lotes mediante API web

Artículo • 18/10/2023

Puede agrupar varias operaciones en una sola solicitud HTTP con una operación por lotes. Estas operaciones se realizan secuencialmente en el orden en que se especifican. El orden de las respuestas coincide con el orden de las solicitudes en la operación por lotes.

El formato para enviar solicitudes `$batch` se define en esta sección de la especificación de OData: [11.7 Solicitudes por lotes](#). El contenido de este tema resume los requisitos de especificación y proporciona a Dataverse ejemplos e información específica.

Cuándo usar solicitudes por lotes

Las solicitudes por lotes proporcionan dos capacidades que se pueden usar juntas:

- Puede enviar solicitudes para múltiples operaciones con una sola solicitud HTTP.
 - La solicitudes por lotes pueden contener hasta 1000 solicitudes individuales y no pueden contener otras solicitudes por lotes.
 - Las solicitudes de API web `$batch` son equivalentes al mensaje `ExecuteMultiple` disponible en el SDK para .NET. Más información: [Ejemplo: ejecutar varias solicitudes usando SDK para .NET](#).
- Puede agrupar solicitudes de operaciones para que se incluyan como una sola transacción mediante [Conjuntos de cambios](#).
 - Es posible que desee crear, actualizar o eliminar un conjunto de registros relacionados de una manera que garantice que todas las operaciones se realicen correctamente o fallen como grupo.
 - Las solicitudes de API web `$batch` que usan conjuntos de cambios son equivalentes al mensaje `ExecuteTransaction` disponible en el SDK para .NET. Más información: [Ejecutar mensajes en una sola transacción de la base de datos](#)

ⓘ Nota

Recuerde que las entidades asociadas se pueden crear en una sola operación más fácilmente que usando una solicitud por lotes. Más información: [Crear filas de tabla relacionadas en una sola operación](#)

Las solicitudes por lotes también se utilizan a veces para enviar solicitudes `GET` en las que la longitud de la URL puede exceder la [longitud máxima permitida de la URL](#). La gente utiliza solicitudes por lotes porque la URL de la solicitud se incluye en el cuerpo del mensaje, donde se permite una URL de hasta 64 KB (65 536 caracteres). El envío de consultas complejas mediante FetchXML puede generar direcciones URL largas. Más información: [Uso de FetchXML con una solicitud en lote](#).

En comparación con otras operaciones que se pueden realizar con la API web, las solicitudes por lotes son más difíciles de redactar. Los cuerpos de solicitud y respuesta sin procesar son esencialmente un documento de texto que debe cumplir requisitos específicos. Para acceder a los datos en una respuesta, necesita analizar el texto de la respuesta o localizar una biblioteca de código auxiliar para tener acceso a los datos de la respuesta. Consulte [Métodos auxiliares de .NET](#).

Solicitudes por lotes

Use una solicitud `POST` para enviar una operación por lotes que contenga varias solicitudes.

La solicitud `POST` que contiene el lote debe tener un encabezado [Content-Type](#) con un valor establecido como `multipart/mixed` con un conjunto de `boundary` para incluir el identificador del lote utilizando este patrón:

```
POST [Organization Uri]/api/data/v9.2/$batch HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
Content-Type: multipart/mixed; boundary="batch_<unique identifier>"
```

El identificador único no necesita ser un GUID, pero debe ser único.

Cada elemento dentro del lote debe ir precedido del identificador de lote con un encabezado `Content-Type` y [Content-Transfer-Encoding](#) como el siguiente:

```
--batch_<unique identifier>
Content-Type: application/http
Content-Transfer-Encoding: binary
```

ⓘ Importante

Solo los elementos de carga útil con un identificador de lote que coincide con el identificador de lote enviado en el se ejecutará el encabezado `Content-Type`. Si ningún elemento de carga útil utiliza el identificador de lote `Content-Type`, la solicitud por lotes se realizará correctamente sin ejecutar ningún elemento de carga útil.

El final de la solicitud en lote debe contener un indicador de finalización como el siguiente:

```
--batch_<unique identifier>--
```

ⓘ Nota

El protocolo HTTP requiere que todos los finales de línea en las cargas útiles de solicitud \$batch sean CRLF. Otros finales de línea pueden provocar errores de deserialización. Por ejemplo: `System.ArgumentException: Stream was not readable..` Si no puede utilizar CRLF, puede agregar dos finales de línea que no sean CRLF al final de la carga útil de la solicitud para resolver la mayoría de los errores de deserialización.

El siguiente ejemplo es una solicitud por lotes sin conjuntos de cambios. Este ejemplo:

- Crea tres registros de tareas asociados a una cuenta con `accountid` igual a `00000000-0000-0000-0000-000000000001`.
- Recupera los registros de tareas asociados con la cuenta.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/$batch HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
Content-Type: multipart/mixed; boundary=batch_80dd1615-2a10-428a-bb6f-
0e559792721f

--batch_80dd1615-2a10-428a-bb6f-0e559792721f
```

```

Content-Type: application/http
Content-Transfer-Encoding: binary

POST /api/data/v9.2/tasks HTTP/1.1
Content-Type: application/json; type=entry

{
  "subject": "Task 1 in batch",
  "regardingobjectid_account_task@odata.bind": "accounts(00000000-0000-0000-
0000-000000000001)"
}
--batch_80dd1615-2a10-428a-bb6f-0e559792721f
Content-Type: application/http
Content-Transfer-Encoding: binary

POST /api/data/v9.2/tasks HTTP/1.1
Content-Type: application/json; type=entry

{
  "subject": "Task 2 in batch",
  "regardingobjectid_account_task@odata.bind": "accounts(00000000-0000-0000-
0000-000000000001)"
}
--batch_80dd1615-2a10-428a-bb6f-0e559792721f
Content-Type: application/http
Content-Transfer-Encoding: binary

POST /api/data/v9.2/tasks HTTP/1.1
Content-Type: application/json; type=entry

{
  "subject": "Task 3 in batch",
  "regardingobjectid_account_task@odata.bind": "accounts(00000000-0000-0000-
0000-000000000001)"
}
--batch_80dd1615-2a10-428a-bb6f-0e559792721f
Content-Type: application/http
Content-Transfer-Encoding: binary

GET /api/data/v9.2/accounts(00000000-0000-0000-0000-
000000000001)/Account_Tasks?$select=subject HTTP/1.1

--batch_80dd1615-2a10-428a-bb6f-0e559792721f--

```

Respuestas en lote

Cuando tiene éxito, la respuesta por lotes devuelve el estado HTTP `200 OK` y cada elemento de la respuesta está separado por un valor de identificador único `Guid` que no es el mismo que el valor de la solicitud por lotes.

HTTP

```
--batchresponse_<unique identifier>
Content-Type: application/http
Content-Transfer-Encoding: binary
```

El final de la respuesta del lote contiene un indicador de finalización como el siguiente ejemplo:

HTTP

```
--batchresponse_<unique identifier>--
```

El siguiente ejemplo es la respuesta al ejemplo de solicitud por lotes anterior.

Respuesta:

HTTP

HTTP/1.1 200 OK

OData-Version: 4.0

```
--batchresponse_01346794-f2e2-4d45-8cc2-f97e09fe8916
```

Content-Type: application/http

Content-Transfer-Encoding: binary

HTTP/1.1 204 No Content

OData-Version: 4.0

Location: [Organization Uri]/api/data/v9.2/tasks(d31ba648-c592-ed11-aad1-000d3a993550)

OData-EntityId: [Organization Uri]/api/data/v9.2/tasks(d31ba648-c592-ed11-aad1-000d3a993550)

```
--batchresponse_01346794-f2e2-4d45-8cc2-f97e09fe8916
```

Content-Type: application/http

Content-Transfer-Encoding: binary

HTTP/1.1 204 No Content

OData-Version: 4.0

Location: [Organization Uri]/api/data/v9.2/tasks(d41ba648-c592-ed11-aad1-000d3a993550)

OData-EntityId: [Organization Uri]/api/data/v9.2/tasks(d41ba648-c592-ed11-aad1-000d3a993550)

```
--batchresponse_01346794-f2e2-4d45-8cc2-f97e09fe8916
```

Content-Type: application/http

Content-Transfer-Encoding: binary

HTTP/1.1 204 No Content

```

OData-Version: 4.0
Location: [Organization Uri]/api/data/v9.2/tasks(d51ba648-c592-ed11-aad1-
000d3a993550)
OData-EntityId: [Organization Uri]/api/data/v9.2/tasks(d51ba648-c592-ed11-
aad1-000d3a993550)

--batchresponse_01346794-f2e2-4d45-8cc2-f97e09fe8916
Content-Type: application/http
Content-Transfer-Encoding: binary

HTTP/1.1 200 OK
Content-Type: application/json; odata.metadata=minimal; odata.streaming=true
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#tasks(subject)",
  "value": [
    {
      "@odata.etag": "W/\"77180907\"",
      "subject": "Task 1 in batch",
      "activityid": "d31ba648-c592-ed11-aad1-000d3a993550"
    },
    {
      "@odata.etag": "W/\"77180910\"",
      "subject": "Task 2 in batch",
      "activityid": "d41ba648-c592-ed11-aad1-000d3a993550"
    },
    {
      "@odata.etag": "W/\"77180913\"",
      "subject": "Task 3 in batch",
      "activityid": "d51ba648-c592-ed11-aad1-000d3a993550"
    }
  ]
}
--batchresponse_01346794-f2e2-4d45-8cc2-f97e09fe8916--

```

Conjuntos de cambios

Además de solicitudes individuales, una solicitud en lote puede incluir conjuntos de cambios. Cuando varias operaciones están contenidas en un conjunto de cambios, todas las operaciones se consideran *atómicas*. Una operación atómica significa que si alguna de las operaciones falla, todas las operaciones completadas se revierten.

Nota

Las solicitudes `GET` no están permitidas dentro de los conjuntos de cambios. Una operación `GET` no debe cambiar los datos, por lo tanto, no pertenecen a un conjunto de cambios.

Como una solicitud en lote, los conjuntos de cambios deben tener un encabezado `Content-Type` con un valor establecido como `multipart/mixed` con un conjunto `boundary` para incluir el identificador del conjunto de cambios utilizando este patrón:

```
Content-Type: multipart/mixed; boundary="changeset_<unique identifier>"
```

El identificador único no necesita ser un GUID, pero debe ser único. Cada elemento dentro del conjunto de cambios debe ir precedido del identificador del conjunto de cambios con un encabezado `Content-Type` y `Content-Transfer-Encoding` como el siguiente ejemplo:

```
--changeset_<unique identifier>
Content-Type: application/http
Content-Transfer-Encoding: binary
```

Los conjuntos de cambios también pueden incluir un encabezado `Content-ID` con un valor único. Este valor, cuando va precedido de `$`, representa una variable que contiene la Uri para cualquier entidad creada en esa operación. Por ejemplo, cuando establece el valor de `1`, puede hacer referencia a esa entidad usando `$1` más adelante en el conjunto de cambios. Más información: [URI de referencia en una operación](#)

El final del conjunto de cambios debe contener un indicador de finalización como el siguiente ejemplo:

```
--changeset_<unique identifier>--
```

El siguiente ejemplo muestra el uso de un conjunto de cambios para:

- Agrupar la creación de tres tareas asociadas a una cuenta con un valor `accountid` de `00000000-0000-0000-0000-000000000001`.
- Recuperar las cuentas creadas mediante una solicitud GET fuera del conjunto de cambios.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/$batch HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
Content-Type: multipart/mixed; boundary=batch_22975cad-7f57-410d-be15-
6363209367ea

--batch_22975cad-7f57-410d-be15-6363209367ea
Content-Type: multipart/mixed; boundary=changeset_246e6bfe-89a4-4c77-b293-
7a433f082e8a

--changeset_246e6bfe-89a4-4c77-b293-7a433f082e8a
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-ID: 1

POST /api/data/v9.2/tasks HTTP/1.1
Content-Type: application/json; type=entry

{
    "subject": "Task 1 in batch",
    "regardingobjectid_account_task@odata.bind": "accounts(00000000-0000-0000-
0000-000000000001)"
}
--changeset_246e6bfe-89a4-4c77-b293-7a433f082e8a
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-ID: 2

POST /api/data/v9.2/tasks HTTP/1.1
Content-Type: application/json; type=entry

{
    "subject": "Task 2 in batch",
    "regardingobjectid_account_task@odata.bind": "accounts(00000000-0000-0000-
0000-000000000001)"
}
--changeset_246e6bfe-89a4-4c77-b293-7a433f082e8a
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-ID: 3

POST /api/data/v9.2/tasks HTTP/1.1
Content-Type: application/json; type=entry

{
    "subject": "Task 3 in batch",
    "regardingobjectid_account_task@odata.bind": "accounts(00000000-0000-0000-
0000-000000000001)"
```

```
}
```

```
--changeset_246e6bfe-89a4-4c77-b293-7a433f082e8a--
```

```
--batch_22975cad-7f57-410d-be15-6363209367ea
```

```
Content-Type: application/http
```

```
Content-Transfer-Encoding: binary
```

```
GET /api/data/v9.2/accounts(00000000-0000-0000-0000-000000000001)/Account_Tasks?$select=subject HTTP/1.1
```

```
--batch_22975cad-7f57-410d-be15-6363209367ea--
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

--batchresponse_f27ef42d-51b0-4685-bac9-f468f844de2f
Content-Type: multipart/mixed; boundary=changesetresponse_64cc3fff-023a-45b0-b29d-df21583ffa15

--changesetresponse_64cc3fff-023a-45b0-b29d-df21583ffa15
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-ID: 1

HTTP/1.1 204 No Content
OData-Version: 4.0
Location: [Organization Uri]/api/data/v9.2/tasks(e73ffc82-e292-ed11-aad1-000d3a9933c9)
OData-EntityId: [Organization Uri]/api/data/v9.2/tasks(e73ffc82-e292-ed11-aad1-000d3a9933c9)

--changesetresponse_64cc3fff-023a-45b0-b29d-df21583ffa15
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-ID: 2

HTTP/1.1 204 No Content
OData-Version: 4.0
Location: [Organization Uri]/api/data/v9.2/tasks(e83ffc82-e292-ed11-aad1-000d3a9933c9)
OData-EntityId: [Organization Uri]/api/data/v9.2/tasks(e83ffc82-e292-ed11-aad1-000d3a9933c9)

--changesetresponse_64cc3fff-023a-45b0-b29d-df21583ffa15
Content-Type: application/http
```

```

Content-Transfer-Encoding: binary
Content-ID: 3

HTTP/1.1 204 No Content
OData-Version: 4.0
Location: [Organization Uri]/api/data/v9.2/tasks(e93ffc82-e292-ed11-aad1-
000d3a9933c9)
OData-EntityId: [Organization Uri]/api/data/v9.2/tasks(e93ffc82-e292-ed11-
aad1-000d3a9933c9)

--changesetresponse_64cc3fff-023a-45b0-b29d-df21583ffa15--
--batchresponse_f27ef42d-51b0-4685-bac9-f468f844de2f
Content-Type: application/http
Content-Transfer-Encoding: binary

HTTP/1.1 200 OK
Content-Type: application/json; odata.metadata=minimal; odata.streaming=true
OData-Version: 4.0

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#tasks(subject)",
    "value": [
        {
            "@odata.etag": "W/\"77181173\"",
            "subject": "Task 1 in batch",
            "activityid": "e73ffc82-e292-ed11-aad1-000d3a9933c9"
        },
        {
            "@odata.etag": "W/\"77181176\"",
            "subject": "Task 2 in batch",
            "activityid": "e83ffc82-e292-ed11-aad1-000d3a9933c9"
        },
        {
            "@odata.etag": "W/\"77181179\"",
            "subject": "Task 3 in batch",
            "activityid": "e93ffc82-e292-ed11-aad1-000d3a9933c9"
        }
    ]
}
--batchresponse_f27ef42d-51b0-4685-bac9-f468f844de2f--

```

Hacer referencia a URI en una operación

En los conjuntos de cambios, puede usar `$parameter`, como `$1`, `$2`, etc., para hacer referencia a URI devueltos para nuevas entidades creadas anteriormente en el mismo conjunto de cambios. Para obtener más información, consulte la especificación OData v4.0: [11.7.3.1 Solicitudes de referencia en un conjunto de cambios](#).

Esta sección muestra diferentes ejemplos sobre cómo `$parameter` se puede usar en el cuerpo de la solicitud de una operación por lotes para hacer referencia a URI.

Hacer referencia a URI en el cuerpo de la solicitud

El siguiente ejemplo muestra cómo dos referencias de URI se pueden usar en una sola operación.

Solicitud:

HTTP

```
POST [Organization URI]/api/data/v9.2/$batch HTTP/1.1
Content-Type: multipart/mixed;boundary=batch_AAA123
Accept: application/json
OData-MaxVersion: 4.0
OData-Version: 4.0

--batch_AAA123
Content-Type: multipart/mixed; boundary=changeset_dd81ccab-11ce-4d57-b91d-
12c4e25c3cab

--changeset_dd81ccab-11ce-4d57-b91d-12c4e25c3cab
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-ID: 1

POST [Organization URI]/api/data/v9.2/leads HTTP/1.1
Content-Type: application/json

{
    "firstname": "first name",
    "lastname": "last name"
}

--changeset_dd81ccab-11ce-4d57-b91d-12c4e25c3cab
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-ID: 2

POST [Organization URI]/api/data/v9.2/contacts HTTP/1.1
Content-Type: application/json

{"firstname": "first name"}

--changeset_dd81ccab-11ce-4d57-b91d-12c4e25c3cab
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-ID: 3

POST [Organization URI]/api/data/v9.2/accounts HTTP/1.1
Content-Type: application/json
```

```
{  
    "name": "IcM Account",  
    "originatingleadid@odata.bind": "$1",  
    "primarycontactid@odata.bind": "$2"  
}  
  
--changeset_dd81ccab-11ce-4d57-b91d-12c4e25c3cab--  
--batch_AAA123--
```

Respuesta:

HTTP

HTTP/1.1 200 OK

OData-Version: 4.0

--batchresponse_3cace264-86ea-40fe-83d3-954b336c0f4a

Content-Type: multipart/mixed; boundary=changesetresponse_1a5db8a1-ec98-42c4-81f6-6bc6adcfa4bc

--changesetresponse_1a5db8a1-ec98-42c4-81f6-6bc6adcfa4bc

Content-Type: application/http

Content-Transfer-Encoding: binary

Content-ID: 1

HTTP/1.1 204 No Content

OData-Version: 4.0

Location: [Organization URI]/api/data/v9.2/leads(425195a4-7a75-e911-a97a-000d3a34a1bd)

OData-EntityId: [Organization URI]/api/data/v9.2/leads(425195a4-7a75-e911-a97a-000d3a34a1bd)

--changesetresponse_1a5db8a1-ec98-42c4-81f6-6bc6adcfa4bc

Content-Type: application/http

Content-Transfer-Encoding: binary

Content-ID: 2

HTTP/1.1 204 No Content

OData-Version: 4.0

Location: [Organization URI]/api/data/v9.2/contacts(495195a4-7a75-e911-a97a-000d3a34a1bd)

OData-EntityId: [Organization URI]/api/data/v9.2/contacts(495195a4-7a75-e911-a97a-000d3a34a1bd)

--changesetresponse_1a5db8a1-ec98-42c4-81f6-6bc6adcfa4bc

Content-Type: application/http

Content-Transfer-Encoding: binary

Content-ID: 3

HTTP/1.1 204 No Content

OData-Version: 4.0

Location: [Organization URI]/api/data/v9.2/accounts(4f5195a4-7a75-e911-a97a-

```
000d3a34a1bd)
OData-EntityId: [Organization URI]/api/data/v9.2/accounts(4f5195a4-7a75-
e911-a97a-000d3a34a1bd)

--changesetresponse_1a5db8a1-ec98-42c4-81f6-6bc6adcf4bc--
--batchresponse_3cace264-86ea-40fe-83d3-954b336c0f4a--
```

Hacer referencia a URI en dirección URL de la solicitud

El ejemplo ofrecido a continuación muestra cómo puede hacer referencia a un URI usando `$1` en la dirección URL de una solicitud posterior.

Solicitud:

HTTP

```
POST [Organization URI]/api/data/v9.2/$batch HTTP/1.1
Content-Type: multipart/mixed;boundary=batch_AAA123
Accept: application/json
OData-MaxVersion: 4.0
OData-Version: 4.0

--batch_AAA123
Content-Type: multipart/mixed; boundary=changeset_dd81ccab-11ce-4d57-b91d-
12c4e25c3cab

--changeset_dd81ccab-11ce-4d57-b91d-12c4e25c3cab
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-ID: 1

POST [Organization URI]/api/data/v9.2/contacts HTTP/1.1
Content-Type: application/json

{
  "firstname": "First Name",
  "lastname": "Last name"
}

--changeset_dd81ccab-11ce-4d57-b91d-12c4e25c3cab
Content-Transfer-Encoding: binary
Content-Type: application/http
Content-ID: 2

PUT $1/lastname HTTP/1.1
Content-Type: application/json

{
  "value": "BBBBB"
}
```

```
--changeset_dd81ccab-11ce-4d57-b91d-12c4e25c3cab--  
--batch_AAA123--
```

Respuesta:

HTTP

HTTP/1.1 200 OK

OData-Version: 4.0

--batchresponse_2cb48f48-39a8-41ea-aa52-132fa8ab3c2d

Content-Type: multipart/mixed; boundary=changesetresponse_d7528170-3ef3-41bd-be8e-eac971a8d9d4

--changesetresponse_d7528170-3ef3-41bd-be8e-eac971a8d9d4

Content-Type: application/http

Content-Transfer-Encoding: binary

Content-ID: 1

HTTP/1.1 204 No Content

OData-Version: 4.0

Location:[Organization URI]/api/data/v9.2/contacts(f8ea5d2c-8c75-e911-a97a-000d3a34a1bd)

OData-EntityId:[Organization URI]/api/data/v9.2/contacts(f8ea5d2c-8c75-e911-a97a-000d3a34a1bd)

--changesetresponse_d7528170-3ef3-41bd-be8e-eac971a8d9d4

Content-Type: application/http

Content-Transfer-Encoding: binary

Content-ID: 2

HTTP/1.1 204 No Content

OData-Version: 4.0

--changesetresponse_d7528170-3ef3-41bd-be8e-eac971a8d9d4--

--batchresponse_2cb48f48-39a8-41ea-aa52-132fa8ab3c2d--

Hacer referencia a URI en la dirección URL y el cuerpo de la solicitud usando @odata.id

El ejemplo siguiente muestra cómo vincular un registro de entidad Contacto a un registro de entidad Cuenta. Al URI del registro de entidad Cuenta se hace referencia como \$1 y al URI del registro de entidad Contacto se hace referencia como \$2.

Solicitud:

HTTP

```

POST [Organization URI]/api/data/v9.2/$batch HTTP/1.1
Content-Type:multipart/mixed;boundary=batch_AAA123
Accept:application/json
OData-MaxVersion:4.0
OData-Version:4.0

--batch_AAA123
Content-Type: multipart/mixed; boundary=changeset_dd81ccab-11ce-4d57-b91d-
12c4e25c3cab

--changeset_dd81ccab-11ce-4d57-b91d-12c4e25c3cab
Content-Type:application/http
Content-Transfer-Encoding:binary
Content-ID:1

POST [Organization URI]/api/data/v9.2/accounts HTTP/1.1
Content-Type: application/json

{ "name":"Account Name"}

--changeset_dd81ccab-11ce-4d57-b91d-12c4e25c3cab
Content-Type:application/http
Content-Transfer-Encoding:binary
Content-ID:2

POST [Organization URI]/api/data/v9.2/contacts HTTP/1.1
Content-Type:application/json

{ "firstname":"Contact first name"}

--changeset_dd81ccab-11ce-4d57-b91d-12c4e25c3cab
Content-Type:application/http
Content-Transfer-Encoding:binary
Content-ID:3

PUT $1/primarycontactid/$ref HTTP/1.1
Content-Type:application/json

{"@odata.id":"$2"}

--changeset_dd81ccab-11ce-4d57-b91d-12c4e25c3cab--
--batch_AAA123--

```

Respuesta:

HTTP

```

HTTP/1.1 200 OK
OData-Version: 4.0

--batchresponse_0740a25c-d8e1-41a5-9202-1b50a297864c
Content-Type: multipart/mixed; boundary=changesetresponse_19ca0da8-d8bb-

```

```
4273-a3f7-fe0d0fadfe5f
```

```
--changesetresponse_19ca0da8-d8bb-4273-a3f7-fe0d0fadfe5f
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-ID: 1

HTTP/1.1 204 No Content
OData-Version: 4.0
Location:[Organization URI]/api/data/v9.2/accounts(3dcf8c02-8c75-e911-a97a-
000d3a34a1bd)
OData-EntityId:[Organization URI]/api/data/v9.2/accounts(3dcf8c02-8c75-e911-
a97a-000d3a34a1bd)

--changesetresponse_19ca0da8-d8bb-4273-a3f7-fe0d0fadfe5f
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-ID: 2

HTTP/1.1 204 No Content
OData-Version: 4.0
Location:[Organization URI]/api/data/v9.2/contacts(43cf8c02-8c75-e911-a97a-
000d3a34a1bd)
OData-EntityId:[Organization URI]/api/data/v9.2/contacts(43cf8c02-8c75-e911-
a97a-000d3a34a1bd)

--changesetresponse_19ca0da8-d8bb-4273-a3f7-fe0d0fadfe5f
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-ID: 3

HTTP/1.1 204 No Content
OData-Version: 4.0

--changesetresponse_19ca0da8-d8bb-4273-a3f7-fe0d0fadfe5f--
--batchresponse_0740a25c-d8e1-41a5-9202-1b50a297864c--
```

Hacer referencia a URI en propiedades de dirección URL y navegación

El ejemplo siguiente muestra cómo usar el URI de la organización de un registro de contacto y vincularlo a un registro de cuenta utilizando la propiedad de navegación de un solo valor `primarycontactid`. Al URI del registro de entidad Cuenta se hace referencia como `$1` y al URI del registro de entidad Contacto se hace referencia como `$2` en la solicitud `PATCH`.

Solicitud:

```
HTTP
```

```

POST [Organization URI]/api/data/v9.2/$batch HTTP/1.1
Content-Type:multipart/mixed;boundary=batch_AAA123
Accept:application/json
OData-MaxVersion:4.0
OData-Version:4.0

--batch_AAA123
Content-Type: multipart/mixed; boundary=changeset_dd81ccab-11ce-4d57-b91d-
12c4e25c3cab

--changeset_dd81ccab-11ce-4d57-b91d-12c4e25c3cab
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-ID: 1

POST [Organization URI]/api/data/v9.2/accounts HTTP/1.1
Content-Type: application/json

{ "name":"Account name"}

--changeset_dd81ccab-11ce-4d57-b91d-12c4e25c3cab
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-ID: 2

POST [Organization URI]/api/data/v9.2/contacts HTTP/1.1
Content-Type: application/json

{
  "firstname":"Contact first name"
}

--changeset_dd81ccab-11ce-4d57-b91d-12c4e25c3cab
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-ID: 3

PATCH $1 HTTP/1.1
Content-Type: application/json

{
  "primarycontactid@odata.bind": "$2"
}

--changeset_dd81ccab-11ce-4d57-b91d-12c4e25c3cab--
--batch_AAA123--

```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
```

```
OData-Version: 4.0
```

```
--batchresponse_9595d3ae-48f6-414f-a3aa-a3a33559859e
```

```
Content-Type: multipart/mixed; boundary=changesetresponse_0c1567a5-ad0d-48fa-b81d-e6db05cad01c
```

```
--changesetresponse_0c1567a5-ad0d-48fa-b81d-e6db05cad01c
```

```
Content-Type: application/http
```

```
Content-Transfer-Encoding: binary
```

```
Content-ID: 1
```

```
HTTP/1.1 204 No Content
```

```
OData-Version: 4.0
```

```
Location: [Organization URI]/api/data/v9.2/accounts(6cd81853-7b75-e911-a97a-000d3a34a1bd)
```

```
OData-EntityId: [Organization URI]/api/data/v9.2/accounts(6cd81853-7b75-e911-a97a-000d3a34a1bd)
```

```
--changesetresponse_0c1567a5-ad0d-48fa-b81d-e6db05cad01c
```

```
Content-Type: application/http
```

```
Content-Transfer-Encoding: binary
```

```
Content-ID: 2
```

```
HTTP/1.1 204 No Content
```

```
OData-Version: 4.0
```

```
Location: [Organization URI]/api/data/v9.2/contacts(6ed81853-7b75-e911-a97a-000d3a34a1bd)
```

```
OData-EntityId: [Organization URI]/api/data/v9.2/contacts(6ed81853-7b75-e911-a97a-000d3a34a1bd)
```

```
--changesetresponse_0c1567a5-ad0d-48fa-b81d-e6db05cad01c
```

```
Content-Type: application/http
```

```
Content-Transfer-Encoding: binary
```

```
Content-ID: 3
```

```
HTTP/1.1 204 No Content
```

```
OData-Version: 4.0
```

```
Location: [Organization URI]/api/data/v9.2/accounts(6cd81853-7b75-e911-a97a-000d3a34a1bd)
```

```
OData-EntityId: [Organization URI]/api/data/v9.2/accounts(6cd81853-7b75-e911-a97a-000d3a34a1bd)
```

```
--changesetresponse_0c1567a5-ad0d-48fa-b81d-e6db05cad01c--
```

```
--batchresponse_9595d3ae-48f6-414f-a3aa-a3a33559859e--
```

ⓘ Nota

Al hacer referencia a un `Content-ID` antes de que se haya declarado en el cuerpo de la solicitud se devolverá la solicitud errónea **HTTP 400** del error.

El ejemplo siguiente muestra un cuerpo de la solicitud que puede producir este error.

Cuerpo de la solicitud

HTTP

```
--batch_AAA123
Content-Type: multipart/mixed; boundary=changeset_BBB456

--changeset_BBB456
Content-Type: application/http
Content-Transfer-Encoding:binary
Content-ID: 2

POST [Organization URI]/api/data/v9.2/phonecalls HTTP/1.1
Content-Type: application/json;type=entry

{
    "phonenumber":"911",
    "regardingobjectid_account_phonecall@odata.bind" : "$1"
}

--changeset_BBB456
Content-Type: application/http
Content-Transfer-Encoding:binary
Content-ID: 1

POST [Organization URI]/api/data/v9.2/accounts HTTP/1.1
Content-Type: application/json;type=entry

{
    "name":"QQQQ",
    "revenue": 1.50
}

--changeset_BBB456--
--batch_AAA123--
```

Respuesta:

HTTP

```
HTTP 400 Bad Request
Content-ID Reference: '$1' does not exist in the batch context.
```

Control de errores

Cuando ocurre un error para una solicitud dentro de un lote, el error de esa solicitud se devuelve para la solicitud de lote y las solicitudes adicionales no se procesan.

Puede usar el encabezado de solicitud `Prefer: odata.continue-on-error` para especificar que se procesen solicitudes adicionales cuando se produzcan errores. La solicitud por lotes devuelve `200 OK` y los errores de respuesta individuales se devuelven en el cuerpo de la respuesta del lote.

Más información: [Especificación OData: 8.2.8.3 Preference odata.continue-on-error ↗](#)

Ejemplo

El siguiente ejemplo intenta crear tres registros de tareas asociados con una cuenta con `accountid` igual a `0000000-0000-0000-0000-000000000001`, pero la longitud de la propiedad `subject` para la primera tarea es demasiado larga.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/$batch HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
Content-Type: multipart/mixed; boundary="batch_431faf5a-f979-4ee6-a374-
d242f8962d41"
Content-Length: 1335

--batch_431faf5a-f979-4ee6-a374-d242f8962d41
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-Length: 436

POST /api/data/v9.2/tasks HTTP/1.1
Content-Type: application/json; type=entry

{
    "subject": "Subject is too long
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx",
    "regardingobjectid_account_task@odata.bind": "accounts(0000000-0000-0000-
0000-000000000001)"
}
--batch_431faf5a-f979-4ee6-a374-d242f8962d41
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-Length: 250
```

```

POST /api/data/v9.2/tasks HTTP/1.1
Content-Type: application/json; type=entry

{
    "subject": "Task 2 in batch",
    "regardingobjectid_account_task@odata.bind": "accounts(00000000-0000-0000-0000-000000000001)"
}
--batch_431faf5a-f979-4ee6-a374-d242f8962d41
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-Length: 250

POST /api/data/v9.2/tasks HTTP/1.1
Content-Type: application/json; type=entry

{
    "subject": "Task 3 in batch",
    "regardingobjectid_account_task@odata.bind": "accounts(00000000-0000-0000-0000-000000000001)"
}
--batch_431faf5a-f979-4ee6-a374-d242f8962d41--

```

Sin configurar el encabezado de solicitud `Prefer: odata.continue-on-error`, el lote falla en la primera solicitud del lote. El error por lotes representa el error de la primera solicitud fallida.

Respuesta:

HTTP

```

HTTP/1.1 400 BadRequest
OData-Version: 4.0

--batchresponse_156da4b8-cd2c-4862-a911-4aaab97c001a
Content-Type: application/http
Content-Transfer-Encoding: binary

HTTP/1.1 400 Bad Request
REQ_ID: 5ecd1cb3-1730-4ffc-909c-d44c22270026
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0

{"error":{"code":"0x80044331","message":"A validation error occurred. The length of the 'subject' attribute of the 'task' entity exceeded the maximum allowed length of '200'."}}
--batchresponse_156da4b8-cd2c-4862-a911-4aaab97c001a--

```

Cuando el encabezado de solicitud `Prefer: odata.continue-on-error` se aplica a la solicitud por lotes, esta se realiza correctamente con un estado de `200 OK` y el error de la primera solicitud se devuelve como parte del cuerpo.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/$batch HTTP/1.1
Prefer: odata.continue-on-error
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
Content-Type: multipart/mixed; boundary=batch_662d4610-7f12-4895-ac4a-
3fdf77cc10a1
Content-Length: 1338

--batch_662d4610-7f12-4895-ac4a-3fdf77cc10a1
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-Length: 439

POST /api/data/v9.2/tasks HTTP/1.1
Content-Type: application/json; type=entry

{
    "subject": "Subject is too long
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx",
    "regardingobjectid_account_task@odata.bind": "accounts(00000000-0000-0000-
0000-000000000001)"
}
--batch_662d4610-7f12-4895-ac4a-3fdf77cc10a1
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-Length: 250

POST /api/data/v9.2/tasks HTTP/1.1
Content-Type: application/json; type=entry

{
    "subject": "Task 2 in batch",
    "regardingobjectid_account_task@odata.bind": "accounts(00000000-0000-0000-
0000-000000000001)"
}
--batch_662d4610-7f12-4895-ac4a-3fdf77cc10a1
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-Length: 250

POST /api/data/v9.2/tasks HTTP/1.1
```

```
Content-Type: application/json; type=entry

{
  "subject": "Task 3 in batch",
  "regardingobjectid_account_task@odata.bind": "accounts(00000000-0000-0000-0000-000000000001)"
}
--batch_662d4610-7f12-4895-ac4a-3fdf77cc10a1--
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

--batchresponse_f44bd09d-573f-4a30-bca0-2e500ee7e139
Content-Type: application/http
Content-Transfer-Encoding: binary

HTTP/1.1 400 Bad Request
REQ_ID: de4c5227-4a28-4ebd-8ced-3392ece1697b
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0

{"error":{"code":"0x80044331","message":"A validation error occurred. The length of the 'subject' attribute of the 'task' entity exceeded the maximum allowed length of '200'."}}
--batchresponse_f44bd09d-573f-4a30-bca0-2e500ee7e139
Content-Type: application/http
Content-Transfer-Encoding: binary

HTTP/1.1 204 No Content
OData-Version: 4.0
Location: [Organization Uri]/api/data/v9.2/tasks(aed2ae8b-3c94-ed11-aad1-000d3a9933c9)
OData-EntityId: [Organization Uri]/api/data/v9.2/tasks(aed2ae8b-3c94-ed11-aad1-000d3a9933c9)

--batchresponse_f44bd09d-573f-4a30-bca0-2e500ee7e139
Content-Type: application/http
Content-Transfer-Encoding: binary

HTTP/1.1 204 No Content
OData-Version: 4.0
Location: [Organization Uri]/api/data/v9.2/tasks(b181a991-3c94-ed11-aad1-000d3a9933c9)
OData-EntityId: [Organization Uri]/api/data/v9.2/tasks(b181a991-3c94-ed11-aad1-000d3a9933c9)
```

Métodos auxiliares de .NET

La [biblioteca de clases WebAPIService \(C#\)](#) es un proyecto de biblioteca de clases auxiliares de muestra que se usa para muestras de API web escritas en .NET. Demuestra una forma en que se pueden reutilizar los patrones comunes que se usan con la API web.

ⓘ Nota

Esta biblioteca de muestras es una ayuda que utilizan todos los ejemplos de API web de C# de Dataverse, pero no es un SDK. Se prueba solo para confirmar que las muestras que lo usan se ejecutan correctamente. Este código de muestra se proporciona 'tal cual' sin garantía de reutilización.

Esta biblioteca incluye clases para crear solicitudes por lotes y procesar respuestas. Por ejemplo, se usaron variaciones en el siguiente código para generar muchos de los ejemplos de solicitud y respuesta HTTP en este artículo.

C#

```
using PowerApps.Samples;
using PowerApps.Samples.Batch;

static async Task Main()
{
    Config config = App.InitializeApp();

    var service = new Service(config);

    JObject account = new()
    {
        {"name", "test account" }
    };

    EntityReference accountRef = await service.Create("accounts", account);

    List<HttpRequestMessage> createRequests = new() {
        new CreateRequest("tasks",new JObject(){
            {"subject","Task 2 in batch" },
            {"regardingobjectid_account_task@odata.bind", accountRef.Path }
        }),
        new CreateRequest("tasks",new JObject(){
            {"subject","Task 2 in batch" },
            {"regardingobjectid_account_task@odata.bind", accountRef.Path }
        })
    };
}
```

```

        {"regardingobjectid_account_task@odata.bind", accountRef.Path }
    }),
    new CreateRequest("tasks",new JObject(){
        {"subject","Task 3 in batch" },
        {"regardingobjectid_account_task@odata.bind", accountRef.Path }
    })
};

BatchRequest batchRequest = new(service.BaseAddress)
{
    Requests = createRequests,
    ContinueOnError = true
};

var batchResponse = await service.SendAsync<BatchResponse>
(batchRequest);

batchResponse.HttpResponseMessages.ForEach(response => {

    string path = response.As<CreateResponse>().EntityReference.Path;
    Console.WriteLine($"Task created at: {path}");

});
}

```

output

```

Task created at: tasks(6743adfa-4a94-ed11-aad1-000d3a9933c9)
Task created at: tasks(6843adfa-4a94-ed11-aad1-000d3a9933c9)
Task created at: tasks(6943adfa-4a94-ed11-aad1-000d3a9933c9)

```

Dentro de esta biblioteca, hay algunos métodos que pueden resultarle útiles en su código .NET.

Más información:

- [Lote WebAPIService](#)
- [Ejemplos de operaciones de datos de API web \(C#\)](#)

Ejemplo de .NET HttpResponseMessage a HttpMessageContent

En .NET, debe enviar solicitudes en lote como [MultipartContent](#), que es una colección de [HttpContent](#). [HttpMessageContent](#) hereda de [HttpContent](#). La [biblioteca de clases WebAPIService \(C#\)](#) clase [BatchRequest](#) ↗ utiliza el siguiente método estático privado

`ToMessageContent` para convertir `HttpRequestMessage` a `HttpMessageContent` que se pueden agregar a `MultipartContent`.

C#

```
/// <summary>
/// Converts a HttpRequestMessage to HttpMessageContent
/// </summary>
/// <param name="request">The HttpRequestMessage to convert.</param>
/// <returns>HttpMessageContent with the correct headers.</returns>
private HttpMessageContent ToMessageContent(HttpRequestMessage request)
{
    //Relative URI is not allowed with MultipartContent
    request.RequestUri = new Uri(
        baseUri: ServiceBaseAddress,
        relativeUri: request.RequestUri.ToString());

    if (request.Content != null)
    {
        if (request.Content.Headers.Contains("Content-Type"))
        {
            request.Content.Headers.Remove("Content-Type");
        }
        request.Content.Headers.Add("Content-Type",
"application/json;type=entry");
    }

    HttpMessageContent messageContent = new(request);

    if (messageContent.Headers.Contains("Content-Type"))
    {
        messageContent.Headers.Remove("Content-Type");
    }
    messageContent.Headers.Add("Content-Type", "application/http");
    messageContent.Headers.Add("Content-Transfer-Encoding", "binary");

    return messageContent;
}
```

Ejemplo de respuesta en lote del analizador de .NET

La biblioteca de clases `WebAPIService` (C#) la clase `BatchResponse` utiliza el siguiente método estático privado `ParseMultipartContent` para analizar el cuerpo de una respuesta en lote en un `List` de `HttpResponseMessage` que se pueden procesar como respuestas individuales.

C#

```

/// <summary>
/// Processes the Multi-part content returned from the batch into a list of
responses.
/// </summary>
/// <param name="content">The Content of the response.</param>
/// <returns></returns>
private static async Task<List<HttpResponseMessage>>
ParseMultipartContent(HttpContent content)
{
    MultipartMemoryStreamProvider batchResponseContent = await
content.ReadAsMultipartAsync();

    List<HttpResponseMessage> responses = new();

    if (batchResponseContent?.Contents != null)
    {
        batchResponseContent.Contents.ToList().ForEach(async httpContent =>
        {

            //This is true for changesets
            if (httpContent.IsMimeMultipartContent())
            {
                //Recursive call
                responses.AddRange(await ParseMultipartContent(httpContent));
            }

            //This is for individual responses outside of a change set.
            else
            {
                //Must change Content-Type for ReadAs HttpResponseMessageAsync
method to work.
                httpContent.Headers.Remove("Content-Type");
                httpContent.Headers.Add("Content-Type",
"application/http;msgtype=response");

                HttpResponseMessage httpResponseMessage = await
httpContent.ReadAs HttpResponseMessageAsync();

                if (httpResponseMessage != null)
                {
                    responses.Add(httpResponseMessage);
                }
            }
        });
    }

    return responses;
}

```

Consulte también

 Nota

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta. [↗](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad ↗](#)).

Suplantar a otro usuario utilizando la API web

Artículo • 02/11/2023

Hay ocasiones en las que el código deberá realizar operaciones en nombre de otro usuario. Si la cuenta del sistema que ejecuta el código tiene los privilegios adecuados, puede realizar operaciones en nombre de otros usuarios.

Requisitos de la suplantación

La suplantación se utiliza para ejecutar la lógica de negocios (código) en nombre de otro usuario de Microsoft Dataverse para proporcionar una característica o servicio deseado con el rol que le corresponde y la seguridad basada en objetos de dicho usuario suplantado. Esto es necesario porque varios clientes y servicios en nombre de un usuario de Dataverse pueden llamar a los servicios web de Dataverse, por ejemplo, en un flujo de trabajo o una solución de ISV personalizada. La suplantación implica dos cuentas distintas de usuario: una cuenta de usuario (A) que se usa al ejecutar código para realizar algunas tareas en nombre de otro usuario (B).

La cuenta de usuario (A) debe tener el privilegio `prvActOnBehalfOfAnotherUser`, que se incluye en el rol de seguridad Delegado. El conjunto real de privilegios que se utiliza para modificar los datos es la intersección de los privilegios que el usuario con rol de delegado posee con la del usuario que se suplanta. Es decir, al usuario (A) se le permite hacer algo solo si el usuario (A) y el usuario (B) tienen el privilegio necesario para realizar la acción.

Cómo suplantar a un usuario

Hay dos formas de suplantar a un usuario, ambas posibles pasando un encabezado con el correspondiente identificador de usuario.

- 1. Preferido:** Suplanta a un usuario en función de la identificación de objeto de Microsoft Entra ID pasando ese valor junto con el `CallerObjectId` del encabezado.
- 2. Datos heredados:** Para suplantar a un usuario en función de su `systemuserid` puede aprovechar `MSCRMCallerID` con el valor de guid correspondiente.

En este ejemplo, se crea una nueva entidad de cuenta en nombre del usuario con un identificador de objeto de Microsoft Entra ID `e39c5d16-675b-48d1-8e67-667427e9c084`.

Solicitud:

HTTP

```
POST [Organization URI]/api/data/v9.0/accounts HTTP/1.1
CallerObjectId: e39c5d16-675b-48d1-8e67-667427e9c084
Accept: application/json
Content-Type: application/json; charset=utf-8
OData-MaxVersion: 4.0
OData-Version: 4.0

{"name": "Sample Account created using impersonation"}
```

Respuesta:

HTTP

```
HTTP/1.1 204 No Content
OData-Version: 4.0
OData-EntityId: [Organization URI]/api/data/v9.0/accounts(00000000-0000-
0000-000000000003)
```

Determinar el usuario real

Cuando una operación como crear una entidad se realiza mediante suplantación, el usuario que realiza la operación se puede encontrar consultando el registro incluida la propiedad de navegación de un solo valor `createdonbehalfby`. Una propiedad de navegación de un solo valor `modifiedonbehalfby` correspondiente está disponible para operaciones que actualizan la entidad.

Solicitud:

HTTP

```
GET [Organization URI]/api/data/v9.0/accounts(00000000-0000-0000-
000000000003)?
$select=name&$expand=createdby($select=fullname),createdonbehalfby($select=f
ullname),owninguser($select=fullname) HTTP/1.1
Accept: application/json
OData-MaxVersion: 4.0
OData-Version: 4.0
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
Content-Type: application/json; odata.metadata=minimal
ETag: W/"506868"

{
    "@odata.context": "[Organization
URI]/api/data/v9.0/$metadata#accounts(name,createdby(fullname,azureactivedirectoryobjectid),createdonbehalfby(fullname,azureactivedirectoryobjectid),owninguser(fullname,azureactivedirectoryobjectid))/$entity",
    "@odata.etag": "W/\"2751197\"",
    "name": "Sample Account created using impersonation",
    "accountid": "00000000-0000-0000-000000000003",
    "createdby": {
        "@odata.etag": "W/\"2632435\"",
        "fullname": "Impersonated User",
        "azureactivedirectoryobjectid": "e39c5d16-675b-48d1-8e67-667427e9c084",
        "systemuserid": "75df116d-d9da-e711-a94b-000d3a34ed47",
        "ownerid": "75df116d-d9da-e711-a94b-000d3a34ed47"
    },
    "createdonbehalfby": {
        "@odata.etag": "W/\"2632445\"",
        "fullname": "Actual User",
        "azureactivedirectoryobjectid": "3d8bed3e-79a3-47c8-80cf-269869b2e9f0",
        "systemuserid": "278742b0-1e61-4fb5-84ef-c7de308c19e2",
        "ownerid": "278742b0-1e61-4fb5-84ef-c7de308c19e2"
    },
    "owninguser": {
        "@odata.etag": "W/\"2632435\"",
        "fullname": "Impersonated User",
        "azureactivedirectoryobjectid": "e39c5d16-675b-48d1-8e67-667427e9c084",
        "systemuserid": "75df116d-d9da-e711-a94b-000d3a34ed47",
        "ownerid": "75df116d-d9da-e711-a94b-000d3a34ed47"
    }
}
```

Consulte también

[Suplantar a otro usuario](#)

[Suplantar a otro usuario utilizando SDK para .NET](#)

[Realizar operaciones mediante la API web](#)

[Componer solicitudes HTTP y administrar errores](#)

[Consultar datos utilizando la API web](#)

[Crear una fila de tabla usando la API web](#)

[Recuperar una fila de tabla usando la API web](#)

[Actualizar y eliminar filas de tablas usando la API web](#)

[Asociar y anular la asociación de filas de tabla mediante la API web](#)

[Usar funciones de la API web](#)

[Usar acciones de la API web](#)

[Ejecute las operaciones por lotes mediante API web](#)

[Realizar operaciones condicionales mediante la API web](#)

 **Nota**

¿Puede indicarnos sus preferencias de idioma de documentación? [Realice una breve encuesta.](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad](#)).

Realizar operaciones condicionales mediante la API web

Artículo • 23/02/2023

Microsoft Dataverse ofrece compatibilidad con un conjunto de operaciones condicionales que dependen del mecanismo de control de versiones de recursos HTTP estándar conocido como *ETags*.

ETags

El protocolo HTTP define una *etiqueta de entidad* o [ETag](#) en el término abreviado, para identificar versiones específicas de un recurso. Las ETags son identificadores opacos cuyos valores exactos dependen de la implementación. Los valores de ETag se producen en dos variedades: validación segura y débil. La validación segura indica que un recurso único, identificado mediante un URI específico, será idéntico en el nivel binario si su valor de ETag correspondiente no se modifica. La validación débil solo garantiza que la representación del recurso equivale semánticamente al mismo valor de ETag.

Dataverse genera una propiedad `@odata.etag` de validación débil para todas las instancias de entidad y esta propiedad se devuelve automáticamente con cada registro de entidad recuperado. Para más información, consulte [Recuperar una fila de tabla mediante la API web](#).

Encabezados If-Match e If-None-Match

Use los encabezados [If-Match](#) e [If-None-Match](#) con valores ETag para comprobar si la versión actual de un recurso coincide con la recuperada por última vez, coincide con cualquier versión anterior o no coincide con ninguna versión. Estas comparaciones conforman la base de la compatibilidad de operaciones condicional. Dataverse proporciona ETags para admitir las recuperaciones condicionales, la simultaneidad optimista y las operaciones upsert limitadas.

Advertencia

El código de cliente no debe proporcionar ningún significado al valor específico de una ETag, ni a ninguna relación aparente entre las ETags más allá de la igualdad o la desigualdad. Por ejemplo, no se garantiza que un valor de ETag para una versión más reciente de un recurso sea mayor que el valor de ETag para una versión

anterior. Además, el algoritmo usado para generar nuevos valores de ETag puede cambiar desigualdad entre las versiones de un servicio.

Recuperaciones condicionales

Las Etags le permiten optimizar recuperaciones de registros siempre que tenga acceso al mismo registro varias veces. Si ha recuperado previamente un registro, puede pasar el valor de ETag con el encabezado `If-None-Match` para solicitar que se recuperen datos solo si han cambiado desde la última vez que se recuperaron. Si los datos han cambiado, la solicitud devuelve un estado HTTP de `200 OK` con los datos más recientes en el cuerpo de la solicitud. Si los datos no han cambiado, se devuelve el código de estado HTTP `304 Not Modified` para indicar que la entidad no se ha modificado.

El siguiente par de mensajes de ejemplo devuelve datos de una entidad de cuenta con el `accountid` igual a `00000000-0000-0000-0000-000000000001` cuando los datos no se han modificado desde que se recuperaron por última vez cuando el valor Etag era `W/"468026"`

Solicitud

```
HTTP

GET [Organization URI]/api/data/v9.0/accounts(00000000-0000-0000-0000-000000000001)?
$select=accountcategorycode,accountnumber,creditonhold,createdon,numberofemployees,name,revenue    HTTP/1.1
Accept: application/json
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: W/"468026"
```

Respuesta

```
HTTP

HTTP/1.1 304 Not Modified
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0
```

Las siguientes secciones describen las limitaciones para usar recuperaciones condicionales.

La tabla debe tener habilitada la simultaneidad optimista

Compruebe si una entidad tiene simultaneidad optimista habilitada mediante la solicitud de API Web que se muestra a continuación. Las entidades que tienen simultaneidad optimista habilitada, tendrán la propiedad `EntityMetadata.IsOptimisticConcurrencyEnabled` para establecer `true`.

HTTP

```
GET [Organization URI]/api/data/v9.0/EntityDefinitions(LogicalName='<Entity Logical Name>')?$select=IsOptimisticConcurrencyEnabled
```

La consulta no debe incluir \$expand

El Etag solo puede detectar si ha cambiado el registro único que se está recuperando. Cuando utilice `$expand` en su consulta, pueden devolverse registros adicionales y no es posible detectar si alguno de esos registros ha cambiado o no. Si la consulta incluye `$expand`, nunca devolverá `304 Not Modified`.

La consulta no debe incluir anotaciones

Cuando el encabezado `Prefer: odata.include-annotations` se incluye con una solicitud `GET`, nunca devolverá `304 Not Modified`. Los valores de las anotaciones pueden referirse a valores de registros relacionados. Estos registros pueden haber cambiado y este cambio no se pudo detectar, por lo que sería incorrecto indicar que nada ha cambiado.

Limitar operaciones de upsert

Una operación upsert funciona normalmente creando una entidad si no existe; de lo contrario, actualiza una entidad existente. Sin embargo, las ETags se pueden usar para restringir más upserts para evitar creaciones o actualizaciones.

Evitar crear en upsert

Si actualiza datos y hay alguna posibilidad de que se eliminó la entidad forma intencionada, no resulta conveniente volver a crear la entidad. Para evitar este problema, agregue un encabezado `If-Match` a la solicitud con un valor de `*`.

Solicitar

HTTP

```
PATCH [Organization URI]/api/data/v9.0/accounts(00000000-0000-0000-0000-000000000001) HTTP/1.1
Content-Type: application/json
OData-MaxVersion: 4.0
OData-Version: 4.0
If-Match: *

{
    "name": "Updated Sample Account",
    "creditonhold": true,
    "address1_latitude": 47.639583,
    "description": "This is the updated description of the sample account",
    "revenue": 6000000,
    "accountcategorycode": 2
}
```

Response

Si se encuentra la entidad, recibirá una respuesta normal con estado `204 No Content`. Si no se encuentra la entidad, recibirá la siguiente respuesta con estado `404 Not Found`.

HTTP

```
HTTP/1.1 404 Not Found
OData-Version: 4.0
Content-Type: application/json; odata.metadata=minimal

{
    "error": {
        "code": "",
        "message": "account With Id = 00000000-0000-0000-0000-000000000001 Does
Not Exist"
    }
}
```

Evitar actualizar en upsert

Si está insertando datos, hay posibilidad de que un registro con el mismo valor `id` ya exista en el sistema y es posible que no desee actualizarlo. Para evitar este problema, agregue un encabezado `If-None-Match` a la solicitud con un valor de `"*"`.

Solicitud

HTTP

```
PATCH [Organization URI]/api/data/v9.0/accounts(00000000-0000-0000-0000-000000000001) HTTP/1.1
Content-Type: application/json
```

```
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: *

{
    "name": "Updated Sample Account",
    "creditonhold": true,
    "address1_latitude": 47.639583,
    "description": "This is the updated description of the sample account",
    "revenue": 6000000,
    "accountcategorycode": 2
}
```

Response

Si no se encuentra la entidad, recibirá una respuesta normal con estado 204 No Content.
Si se encuentra la entidad, recibirá la siguiente respuesta con estado 412 Precondition Failed.

HTTP

```
HTTP/1.1 412 Precondition Failed
OData-Version: 4.0
Content-Type: application/json; odata.metadata=minimal

{
    "error": {
        "code": "",
        "message": "A record with matching key values already exists."
    }
}
```

Aplicar simultaneidad optimista

Puede usar simultaneidad optimista para detectar si ha modificado una entidad desde que se recuperó por última vez. Si la entidad que piensa actualizar o eliminar ha cambiado en el servidor desde que la recuperó, no le conviene completar la operación de actualización o eliminación. Aplicando el patrón mostrado aquí puede detectar esta situación, recuperar la versión más reciente de la entidad, y aplicar los criterios necesarios para reevaluar si intentar de nuevo la operación.

Aplicar simultaneidad optimista al eliminar

La siguiente solicitud de eliminación de una cuenta con accountid de 00000000-0000-0000-0000-000000000001 da error porque el valor de ETag enviado con el encabezado If-

`Match` es diferente del valor actual. Si el valor hubiera coincidido, se esperaría un estado `204 No Content`.

Solicitar

HTTP

```
DELETE [Organization URI]/api/data/v9.0/accounts(00000000-0000-0000-0000-000000000001) HTTP/1.1
If-Match: W/"470867"
Accept: application/json
OData-MaxVersion: 4.0
OData-Version: 4.0
```

Response

JSON

```
HTTP/1.1 412 Precondition Failed
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0

{
  "error": {
    "code": "", "message": "The version of the existing record doesn't match
the RowVersion property provided."
  }
}
```

Aplicar simultaneidad optimista al actualizar

La siguiente solicitud de actualización de una cuenta con `accountid` de `00000000-0000-0000-0000-000000000001` da error porque el valor de ETag enviado con el encabezado `If-Match` es diferente del valor actual. Si el valor hubiera coincidido, se esperaría un estado `204 No Content`.

Solicitar

HTTP

```
PATCH [Organization URI]/api/data/v9.0/accounts(00000000-0000-0000-0000-000000000001) HTTP/1.1
If-Match: W/"470867"
Accept: application/json
OData-MaxVersion: 4.0
OData-Version: 4.0
```

```
{"name": "Updated Account Name"}
```

Response

JSON

```
HTTP/1.1 412 Precondition Failed
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0

{
  "error": {
    "code": "", "message": "The version of the existing record doesn't match
the RowVersion property provided."
  }
}
```

Consulte también

[Ejemplo de operaciones condicionales de la API web \(C#\)](#)

[Ejemplo de operaciones condicionales de la API web \(JavaScript del lado del cliente\)](#)

[Realizar operaciones mediante la API web](#)

[Componer solicitudes HTTP y administrar errores](#)

[Consultar datos utilizando la API web](#)

[Crear una fila de tabla usando la API web](#)

[Recuperar una fila de tabla usando la API web](#)

[Actualizar y eliminar filas de tablas usando la API web](#)

[Asociar y anular la asociación de filas de tabla mediante la API web](#)

[Usar funciones de la API web](#)

[Usar acciones de la API web](#)

[Ejecutar las operaciones por lotes mediante API web](#)

[Suplantar a otro usuario utilizando la API web](#)

ⓘ Nota

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta. [↗](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad ↗](#)).

Detección de datos duplicados con la API web

Artículo • 29/09/2023

Puede detectar filas de tabla duplicadas de una fila existente para mantener la integridad de los datos. Para obtener información detallada sobre Detección de datos duplicados usando código, consulte [Detectar datos duplicados con código](#)

Detectar duplicados durante las operaciones de creación

Utilice el encabezado de solicitud `MSCRM.SuppressDuplicateDetection` con una solicitud `POST` para detectar la creación de un registro duplicado de un registro existente. El valor asignado al encabezado `MSCRM.SuppressDuplicateDetection` determina si se puede completar la operación de crear o actualizar:

- `true`: crear o actualizar el registro, si se encuentra un duplicado.
- `false`: no crear o actualizar el registro si se encuentra un duplicado.

Use encabezado de preferencia `MSCRM.SuppressDuplicateDetection` y establezca el valor en `false` en la solicitud de la API de la Web.

⚠️ Nota

Asegúrese de que existen reglas de detección de duplicados adecuadas. Dataverse incluye reglas de detección de duplicados predeterminadas para cuentas, contactos y clientes potenciales, pero no para otros tipos de registros. Si desea que el sistema detecte duplicados para otros tipos de registro, deberá crear una nueva regla.

- Para obtener información sobre cómo crear una regla de detección de duplicados usando la interfaz de usuario, consulte [Configurar reglas de detección de duplicados para mantener limpios los datos](#).
- Para obtener información sobre cómo crear una regla de detección de duplicados usando código, consulte [Tablas de reglas de duplicados](#)

Ejemplo: Detectar duplicados durante la operación de crear mediante la API web

El ejemplo siguiente muestra cómo detectar duplicados durante las operaciones `Create` y `Update` mediante el encabezado `MSCRM.SuppressDuplicateDetection` de la solicitud de la API web.

Solicitud:

```
HTTP

POST [Organization URI]/org1/api/data/v9.0/leads HTTP/1.1
If-None-Match: null
OData-Version: 4.0
OData-MaxVersion: 4.0
Content-Type: application/json
Accept: application/json
MSCRM.SuppressDuplicateDetection: false

{

    "firstname": "Monte",
    "lastname": "Orton",
    "emailaddress1": "monteorton@example.com"
}
```

Si ya existe un registro de cliente potencial con el mismo atributo `emailaddress1`, se devuelve la siguiente respuesta.

Respuesta:

```
HTTP

HTTP/1.1 500 Internal Server Error
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0

{
    "error": {
        "code": "0x80040333",
        "message": "A record was not created or updated because a duplicate of the current record already exists."
    }
}
```

Asigne el valor `true` al encabezado `MSCRM.SuppressDuplicateDetection` para permitir la creación de un registro duplicado.

Detectar duplicados durante las operaciones de actualización

Establezca el valor del encabezado `MSCRM.SuppressDuplicateDetection` como `false` en la solicitud `PATCH` para evitar la creación de un registro duplicado durante la operación de actualizar. De forma predeterminada, se suprime la detección de duplicados cuando se actualizan registros mediante la API web.

Ejemplo: Detectar duplicados durante la operación de actualización mediante la API web

El ejemplo que aparece a continuación intenta actualizar un registro de entidad cliente potencial existente que incluye el mismo valor del atributo `emailaddress1` que un registro existente.

Solicitud:

HTTP

```
PATCH [Organization URI]/api/data/v9.0/leads(c4567bb6-47a3-e711-811b-e0071b6ac1b1) HTTP/1.1
If-None-Match: null
OData-Version: 4.0
OData-MaxVersion: 4.0
Content-Type: application/json
Accept: application/json
MSCRM.SuppressDuplicateDetection: false
If-Match: *

{
    "firstname": "Monte",
    "lastname": "Orton",
    "emailaddress1": "monteorton@example.com"
}
```

Respuesta:

HTTP

```
HTTP/1.1 500 Internal Server Error
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0

{
    "error": {
        "code": "0x80040333",
```

```
        "message": "A record was not created or updated because a duplicate  
of the current record already exists."  
    }  
}
```

Consulte también

[Detección de datos duplicados con SDK para .NET](#)

Nota

¿Puede indicarnos sus preferencias de idioma de documentación? **Realice una breve encuesta.**  (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales (**declaración de privacidad** ).

Solución de problemas de errores de cliente de Dataverse Web API

Artículo • 25/10/2023

En este artículo se describen los errores comunes de cliente que puede encontrar al usar la [API web de Dataverse](#) y cómo puede evitarlos.

Recurso no encontrado para el segmento

Síntomas

Solicitud

HTTP

```
GET [Organization URI]/api/data/v9.2/Account HTTP/1.1
```

Respuesta

HTTP

HTTP/1.1 404 Not Found

```
{
  "error": {
    "code": "0x8006088a",
    "message": "Resource not found for the segment 'Account'."}
```

Causa

Este error se produce cuando se usa el nombre incorrecto para un recurso. Ese recurso puede ser el nombre de un conjunto de entidades, una función o una acción. Estos nombres de recursos distinguen mayúsculas de minúsculas.

En el ejemplo anterior, hay un conjunto de entidades denominado `accounts`, pero no uno denominado `Account`.

Cómo evitar

- Si el recurso es un tipo de entidad, consulte el [documento Servicio](#) de API web, que proporcionará una lista de todos los nombres de conjuntos de entidades conocidos.
- Si el recurso es una función o una acción, compruebe que el nombre que usa existe en el [documento de \\$metadata CSDL](#).

No se encontró una propiedad denominada '{nombre de propiedad}' en el tipo 'Microsoft.Dynamics.CRM. {nombre de entidad}'

Síntomas

Solicitud

HTTP

```
GET [Organization URI]/api/data/v9.2/accounts?$select=Name HTTP/1.1
```

Respuesta

HTTP

HTTP/1.1 400 Bad Request

```
{
  "error": {
    "code": "0x0",
    "message": "Could not find a property named 'Name' on type
'Microsoft.Dynamics.CRM.account'."
  }
}
```

Causa

Este error se produce cuando se usa el nombre incorrecto de una propiedad. Los nombres de las propiedades distinguen mayúsculas de minúsculas.

En el ejemplo anterior, hay una propiedad denominada `name`, pero no una denominada `Name`.

Cómo evitar

Compruebe que el nombre que usa existe en el [documento de \\$metadata CSDL](#). Para obtener más información, consulte [Propiedades de la API web](#).

No se encontró ningún recurso HTTP que coincida con el URI de solicitud.

Síntomas

Solicitud

HTTP

```
POST [Organization URI]/api/data/v9.2/WhoAmI
```

Respuesta

HTTP

HTTP/1.1 404 Not Found

```
{
  "error": {
    "code": "",
    "message": "No HTTP resource was found that matches the request URI
'[Organization URI]/api/data/v9.2/WhoAmI'."}
```

Causa

Este error se produce cuando el método HTTP incorrecto se aplica a una función o acción. En este caso, [la función WhoAmI](#) requiere el uso de `GET` pero `POST` se usó.

Cómo evitar

Tenga en cuenta qué tipo de operación de OData está usando y el método HTTP correcto para usar. Para obtener más información, consulte:

- [Acciones de API web](#)
- [Uso de acciones de API web](#)

- Funciones de API web
- Uso de funciones de API web

Se encontró la propiedad {property name} no válida en la entidad 'Microsoft.Dynamics.CRM.{nombre de entidad}'

Síntomas

Solicitud

```
HTTP  
  
POST [Organization URI]/api/data/v9.2/accounts HTTP/1.1  
  
{  
    "Name": "Account name"  
}
```

Respuesta

```
HTTP  
  
HTTP/1.1 400 Bad Request  
  
{  
    "error": {  
        "code": "0x0",  
        "message": "An error occurred while validating input parameters:  
            Microsoft.Crm.CrmException: Invalid property 'Name' was found in entity  
            'Microsoft.Dynamics.CRM.account'.  
                ---> Microsoft.OData.ODataException: Does not support untyped value in  
                non-open type.<truncated for brevity>"  
    }  
}
```

Causa

Este error se produce porque se usó el nombre incorrecto de una propiedad. Los nombres de propiedad distinguen mayúsculas de minúsculas y `Name` se usaron en lugar de `name`.

Cómo evitar

Compruebe que el nombre de propiedad que usa existe en el [documento de \\$metadata CSDL](#). Para obtener más información, consulte [Propiedades de la API web](#).

Error identificado en la carga proporcionada por el usuario para la entidad '{nombre de entidad}'

Hay dos problemas independientes que pueden producirse con este error. La diferencia está en InnerException.

- [InnerException : Microsoft.OData.ODataException](#): se encontró una propiedad no declarada [...] en la carga.
- [InnerException : System.ArgumentException](#): Stream no era legible.

InnerException : Microsoft.OData.ODataException: se encontró una propiedad no declarada [...] en la carga.

Este error se produce cuando se envía un nombre de propiedad de navegación no válido con una solicitud.

Síntomas

Solicitud

HTTP

```
POST [Organization URI]/api/data/v9.0/contacts HTTP/1.1

{
    "firstname": "test",
    "lastname": "contact",
    "parentcustomerid@odata.bind": "accounts(a779956b-d748-ed11-bb44-6045bd01152a)"
}
```

Respuesta

HTTP

HTTP/1.1 400 Bad Request

```
{  
    "error": {  
        "code": "0x80048d19",  
        "message": "Error identified in Payload provided by the user for  
Entity :'contacts' ---->  
        InnerException : Microsoft.OData.ODataException: An undeclared  
property 'parentcustomerid'  
            which only has property annotations in the payload but no property  
value was found in the payload.  
            In OData, only declared navigation properties and declared named  
streams can be represented as  
            properties without values <truncated for brevity>."  
    }  
}
```

Causa

Este error se produce porque no hay ninguna propiedad de navegación de un solo valor en el tipo de entidad de contacto denominado `parentcustomerid`. Para obtener más información, vea [Póngase en contacto con las propiedades de navegación de un solo valor de EntityType](#).

`parentcustomerid` es el nombre lógico de una columna de búsqueda en la tabla de contactos. Todas las búsquedas se representan mediante propiedades de navegación de un solo valor en OData. Los nombres de las propiedades de búsqueda no siempre coinciden con el nombre de propiedad de navegación de un solo valor correspondiente.

En este caso, la `parentcustomerid` columna es un tipo de búsqueda de cliente, un tipo de [búsqueda de varias tablas](#) que podría vincularse a la cuenta o a las tablas de contactos. Para admitir esta búsqueda de clientes, hay dos relaciones independientes y cada una tiene una propiedad de navegación de un solo valor diferente. La propiedad de navegación de un solo valor correcta en este caso es `parentcustomerid_account`.

Cómo evitar

Compruebe que el nombre de la propiedad de navegación que usa existe en el [documento de \\$metadata CSDL](#). Para obtener más información, consulte [Propiedades de navegación de la API web](#), especialmente la sección [Búsquedas de varias tablas](#).

InnerException : System.ArgumentException: Stream no se puede leer

Este error se produce al [ejecutar operaciones por lotes](#).

Síntomas

Recibe el siguiente error al enviar una `$batch` solicitud.

Respuesta

HTTP

HTTP/1.1 400 Bad Request

```
--batchresponse_5bd81edb-2ef9-4269-85c3-4623981e6c6e
Content-Type: application/http
Content-Transfer-Encoding: binary
```

HTTP/1.1 400 Bad Request

```
REQ_ID: 4c8c75eb-10bf-47f9-9998-c119146d511f
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0
```

```
{"error": {"code": "0x80048d19", "message": "Error identified in Payload
provided by the user for Entity : 'accounts',
For more information on this error please follow this help link
https://go.microsoft.com/fwlink/?linkid=2195293
----> InnerException : System.ArgumentException: Stream was not
readable.\r\n
at System.IO.StreamReader..ctor(Stream stream, Encoding encoding, Boolean
detectEncodingFromByteOrderMarks, Int32 bufferSize, Boolean leaveOpen)\r\n
at System.IO.StreamReader..ctor(Stream stream, Encoding encoding)\r\n
at
Microsoft.OData.JsonLight.ODataJsonLightInputContext.CreateTextReader(Stream
messageStream, Encoding encoding)\r\n
at
Microsoft.OData.JsonLight.ODataJsonLightInputContext..ctor(ODataMessageInfo
messageInfo, ODataMessageReaderSettings messageReaderSettings)\r\n
at Microsoft.OData.Json.ODataJsonFormat.CreateInputContext(ODataMessageInfo
messageInfo, ODataMessageReaderSettings messageReaderSettings)\r\n
at Microsoft.OData.ODataManager.ReadFromInput[T](Func`2 readFunc,
ODataPayloadKind[] payloadKinds)\r\n
at
System.Web.OData.Formatter.Deserialization.ODataResourceDeserializer.Read(OD
ataMessageReader messageReader, Type type, ODataDeserializerContext
readContext)\r\n
at System.Web.OData.Formatter.ODataMediaTypeFormatter.ReadFromStream(Type
type, Stream readStream, HttpContent content, IFormatterLogger
formatterLogger)."}}

--batchresponse_5bd81edb-2ef9-4269-85c3-4623981e6c6e--
```

Causa

Este error de deserialización se debe al uso de finales de línea distintos de CRLF en el cuerpo de la solicitud por lotes.

Dependiendo del editor que use, estos caracteres que no son de impresión pueden ser difíciles de ver. Si usa [Notepad++](#), puede usar la opción **Mostrar todos los caracteres** para que estos caracteres sean visibles.

Esta carga funcionará:

```
--batch_80dd1615-2a10-428a-bb6f-0e559792721f
Content-Type: application/http
Content-Transfer-Encoding: binary
POST /api/data/v9.1/accounts HTTP/1.1
Content-Type: application/json; type=entry
{
  "name": "Sample Account"
}
--batch_80dd1615-2a10-428a-bb6f-0e559792721f--
```

Se producirá un error en esta carga porque la última línea no termina con CRLF.

```
--batch_80dd1615-2a10-428a-bb6f-0e559792721f
Content-Type: application/http
Content-Transfer-Encoding: binary
POST /api/data/v9.1/accounts HTTP/1.1
Content-Type: application/json; type=entry
{
  "name": "Sample Account"
}
--batch_80dd1615-2a10-428a-bb6f-0e559792721f--|
```

En este caso, basta con agregar un retorno de carro al final de la última línea para que tenga éxito.

Cómo evitar

Asegúrese de que todos los finales de línea del cuerpo de la \$batch solicitud sean CRLF. Si no puede usar CRLF, agregue dos finales que no sean de línea al final del cuerpo de la solicitud por lotes para resolver este error de deserialización. Para obtener más información, vea [Solicitudes de Batch](#).

Vea también

- [Operaciones y tipos de API web](#)
- [Redactar solicitudes HTTP y controlar errores](#)

Comentarios

¿Le ha resultado útil esta página?

 Sí

 No

[Proporcionar comentarios sobre el producto ↗](#)

Utilizar la API web con definiciones de tabla

Artículo • 05/10/2023

Puede realizar cualquier operación de definición de tablas y columnas (metadatos) con la API web que puede realizar mediante el SDK para .NET. Esta sección proporciona instrucciones sobre cómo usar la API web con los tipos de entidad incluidos en [Web API Metadata Entity Type Reference](#).

Hay cuatro rutas de conjuntos de entidades expuestas para realizar operaciones con entidades de definición como se describe en la siguiente tabla.

Ruta del conjunto de entidades	Descripción
<i>[Organization URI]</i> /api/data/v9.2/EntityDefinitions	Contiene una colección EntityType
<i>[Organization URI]</i> /api/data/v9.2/RelationshipDefinitions	Contiene ManyToManyRelationshipMetadata , EntityType y OneToManyRelationshipMetadata , EntityType ya que ambas heredan de RelationshipMetadataBase EntityType .
<i>[Organization URI]</i> /api/data/v9.2/GlobalOptionSetDefinitions	Contiene una colección de valores BooleanOptionSetMetadata EntityType y OptionSetMetadata EntityType definidos globalmente, ya que ambos heredan de OptionSetMetadata EntityType .
<i>[Organization URI]</i> /api/data/v9.2/ManagedPropertyDefinitions	Sólo para uso interno.

Cada tipo de entidad de definición utiliza [MetadataId](#) como la propiedad de identificador único, que hereda de [MetadataBase](#) [EntityType](#). Si bien todas las entidades de definición tienen un [MetadataId](#), no puede consultarlos todos directamente. Por ejemplo, puede consultar y realizar operaciones en atributos (columnas de tabla) solo en el contexto de la [EntityMetadata](#) que los contiene.

Estas entidades de definición tienen algunas diferencias sustanciales con las tablas que almacenan datos comerciales y de aplicaciones, por ejemplo:

- Las propiedades de las entidades de definición utilizan muchos de los tipos complejos y de enumeración definidos en [Web API Complex Type Reference](#) y [Web API Enum Type Reference](#) en lugar de los tipos de datos primitivos utilizados en entidades que heredan de [crmbaseentity](#) [EntityType](#).

- Las entidades de definición siguen una convención de nomenclatura diferente y mantienen el estilo de nomenclatura de Pascal Case que se usa en los ensamblados del SDK para .NET.
- Las entidades de definición hacen un uso más extenso de la herencia, lo que implica que tiene hacer conversiones para recuperar los datos que desea.

En esta sección

[Consultar definiciones de tabla con la API web](#)

Puede utilizar la API web para consultar definiciones de tablas o columnas mediante un estilo de consulta RESTful.

[Recuperar definiciones de tablas por nombre o Metadatald](#)

Sus aplicaciones pueden adaptarse a los cambios de configuración consultando las definiciones de tabla y de columna. Cuando conoce una de las propiedades clave de un elemento de definición, puede recuperar definiciones utilizando la API web.

[Crear y actualizar definiciones de tabla mediante la API web](#)

Puede crear y actualizar tablas y columnas utilizando la API web para lograr los mismos resultados que obtiene con el SDK para .NET [CreateEntityRequest](#) y [UpdateEntityRequest](#).

[Crear y actualizar definiciones de columna mediante la API web](#)

Puede crear y actualizar columnas utilizando la API web para lograr los mismos resultados que obtiene con el SDK para .NET [CreateAttributeRequest](#) y [UpdateAttributeRequest](#).

[Crear y actualizar relaciones de tabla mediante la API web](#)

Puede verificar si las tablas son elegibles para participar en una relación con otras tablas y luego crear o actualizar esas relaciones usando la API web.

[Búsquedas en varias tablas](#)

Las columnas de tipo de búsqueda de varias tablas permiten al usuario utilizar una tabla específica que tiene varias tablas de relaciones uno a varios (1:N) en otras tablas del entorno. Una sola columna de tipo de búsqueda puede hacer referencia a otras tablas. Un valor de búsqueda enviado a la columna de tipo de tabla múltiple se comparará con un registro en cualquiera de las tablas relacionadas. Las búsquedas de tablas múltiples se pueden crear con tablas locales y tablas virtuales como tablas referenciadas.

[Crear y actualizar selecciones \(conjuntos de opciones\) mediante la API web](#)

Explica cómo trabajar con columnas de opciones globales y locales. Cómo recuperar valores de opciones, agregar, actualizar, eliminar y reordenar opciones.

Consulte también

[Examinar las definiciones de tablas para su entorno](#)

[Use la API web Microsoft Dataverse Ejemplo de operaciones de metadatos de la API Web](#)

[Ejemplo de operaciones de metadatos de API web \(C#\)](#)

Nota

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta.[↗](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad ↗](#)).

Consultar definiciones de tabla con la API web

Artículo • 15/03/2023

Dado que Microsoft Dataverse es una aplicación basada en metadatos, los desarrolladores pueden necesitar consultar las definiciones del sistema en tiempo de ejecución para adaptarse a cómo se ha configurado una organización. Esta funcionalidad usa un estilo de RESTful de la consulta.

ⓘ Nota

También puede generar una consulta con un estilo basado en objetos usando **EntityQueryExpression ComplexType** con **RetrieveMetadataChanges Function**. Esta función permite capturar cambios en las definiciones de la tabla entre dos períodos de tiempo, así como devolver un conjunto limitado de definiciones descritas por una consulta que especifique. Más información: [Definiciones de esquema de consulta](#)

Consultar el tipo de entidad EntityMetadata

Usará las mismas técnicas descritas en [Consultar datos utilizando la API web](#) cuando consulte EntityMetadata, con algunas variaciones. Use la ruta del conjunto de entidades `EntityDefinitions` para recuperar información acerca del `EntityType`. Las entidades de EntityMetadata contienen muchos datos por lo que deberá tener cuidado de recuperar solo los datos necesarios. El siguiente ejemplo muestra los datos devueltos solo para las propiedades `DisplayName`, `IsKnowledgeManagementEnabled` y `EntitySetName` de la definición para la entidad `Account`. El valor de la propiedad `MetadataId` siempre se devuelve.

Solicitud

HTTP

```
GET [Organization  
URI]/api/data/v9.0/EntityDefinitions(LogicalName='account')?  
$select=DisplayName,IsKnowledgeManagementEnabled,EntitySetName HTTP/1.1  
Accept: application/json  
OData-MaxVersion: 4.0  
OData-Version: 4.0
```

Respuesta

HTTP

```
HTTP/1.1 200 OK
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0

{
  "@odata.context": "[Organization
URI]/api/data/v9.0/$metadata#EntityDefinitions(DisplayName,IsKnowledgeManagementEnabled,EntitySetName)",
  "value": [
    {
      "DisplayName": {
        "LocalizedLabels": [
          {
            "Label": "Account",
            "LanguageCode": 1033,
            "IsManaged": true,
            "MetadataId": "2a4901bf-2241-db11-898a-0007e9e17ebd",
            "HasChanged": null
          }
        ],
        "UserLocalizedLabel": {
          "Label": "Account",
          "LanguageCode": 1033,
          "IsManaged": true,
          "MetadataId": "2a4901bf-2241-db11-898a-0007e9e17ebd",
          "HasChanged": null
        }
      },
      "IsKnowledgeManagementEnabled": false,
      "EntitySetName": "accounts",
      "MetadataId": "70816501-edb9-4740-a16c-6a5efbc05d84"
    }
  ]
}
```

Puede usar cualquiera de las propiedades `EntityMetadata` con opciones de consulta del sistema `$select` y puede usar `$filter` en cualquier propiedad que use valores primitivos o de enumeración.

No tiene hay límites en el número al número de entidades de metadatos que serán devueltas en una consulta. No hay paginación. Se devolverán todos los recursos que coincidan en la primera respuesta.

Límite de idiomas devueltos

Cuando un entorno tiene muchos idiomas aprovisionados, la cantidad de datos devueltos puede crecer mucho. Para obtener el mejor rendimiento, limite las etiquetas de idioma devueltas mediante el parámetro `LabelLanguages` con el valor LCID del idioma que desea devolver.

Los códigos de idioma son identificadores de configuración regional de cuatro o cinco dígitos. Los valores de identificadores de configuración regional válidos pueden encontrarse en el [gráfico de identificadores de configuración regional \(LCID\)](#).

Por ejemplo, agregar lo siguiente limitará las etiquetas de idioma localizadas al inglés: `&LabelLanguages=1033`.

Use tipos de enumeración en operaciones \$filter

Cuando necesite filtrar entidades de metadatos basadas en el valor de una propiedad que use una enumeración, debe incluir el espacio de nombres de la enumeración delante del valor de cadena. Los tipos de enumeración se utilizan como valores de propiedad solo en entidades de metadatos y tipos complejos. Por ejemplo, si necesita filtrar entidades en función de la propiedad `OwnershipType`, que usa [OwnershipTypes Enumtype](#), puede usar el siguiente `$filter` para devolver únicamente las entidades que son `UserOwned`.

HTTP

```
GET [Organization URI]/api/data/v9.0/EntityDefinitions?
$select=LogicalName&$filter=OwnershipType eq
'Microsoft.Dynamics.CRM.OwnershipTypes'UserOwned'
```

Use tipos complejos en operaciones \$filter

Cuando necesite filtrar entidades de metadatos basadas en el valor de una propiedad que use un tipo complejo, debe incluir la ruta al tipo primitivo subyacente. Los tipos complejos se utilizan como valores de propiedad solo en entidades de metadatos. Por ejemplo, si necesita filtrar entidades en función de la propiedad `CanCreateAttributes`, que usa el [BooleanManagedProperty ComplexType](#), puede usar el siguiente `$filter` para devolver únicamente las entidades que tengan un `Value` de `true`.

HTTP

```
GET [Organization URI]/api/data/v9.0/EntityDefinitions?  
$select=LogicalName&$filter=CanCreateAttributes/Value eq true
```

Este modelo funciona con [BooleanManagedProperty ComplexType](#) porque el valor primitivo a comprobar tiene un nivel de profundidad. Sin embargo, esto no funciona en las propiedades de [Label ComplexType](#).

Consultar atributos EntityMetadata

Puede ver atributos de entidad en el contexto de una entidad expandiendo la propiedad de navegación valorada como colección [Attributes](#), pero esto solo incluirá las propiedades comunes disponibles en el [AttributeMetadata EntityType](#) que todos los atributos comparten. Por ejemplo, la siguiente consulta devolverá el [LogicalName](#) de la entidad y todos los atributos expandidos que tengan un valor de [AttributeType](#) igual al valor de [AttributeTypeCode EnumType](#) de [Picklist](#).

HTTP

```
GET [Organization  
URI]/api/data/v9.0/EntityDefinitions(LogicalName='account')?  
$select=LogicalName&$expand=Attributes($select=LogicalName;$filter=Attribute  
Type eq Microsoft.Dynamics.CRM.AttributeTypeCode'Picklist')
```

Pero no puede incluir las propiedades navegación valoradas como colección [OptionSet](#) o [GlobalOptionSet](#) que los atributos [PicklistAttributeMetadata EntityType](#) tienen en el filtro [\\$select](#) de esta consulta.

Para recuperar las propiedades de un tipo específico de atributo debe convertir la propiedad de navegación valorada como colección [Attributes](#) al tipo que desee. La siguiente consulta devolverá solo los atributos [PicklistAttributeMetadata EntityType](#) e incluirá además los [LogicalName](#) y también expandirá las propiedades de navegación valoradas como colección de [OptionSet](#) y [GlobalOptionSet](#).

HTTP

```
GET [Organization  
URI]/api/data/v9.0/EntityDefinitions(LogicalName='account')/Attributes/Micro  
soft.Dynamics.CRM.PicklistAttributeMetadata?  
$select=LogicalName&$expand=OptionSet,GlobalOptionSet
```

ⓘ Nota

A pesar de que las propiedades de navegación valoradas como colección `OptionSet` y `GlobalOptionSet` se definen en `EnumAttributeMetadata EntityType`, no puede convertir los atributos a este tipo. Esto significa que si desea filtrar por otros tipos que también hereden estas propiedades (consulte [Tipos de entidades que heredan de `EnumAttributeMetadata`](#)), debe realizar consultas aparte para filtrar por cada tipo.

Otro ejemplo de esto es acceder a la propiedad `Precision` disponible en los atributos `MoneyAttributeMetadata EntityType` y `DecimalAttributeMetadata EntityType`. Para acceder a esta propiedad, debe convertir la colección de atributos como `MoneyAttributeMetadata EntityType` o `DecimalAttributeMetadata EntityType`. Un ejemplo que muestra la conversión a `MoneyAttributeMetadata` se muestra a continuación.

HTTP

```
GET [Organization  
URI]/api/data/v9.0/EntityDefinitions(LogicalName='account')/Attributes/Micro  
soft.Dynamics.CRM.MoneyAttributeMetadata?$select=LogicalName,Precision
```

Filtrar por nivel requerido

La propiedad `RequiredLevel` de `AttributeMetadata EntityType` utiliza un `AttributeRequiredLevelManagedProperty ComplexType` donde la propiedad `Value` es un `AttributeRequiredLevel EnumType`. En este caso, debe combinar patrones de [Utilizar tipos complejos en operaciones \\$filter](#) y [Utilizar tipos de enumeración en operaciones \\$filter](#) para filtrar por esta propiedad única. La siguiente consulta filtrará esos atributos en la entidad de cuenta que sean `ApplicationRequired`.

HTTP

```
GET [Organization  
URI]/api/data/v9.0/EntityDefinitions(LogicalName='account')/Attributes?  
$select=SchemaName&$filter=RequiredLevel/Value eq  
Microsoft.Dynamics.CRM.AttributeRequiredLevel'ApplicationRequired'
```

Recuperar atributos

Si conoce el `MetadataId` para `EntityMetadata` y `AttributeMetadata`, o el valor `LogicalName` de cualquiera de ellos, puede recuperar un atributo individual y acceder a los valores de las propiedades mediante una consulta como la siguiente. Esta consulta

recupera la propiedad `LogicalName` del atributo además de expandir la propiedad de navegación valorada como colección `OptionSet`. Debe convertir el atributo como `Microsoft.Dynamics.CRM.PicklistAttributeMetadata` para acceder a la propiedad de navegación valorada como colección `OptionSet`.

Solicitar

HTTP

```
GET [Organization
URI]/api/data/v9.0/EntityDefinitions(LogicalName='account')/Attributes(5967e
7cc-afbb-4c10-bf7e-
e7ef430c52be)/Microsoft.Dynamics.CRM.PicklistAttributeMetadata?
$select=LogicalName&$expand=OptionSet HTTP/1.1
Accept: application/json
Content-Type: application/json; charset=utf-8
OData-MaxVersion: 4.0
OData-Version: 4.0
```

Response

HTTP

```
HTTP/1.1 200 OK
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0

{
  "@odata.context": "[Organization
URI]/api/data/v9.0/$metadata#EntityDefinitions(70816501-edb9-4740-a16c-
6a5efbc05d84)/Attributes/Microsoft.Dynamics.CRM.PicklistAttributeMetadata(Lo
gicalName,OptionSet)/$entity",
  "LogicalName": "preferredappointmentdaycode",
  "MetadataId": "5967e7cc-afbb-4c10-bf7e-e7ef430c52be",
  "OptionSet@odata.context": "[Organization
URI]/api/data/v9.0/$metadata#EntityDefinitions(70816501-edb9-4740-a16c-
6a5efbc05d84)/Attributes(5967e7cc-afbb-4c10-bf7e-
e7ef430c52be)/Microsoft.Dynamics.CRM.PicklistAttributeMetadata/OptionSet/$en
tity",
  "OptionSet": {
    "Options": [
      {
        "Value": 0,
        "Label": {
          "LocalizedLabels": [
            {
              "Label": "Sunday",
              "LanguageCode": 1033,
              "IsManaged": true,
              "MetadataId": "21d6a218-2341-db11-898a-0007e9e17ebd",
              "HasChanged": null
            }
          ]
        }
      }
    ]
  }
}
```

```
        },
      ],
    "UserLocalizedLabel": {
      "Label": "Sunday",
      "LanguageCode": 1033,
      "IsManaged": true,
      "MetadataId": "21d6a218-2341-db11-898a-0007e9e17ebd",
      "HasChanged": null
    }
  },
  "Description": {
    "LocalizedLabels": [],
    "UserLocalizedLabel": null
  },
  "Color": null,
  "IsManaged": true,
  "MetadataId": null,
  "HasChanged": null
}
Additional options removed for brevity
],
"Description": {
  "LocalizedLabels": [
    {
      "Label": "Day of the week that the account prefers for scheduling service activities.",
      "LanguageCode": 1033,
      "IsManaged": true,
      "MetadataId": "1b67144d-ece0-4e83-a38b-b4d48e3f35d5",
      "HasChanged": null
    }
  ],
  "UserLocalizedLabel": {
    "Label": "Day of the week that the account prefers for scheduling service activities.",
    "LanguageCode": 1033,
    "IsManaged": true,
    "MetadataId": "1b67144d-ece0-4e83-a38b-b4d48e3f35d5",
    "HasChanged": null
  }
},
"DisplayName": {
  "LocalizedLabels": [
    {
      "Label": "Preferred Day",
      "LanguageCode": 1033,
      "IsManaged": true,
      "MetadataId": "ebb7e979-f9e3-40cd-a86d-50b479b1c5a4",
      "HasChanged": null
    }
  ],
  "UserLocalizedLabel": {
    "Label": "Preferred Day",
    "LanguageCode": 1033,
    "IsManaged": true,
```

```

        "MetadataId": "ebb7e979-f9e3-40cd-a86d-50b479b1c5a4",
        "HasChanged": null
    },
    "IsCustomOptionSet": false,
    "IsGlobal": false,
    "IsManaged": true,
    "IsCustomizable": {
        "Value": true,
        "CanBeChanged": false,
        "ManagedPropertyLogicalName": "iscustomizable"
    },
    "Name": "account_preferredappointmentdaycode",
    "OptionsetType": "Picklist",
    "IntroducedVersion": null,
    "MetadataId": "53f9933c-18a0-40a6-b4a5-b9610a101735",
    "HasChanged": null
}
}

```

Si no requiere ninguna propiedad del atributo y solo desea los valores de una propiedad de navegación valorada como colección como `OptionsSet`, puede incluir esto en la URL y limitar las propiedades con una opción de consulta del sistema `$select` para una consulta más eficaz. En el siguiente ejemplo solo la propiedad `Options` del `OptionSet` está incluida.

Solicitar

HTTP

```

GET [Organization
URI]/api/data/v9.0/EntityDefinitions(LogicalName='account')/Attributes(5967e
7cc-afbb-4c10-bf7e-
e7ef430c52be)/Microsoft.Dynamics.CRM.PicklistAttributeMetadata/OptionSet?
$select=Options HTTP/1.1
Accept: application/json
Content-Type: application/json; charset=utf-8
OData-MaxVersion: 4.0
OData-Version: 4.0

```

Response

HTTP

```

HTTP/1.1 200 OK
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0

{
"@odata.context": "[Organization

```

```
URI]/api/data/v9.0/$metadata#EntityDefinitions('account')/Attributes(5967e7c
c-afbb-4c10-bf7e-
e7ef430c52be)/Microsoft.Dynamics.CRM.PicklistAttributeMetadata/OptionSet(Opt
ions)[$entity",
"Options": [{
  "Value": 0,
  "Label": {
    "LocalizedLabels": [
      {
        "Label": "Sunday",
        "LanguageCode": 1033,
        "IsManaged": true,
        "MetadataId": "21d6a218-2341-db11-898a-0007e9e17ebd",
        "HasChanged": null
      }
    ],
    "UserLocalizedLabel": {
      "Label": "Sunday",
      "LanguageCode": 1033,
      "IsManaged": true,
      "MetadataId": "21d6a218-2341-db11-898a-0007e9e17ebd",
      "HasChanged": null
    }
  },
  "Description": {
    "LocalizedLabels": [],
    "UserLocalizedLabel": null
  },
  "Color": null,
  "IsManaged": true,
  "MetadataId": null,
  "HasChanged": null
}
],
Additional options removed for brevity
],
"MetadataId": "53f9933c-18a0-40a6-b4a5-b9610a101735"
}
```

Consultar metadatos de relaciones

Puede recuperar metadatos de relaciones en el contexto de una entidad determinada en gran parte de la misma forma que pueda consultar atributos. Las propiedades de navegación valorada como colección `ManyToManyRelationships`, `ManyToOneRelationships` y `OneToManyRelationships` se pueden consultar como la propiedad de navegación valorada como colección `Attributes`. Más información: [Consultar atributos EntityMetadata](#)

Sin embargo, las relaciones entre entidades también se pueden consultar con el conjunto de entidades `RelationshipDefinitions`. Puede usar una consulta como la siguiente para obtener la propiedad `SchemaName` para cada relación.

HTTP

```
GET [Organization URI]/api/data/v9.0/RelationshipDefinitions?  
$select=SchemaName
```

Las propiedades disponibles al consultar este conjunto de entidades se limitan a las que están en [RelationshipMetadataBase EntityType](#). Para tener acceso a propiedades de los tipos de entidad que se heredan de [RelationshipMetadataBase](#) es necesario incluir una conversión en la consulta como la siguiente para devolver solo [OneToManyRelationshipMetadata EntityType](#).

HTTP

```
GET [Organization  
URI]/api/data/v9.0/RelationshipDefinitions/Microsoft.Dynamics.CRM.OneToManyR  
elationshipMetadata?$select=SchemaName
```

Puesto que las entidades devueltas tienen el tipo [OneToManyRelationshipMetadata](#), puede filtrar por propiedades como [ReferencedEntity](#) para crear una consulta para devolver solo las relaciones entre entidades de una a varias para una entidad específica, como la entidad de cuenta como se muestra en la siguiente consulta:

HTTP

```
GET [Organization  
URI]/api/data/v9.0/RelationshipDefinitions/Microsoft.Dynamics.CRM.OneToManyR  
elationshipMetadata?$select=SchemaName&$filter=ReferencedEntity eq 'account'
```

Esa consulta esencialmente devolverá los mismos resultados que la siguiente consulta, que se filtra porque está incluida en propiedad de navegación valorada como colección de [EntityMetadataOneToManyRelationships](#) de la entidad de cuenta. La diferencia es que para la consulta anterior no necesita saber el [MetadataId](#) para la entidad de cuenta.

HTTP

```
GET [Organization  
URI]/api/data/v9.0/EntityDefinitions(LogicalName='account')/OneToManyRelatio  
nships?$select=SchemaName
```

Consultar conjuntos de opciones globales

Puede usar la ruta de conjuntos de entidades `GlobalOptionSetDefinitions` para recuperar información sobre conjuntos de opciones globales, pero esta ruta de acceso no admite el uso de la opción de consulta del sistema `$filter`. Por lo tanto, solo puede recuperar un solo conjunto de opciones global mediante el `MetadataId` o el nombre único.

Ejemplo: por MetadataId

El siguiente ejemplo muestra cómo recuperar un conjunto de opciones global usando el `MetadataId`.

HTTP

```
GET [Organization URI]/api/data/v9.0/GlobalOptionSetDefinitions(08fa2cb2-e3fe-497a-9b5d-ee887f5cc3cd)
```

Ejemplo: por nombre

El siguiente ejemplo muestra cómo recuperar un conjunto de opciones global por nombre:

HTTP

```
GET [Organization  
URI]/api/data/v9.0/GlobalOptionSetDefinitions(Name='incident_caseorigincode'  
)
```

También puede obtener acceso a la definición de un conjunto de opciones globales dentro de la propiedad de navegación valorada como colección `GlobalOptionSet` para cualquier atributo que la utilice. Esto está disponible para todos los [tipos derivados](#) `EnumAttributeMetadata` `EntityType`. Más información: [Recuperar atributos](#)

Consultar también

[Usar la API web con metadatos de Dataverse](#)

[Recuperar metadatos por nombre o identificador de metadatos](#)

[Entidades de metadatos y atributos con la API web](#)

[Relaciones de entidades de metadatos con la API web](#)

 Nota

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta. [↗](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales (**declaración de privacidad** [↗](#)).

Recuperar definiciones de tablas por nombre o MetadataId

Artículo • 25/08/2023

Sus aplicaciones pueden adaptarse a los cambios de configuración consultando las definiciones de tabla y de columna (metadatos). Cuando conoce una de las propiedades clave de un elemento de definición, puede recuperar definiciones utilizando la API web.

ⓘ Nota

Este tema describe cómo recuperar definiciones de tablas y otras entidades de esquema por nombre. Sin embargo, también puede usar los nombres al realizar otras operaciones para crear, actualizar y eliminar entidades de esquema.

Recuperar elementos de definición por nombre

Todos los elementos de definición recuperables tienen una clave principal `MetadataId` que se puede utilizar para recuperar elementos individuales. Para las definiciones que tienen un clave alternativa, puede recuperarlas por nombre.

Recuperar elementos de definición por nombre es más fácil porque probablemente ya tengan alguna referencia al nombre del elemento en su código. La siguiente tabla enumera las propiedades de clave alternativa para recuperar dichos elementos por nombre.

Elemento de la definición	Clave alternativa	Ejemplo
Entity	LogicalName	<code>GET /api/data/v9.2/EntityDefinitions(LogicalName='account')</code>
Atributo	LogicalName	<code>GET /api/data/v9.2/EntityDefinitions(LogicalName='account')/Attributes(LogicalName='emailaddress1')</code>
Relación	SchemaName	<code>GET /api/data/v9.2/RelationshipDefinitions(SchemaName='Account_Tasks')</code>
Conjunto de opciones globales	Nombre	<code>GET /api/data/v9.2/GlobalOptionSetDefinitions(Name='metric_goaltype')</code>

Ejemplo: Recuperar elementos de definición por nombre

Un elemento de definición común que las personas desean recuperar son las opciones configuradas para un atributo en particular. El siguiente ejemplo muestra cómo recuperar las propiedades de `OptionSet` y `GlobalOptionSet` de un [PicklistAttributeMetadata EntityType](#).

ⓘ Nota

Expandiendo ambas propiedades de navegación de un solo valor `OptionSet` y `GlobalOptionSet` de [PicklistAttributeMetadata EntityType](#) le permite obtener la definición de la opción si el atributo está

configurado para usar conjuntos de opciones globales o el conjunto de opciones 'local' dentro de la entidad. Si es un conjunto de opciones 'local', la propiedad `GlobalOptionSet` será NULL como se muestra a continuación.

Si el atributo usaba un conjunto de opciones global, la propiedad `GlobalOptionSet` contendría las opciones definidas y la propiedad `OptionSet` sería NULL.

Solicitud:

HTTP

```
GET [Organization
URI]/api/data/v9.2/EntityDefinitions(LogicalName='account')/Attributes(LogicalName='accountcategorycode')/Microsoft.Dynamics.CRM.PicklistAttributeMetadata?
$select=LogicalName&$expand=OptionSet($select=Options),GlobalOptionSet($select=Options) HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
Accept: application/json
Content-Type: application/json; charset=utf-8
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0

{
    "@odata.context": "[Organization
URI]/api/data/v9.2/$metadata#EntityDefinitions('account')/Attributes/Microsoft.Dynamics.CRM.PicklistAttributeMetadata(LogicalName,OptionSet,GlobalOptionSet,OptionSet(Options),GlobalOptionSet(Options))/$entity",
    "LogicalName": "accountcategorycode",
    "MetadataId": "118771ca-6fb9-4f60-8fd4-99b6124b63ad",
    "OptionSet@odata.context": "[Organization
URI]/api/data/v9.2/$metadata#EntityDefinitions('account')/Attributes(118771ca-6fb9-4f60-8fd4-99b6124b63ad)/Microsoft.Dynamics.CRM.PicklistAttributeMetadata/OptionSet(Options)/$entity",
    "OptionSet": {
        "Options": [
            {
                "Value": 1,
                "Label": {
                    "LocalizedLabels": [
                        {
                            "Label": "Preferred Customer",
                            "LanguageCode": 1033,
                            "IsManaged": true,
                            "MetadataId": "0bd8a218-2341-db11-898a-0007e9e17ebd",
                            "HasChanged": null
                        }
                    ],
                    "UserLocalizedLabel": {
                        "Label": "Preferred Customer",
                        "LanguageCode": 1033,
                        "IsManaged": true,
                        "MetadataId": "0bd8a218-2341-db11-898a-0007e9e17ebd",
                        "HasChanged": null
                    }
                }
            },
            "Description": {
                "LocalizedLabels": [
                ],
                "UserLocalizedLabel": null
            }
        ]
    }
}
```

```

        },
        "Color": null,
        "IsManaged": true,
        "MetadataId": null,
        "HasChanged": null
    }, {
        "Value": 2,
        "Label": {
            "LocalizedLabels": [
                {
                    "Label": "Standard",
                    "LanguageCode": 1033,
                    "IsManaged": true,
                    "MetadataId": "0dd8a218-2341-db11-898a-0007e9e17ebd",
                    "HasChanged": null
                }],
            "UserLocalizedLabel": {
                "Label": "Standard",
                "LanguageCode": 1033,
                "IsManaged": true,
                "MetadataId": "0dd8a218-2341-db11-898a-0007e9e17ebd",
                "HasChanged": null
            }
        },
        "Description": {
            "LocalizedLabels": [
                ],
            "UserLocalizedLabel": null
        },
        "Color": null,
        "IsManaged": true,
        "MetadataId": null,
        "HasChanged": null
    }],
    "MetadataId": "b994cdd8-5ce9-4ab9-bdd3-8888ebdb0407"
},
"GlobalOptionSet": null
}

```

Recuperar elementos de definición por MetadataId

Dado que `MetadataId` es la clave principal para los elementos de definición, la recuperación de elementos individuales sigue el mismo patrón utilizado para recuperar tablas de datos profesionales.

Elemento de la definición	Ejemplo
Entity	<code>GET /api/data/v9.2/EntityDefinitions(<Entity MetadataId>)</code>
Atributo	<code>GET /api/data/v9.2/EntityDefinitions(<Entity MetadataId>)/Attributes(<Attribute MetadataId>)</code>
Relación	<code>GET /api/data/v9.2/RelationshipDefinitions(<Relationship MetadataId>)</code>
Conjunto de opciones globales	<code>GET /api/data/v9.2/GlobalOptionSetDefinitions(<OptionSet MetadataId>)</code>

Ejemplo: Recuperar elementos de definición por MetadataId

Para lograr el mismo resultado que se muestra en [Ejemplo: Recuperar elementos de definición por nombre](#), debe realizar una serie de operaciones de consulta para obtener `MetadataId` filtrando por la entidad

`LogicalName` y después por el atributo `LogicalName`.

Solicitud:

```
HTTP

GET [Organization URI]/api/data/v9.2/EntityDefinitions?
$filter=LogicalName%20eq%20'account'&$select=MetadataId HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
Accept: application/json
Content-Type: application/json; charset=utf-8
```

Respuesta:

```
HTTP

HTTP/1.1 200 OK
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0

{
  "@odata.context": "[Organization
URI]/api/data/v9.2/$metadata#EntityDefinitions(MetadataId)", "value": [
    {
      "MetadataId": "70816501-edb9-4740-a16c-6a5efbc05d84"
    }
  ]
}
```

Solicitud:

```
HTTP

GET [Organization URI]/api/data/v9.2/EntityDefinitions(70816501-edb9-4740-a16c-
6a5efbc05d84)/Attributes?$filter=LogicalName%20eq%20'accountcategorycode'&$select=MetadataId
HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
Accept: application/json
Content-Type: application/json; charset=utf-8
```

Respuesta:

```
HTTP

HTTP/1.1 200 OK
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0

{
  "@odata.context": "[Organization URI]/api/data/v9.2/$metadata#EntityDefinitions(70816501-
edb9-4740-a16c-6a5efbc05d84)/Attributes(MetadataId)", "value": [
    {
      "@odata.type": "#Microsoft.Dynamics.CRM.PicklistAttributeMetadata",
      "MetadataId": "118771ca-6fb9-4f60-8fd4-99b6124b63ad"
    }
  ]
}
```

Solicitud:

HTTP

```
GET [Organization URI]/api/data/v9.2/EntityDefinitions(70816501-edb9-4740-a16c-  
6a5efbc05d84)/Attributes(118771ca-6fb9-4f60-8fd4-  
99b6124b63ad)/Microsoft.Dynamics.CRM.PicklistAttributeMetadata?  
$select=LogicalName&$expand=OptionSet($select=Options),GlobalOptionSet($select=Options) HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
Accept: application/json  
Content-Type: application/json; charset=utf-8
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK  
Content-Type: application/json; odata.metadata=minimal  
OData-Version: 4.0  
  
{  
    "@odata.context": "[Organization URI]/api/data/v9.2/$metadata#EntityDefinitions(70816501-  
edb9-4740-a16c-  
6a5efbc05d84)/Attributes/Microsoft.Dynamics.CRM.PicklistAttributeMetadata(LogicalName,OptionSet,  
GlobalOptionSet,OptionSet(Options),GlobalOptionSet(Options))/$entity",  
    "LogicalName": "accountcategorycode",  
    "MetadataId": "118771ca-6fb9-4f60-8fd4-99b6124b63ad",  
    "OptionSet@odata.context": "[Organization  
URI]/api/data/v9.2/$metadata#EntityDefinitions(70816501-edb9-4740-a16c-  
6a5efbc05d84)/Attributes(118771ca-6fb9-4f60-8fd4-  
99b6124b63ad)/Microsoft.Dynamics.CRM.PicklistAttributeMetadata/OptionSet(Options)/$entity",  
    "OptionSet": {  
        "Options": [{  
            "Value": 1,  
            "Label": {  
                "LocalizedLabels": [  
                    {"Label": "Preferred Customer",  
                     "LanguageCode": 1033,  
                     "IsManaged": true,  
                     "MetadataId": "0bd8a218-2341-db11-898a-0007e9e17ebd",  
                     "HasChanged": null  
                ],  
                "UserLocalizedLabel": {  
                    "Label": "Preferred Customer",  
                     "LanguageCode": 1033,  
                     "IsManaged": true,  
                     "MetadataId": "0bd8a218-2341-db11-898a-0007e9e17ebd",  
                     "HasChanged": null  
                }  
            },  
            "Description": {  
                "LocalizedLabels": [  
                    ],  
                "UserLocalizedLabel": null  
            },  
            "Color": null,  
            "IsManaged": true,  
            "MetadataId": null,  
            "HasChanged": null  
        }, {  
            "Value": 2,  
            "Label": {  
                "LocalizedLabels": [  
                    ],  
                "UserLocalizedLabel": null  
            },  
            "Color": null,  
            "IsManaged": true,  
            "MetadataId": null,  
            "HasChanged": null  
        }  
    }  
}
```

```
"LocalizedLabels": [
    {
        "Label": "Standard",
        "LanguageCode": 1033,
        "IsManaged": true,
        "MetadataId": "0dd8a218-2341-db11-898a-0007e9e17ebd",
        "HasChanged": null
    },
    "UserLocalizedLabel": {
        "Label": "Standard",
        "LanguageCode": 1033,
        "IsManaged": true,
        "MetadataId": "0dd8a218-2341-db11-898a-0007e9e17ebd",
        "HasChanged": null
    }
],
"Description": {
    "LocalizedLabels": [
        ],
        "UserLocalizedLabel": null
    },
    "Color": null,
    "IsManaged": true,
    "MetadataId": null,
    "HasChanged": null
}],
"MetadataId": "b994cdd8-5ce9-4ab9-bdd3-8888ebdb0407"
},
"GlobalOptionSet": null
}
```

Consulte también

[Utilizar la API web con definiciones de tabla](#)

[Consultar definiciones de tabla con la API web](#)

[Crear y actualizar definiciones de tabla mediante la API web](#)

[Crear y actualizar relaciones de tabla mediante la API web](#)

[Ejemplo de operaciones de metadatos de API web](#)

[Ejemplo de operaciones de metadatos de API web \(C#\)](#)

ⓘ Nota

¿Puede indicarnos sus preferencias de idioma de documentación? [Realice una breve encuesta.](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad](#)).

Crear y actualizar definiciones de tabla mediante la API web

Artículo • 29/09/2023

Puede realizar las mismas operaciones en las definiciones de tabla utilizando la API web que con el SDK para .NET. Este artículo se centra en el trabajo con definiciones de tablas (metadatos) utilizando la API web. Para encontrar detalles sobre las propiedades de definición de la tabla, consulte [Personalizar las definiciones de las tablas](#) y [EntityMetadata EntityType](#).

💡 Sugerencia

Las entidades, los atributos y los conjuntos de opciones globales (también conocidos como tablas, columnas y opciones) son todos componentes de la solución. Cuando se crean puede asociarlos a una solución utilizando el encabezado de solicitud `MSCRM.SolutionUniqueName` y estableciendo un valor en el nombre único de la solución de la que debe formar parte.

Crear definiciones de

Para crear una definición de tabla, `POST` la representación JSON de los datos de definición de entidad en la ruta establecida de la entidad `EntityDefinitions`. La entidad debe incluir la definición para el atributo de nombre principal. No necesita establecer valores para todas las propiedades. Se requieren los elementos de la lista excepto `Description`, aunque se recomienda establecer una descripción. Los valores de propiedad que no especifica se establecen como valores predeterminados. Para comprender los valores predeterminados, consulte el ejemplo en la sección [Actualizar definiciones de tabla](#). El ejemplo en este artículo usa las siguientes propiedades de la entidad.

Propiedad	valor
<code>EntityMetadata</code>	
<code>SchemaName</code>	<code>new_BankAccount</code> Nota: Debe incluir el prefijo de personalización que coincide con el editor de soluciones. Aquí se usa el valor predeterminado " new_ ", pero debe elegir el prefijo que funcione para la solución.
<code>DisplayName</code>	Cuenta bancaria

Propiedad	valor
EntityMetadata	
DisplayCollectionName	Cuentas bancarias
Description	Una entidad para almacenar información acerca de las cuentas bancarias del cliente.
OwnershipType	UserOwned Nota: para los valores que puede establecer aquí, consulte OwnershipTypes EnumType .
IsActivity	falso
HasActivities	falso
HasNotes	falso

Además de las propiedades enumeradas anteriormente, la propiedad **EntityMetadataAttributes** debe contener una matriz que incluya un [StringAttributeMetadata EntityType](#) para representar el atributo de nombre principal para la entidad. El atributo propiedad **IsPrimaryName** debe ser verdadero. En la siguiente tabla se describen las distintas propiedades establecidas en el ejemplo.

Propiedad	valor
Atributo principal	
SchemaName	new_AccountName
RequiredLevel	None Nota: Para conocer los valores que puede establecer aquí, consulte AttributeRequiredLevelManagedProperty ComplexType y AttributeRequiredLevel EnumType .
MaxLength	100
FormatName	Text Nota: el atributo de nombre principal debe usar formato de texto. Para las opciones de formato disponibles para otros atributos de cadena, consulte Formatos de cadena .
DisplayName	Nombre de cuenta
Description	Escriba el nombre de la cuenta bancaria.
IsPrimaryName	true

ⓘ Nota

Cuando se crean o se actualizan etiquetas con **Label ComplexType**, solo debe establecer la propiedad `LocalizedLabels`. El valor `UserLocalizedLabel` devuelto se basa en la preferencia de idioma del usuario y es de solo lectura.

El siguiente ejemplo muestra la creación de una tabla personalizada con las propiedades establecidas. El idioma es inglés con el Id. de configuración regional (LCID) de 1033. Los valores de identificadores de configuración regional válidos pueden encontrarse en el [gráfico de identificadores de configuración regional \(LCID\)](#).

Solicitud:

HTTP

```
POST [Organization URI]/api/data/v9.2/EntityDefinitions HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
Accept: application/json
Content-Type: application/json; charset=utf-8
OData-MaxVersion: 4.0
OData-Version: 4.0

{
    "@odata.type": "Microsoft.Dynamics.CRM.EntityMetadata",
    "Attributes": [
        {
            "AttributeType": "String",
            "AttributeName": {
                "Value": "StringType"
            },
            "Description": {
                "@odata.type": "Microsoft.Dynamics.CRM.Label",
                "LocalizedLabels": [
                    {
                        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                        "Label": "Type the name of the bank account",
                        "LanguageCode": 1033
                    }
                ]
            },
            "DisplayName": {
                "@odata.type": "Microsoft.Dynamics.CRM.Label",
                "LocalizedLabels": [
                    {
                        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                        "Label": "Account Name",
                        "LanguageCode": 1033
                    }
                ]
            },
            "IsPrimaryName": true,
            "RequiredLevel": {
                "Value": "None",
            }
        }
    ]
}
```

```

    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
},
"SchemaName": "new_AccountName",
"@odata.type": "Microsoft.Dynamics.CRM.StringAttributeMetadata",
"FormatName": {
    "Value": "Text"
},
"MaxLength": 100
},
],
"Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
"LocalizedLabels": [
{
    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
    "Label": "An entity to store information about customer bank accounts",
    "LanguageCode": 1033
}
]
},
"DisplayCollectionName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
"LocalizedLabels": [
{
    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
    "Label": "Bank Accounts",
    "LanguageCode": 1033
}
]
},
"DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
"LocalizedLabels": [
{
    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
    "Label": "Bank Account",
    "LanguageCode": 1033
}
]
},
"HasActivities": false,
"HasNotes": false,
"IsActivity": false,
"OwnershipType": "UserOwned",
"SchemaName": "new_BankAccount"
}

```

Respuesta:

HTTP

```
HTTP/1.1 204 No Content
OData-Version: 4.0
OData-EntityId: [Organization URI]/api/data/v9.2/EntityDefinitions(417129e1-207c-e511-80d2-00155d2a68d2)
```

Actualizar definiciones de tabla

Importante

No puede utilizar el método `PATCH` para actualizar las entidades del modelo de datos. Las definiciones de la tabla tienen paridad con el SDK para .NET

[UpdateEntityRequest Class](#) que reemplaza la definición de entidad con la incluida.

Por lo tanto, debe utilizar el método `PUT` al actualizar las entidades del modelo de datos y tener cuidado de incluir todas las propiedades existentes que no tiene la intención de cambiar. No puede actualizar propiedades individuales.

Cuando actualice las definiciones de tabla con etiquetas, debe incluir un encabezado de solicitud `MSCRM.MergeLabels` personalizado para controlar cómo se deben gestionar las etiquetas en la actualización. Si una etiqueta para un elemento tiene etiquetas para otros idiomas y lo actualiza con una etiqueta que contiene sólo una etiqueta para un idioma específico, el encabezado `MSCRM.MergeLabels` controla si sobrescribe las etiquetas existentes o combina la nueva etiqueta con etiquetas de idioma existentes. Con `MSCRM.MergeLabels` establecido como `true`, las etiquetas nuevas definidas solo sobrescribirán las etiquetas existentes cuando el código de idioma coincide. Si desea sobrescribir las etiquetas existentes para incluir solo las etiquetas que incluye, establezca `MSCRM.MergeLabels` como `false`.

Importante

Si no incluye un encabezado `MSCRM.MergeLabels`, el comportamiento predeterminado es como si el valor fuera `false` y las etiquetas localizadas no incluidas en la actualización se perderán.

Cuando actualiza la definición de una tabla o columna, debe utilizar [PublishXml Action](#) o [PublishAllXml Action](#) antes de que los cambios que realice se apliquen a la aplicación. Más información: [Publicación de personalizaciones](#)

Normalmente, recupera la definición JSON del atributo de entidad y modificará las propiedades antes de devolverlo. El siguiente ejemplo contiene todas las propiedades

de definición de la tabla creada en el ejemplo [Crear definiciones de tabla](#), pero se ha cambiado `DisplayName` a "Bank Business Name". Puede ser útil tener en cuenta que JSON aquí proporciona los valores predeterminados para las propiedades que no se establecen en el ejemplo [Crear definiciones de tabla](#).

ⓘ Nota

Algunos de los ejemplos siguientes utilizan el valor de clave principal `MetadataId`. También puede usar la clave alternativa `LogicalName` para hacer referencia a entidades de esquema. Más información: [Recuperar definiciones de tablas por nombre o MetadataId](#)

Solicitud:

HTTP

```
PUT [Organization URI]/api/data/v9.2/EntityDefinitions(417129e1-207c-e511-80d2-00155d2a68d2) HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
Accept: application/json
Content-Type: application/json; charset=utf-8
OData-MaxVersion: 4.0
OData-Version: 4.0
MSCRM.MergeLabels: true

{
  "@odata.context": "[Organization
URI]/api/data/v9.2/$metadata#EntityDefinitions/$entity",
  "ActivityTypeMask": 0,
  "AutoRouteToOwnerQueue": false,
  "CanTriggerWorkflow": true,
  "Description": {
    "LocalizedLabels": [
      {
        "Label": "An entity to store information about customer bank accounts",
        "LanguageCode": 1033,
        "IsManaged": false,
        "MetadataId": "edc3abd7-c5ae-4822-a3ed-51734fdd0469",
        "HasChanged": null
      }
    ]
  },
  "DisplayCollectionName": {
    "LocalizedLabels": [
      {
        "Label": "Bank Accounts",
        "LanguageCode": 1033,
        "IsManaged": false,
        "MetadataId": "7c758e0c-e9cf-4947-93b0-50ec30b20f60",
        "HasChanged": null
      }
    ]
  }
}
```

```
        "HasChanged": null
    }
]
},
"DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Bank Business Name",
            "LanguageCode": 1033
        }
    ]
},
"EntityHelpUrlEnabled": false,
"EntityHelpUrl": null,
"IsDocumentManagementEnabled": false,
"IsOneNoteIntegrationEnabled": false,
"IsInteractionCentricEnabled": false,
"IsKnowledgeManagementEnabled": false,
"AutoCreateAccessTeams": false,
"IsActivity": false,
"IsActivityParty": false,
"IsAuditEnabled": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyauditsettings"
},
"IsAvailableOffline": false,
"IsChildEntity": false,
"IsAIRUpdated": false,
"IsValidForQueue": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyqueuesettings"
},
"IsConnectionsEnabled": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyconnectionsettings"
},
"IconLargeName": null,
"IconMediumName": null,
"IconSmallName": null,
"IsCustomEntity": true,
"IsBusinessProcessEnabled": false,
"IsCustomizable": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "iscustomizable"
},
"IsRenameable": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "isrenameable"
}
```

```
},
  "IsMappable": {
    "Value": true,
    "CanBeChanged": false,
    "ManagedPropertyLogicalName": "ismappable"
  },
  "IsDuplicateDetectionEnabled": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyduplicatedetectionsettings"
  },
  "CanCreateAttributes": {
    "Value": true,
    "CanBeChanged": false,
    "ManagedPropertyLogicalName": "cancreateattributes"
  },
  "CanCreateForms": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "cancreateforms"
  },
  "CanCreateViews": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "cancreateviews"
  },
  "CanCreateCharts": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "cancreatecharts"
  },
  "CanBeRelatedEntityInRelationship": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canberelatedentityinrelationship"
  },
  "CanBePrimaryEntityInRelationship": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canbeprimaryentityinrelationship"
  },
  "CanBeInManyToMany": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canbeinmanytomany"
  },
  "CanEnableSyncToExternalSearchIndex": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canenablesynctoexternalsearchindex"
  },
  "SyncToExternalSearchIndex": false,
  "CanModifyAdditionalSettings": {
    "Value": true,
    "CanBeChanged": true,
```

```
"ManagedPropertyLogicalName": "canmodifyadditionalsettings"
},
"CanChangeHierarchicalRelationship": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canchangehierarchicalrelationship"
},
"IsOptimisticConcurrencyEnabled": true,
"ChangeTrackingEnabled": false,
"IsImportable": true,
"IsIntersect": false,
"IsMailMergeEnabled": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifymailmergesettings"
},
"IsManaged": false,
"IsEnabledForCharts": true,
"IsEnabledForTrace": false,
"IsValidForAdvancedFind": true,
"IsVisibleInMobile": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifymobilevisibility"
},
"IsVisibleInMobileClient": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifymobileclientvisibility"
},
"IsReadOnlyInMobileClient": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifymobileclientreadonly"
},
"IsOfflineInMobileClient": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifymobileclientoffline"
},
"DaysSinceRecordLastModified": 0,
"IsReadingPaneEnabled": true,
"IsQuickCreateEnabled": false,
"LogicalName": "new_bankaccount",
"ObjectTypeCode": 10009,
"OwnershipType": "UserOwned",
"PrimaryNameAttribute": "new_accountname",
"PrimaryImageAttribute": null,
"PrimaryIdAttribute": "new_bankaccountid",
"Privileges": [
{
    "CanBeBasic": true,
    "CanBeDeep": true,
    "CanBeGlobal": true,
    "CanBeLocal": true,
```

```
"CanBeEntityReference": false,
"CanBeParentEntityReference": false,
"Name": "prvCreatenew_BankAccount",
"PrivilegeId": "d1a8de4b-27df-42e1-bc5c-b863e002b37f",
"PrivilegeType": "Create"
},
{
"CanBeBasic": true,
"CanBeDeep": true,
"CanBeGlobal": true,
"CanBeLocal": true,
"CanBeEntityReference": false,
"CanBeParentEntityReference": false,
"Name": "prvReadnew_BankAccount",
"PrivilegeId": "726043b1-de2c-487e-9d6d-5629fca2bf22",
"PrivilegeType": "Read"
},
{
"CanBeBasic": true,
"CanBeDeep": true,
"CanBeGlobal": true,
"CanBeLocal": true,
"CanBeEntityReference": false,
"CanBeParentEntityReference": false,
"Name": "prvWritenew_BankAccount",
"PrivilegeId": "fa50c539-b6c7-4eaf-bd49-fd8224bc51b6",
"PrivilegeType": "Write"
},
{
"CanBeBasic": true,
"CanBeDeep": true,
"CanBeGlobal": true,
"CanBeLocal": true,
"CanBeEntityReference": false,
"CanBeParentEntityReference": false,
"Name": "prvDeletenew_BankAccount",
"PrivilegeId": "17c1fd6e-f856-45e7-b563-796f53108b85",
"PrivilegeType": "Delete"
},
{
"CanBeBasic": true,
"CanBeDeep": true,
"CanBeGlobal": true,
"CanBeLocal": true,
"CanBeEntityReference": false,
"CanBeParentEntityReference": false,
"Name": "prvAssignnew_BankAccount",
"PrivilegeId": "133ca81d-668e-4c19-a71e-10c6dfe099cd",
"PrivilegeType": "Assign"
},
{
"CanBeBasic": true,
"CanBeDeep": true,
"CanBeGlobal": true,
"CanBeLocal": true,
```

```

    "CanBeEntityReference": false,
    "CanBeParentEntityReference": false,
    "Name": "prvSharenew_BankAccount",
    "PrivilegeId": "15f27df4-9c67-47c9-b1f1-274e1c44f24a",
    "PrivilegeType": "Share"
},
{
    "CanBeBasic": true,
    "CanBeDeep": true,
    "CanBeGlobal": true,
    "CanBeLocal": true,
    "CanBeEntityReference": false,
    "CanBeParentEntityReference": false,
    "Name": "prvAppendnew_BankAccount",
    "PrivilegeId": "ac8b1920-8f93-4e9d-94e3-c680e2a2f228",
    "PrivilegeType": "Append"
},
{
    "CanBeBasic": true,
    "CanBeDeep": true,
    "CanBeGlobal": true,
    "CanBeLocal": true,
    "CanBeEntityReference": false,
    "CanBeParentEntityReference": false,
    "Name": "prvAppendTonew_BankAccount",
    "PrivilegeId": "f63a5f46-3bc7-4eac-81d0-7f77f566ef46",
    "PrivilegeType": "AppendTo"
}
],
{
    "RecurrenceBaseEntityLogicalName": null,
    "ReportViewName": "Filterednew_BankAccount",
    "SchemaName": "new_BankAccount",
    "IntroducedVersion": "1.0",
    "IsStateModelAware": true,
    "EnforceStateTransitions": false,
    "EntityColor": null,
    "LogicalCollectionName": "new_bankaccounts",
    "CollectionSchemaName": "new_BankAccounts",
    "EntitySetName": "new_bankaccounts",
    "IsEnabledForExternalChannels": false,
    "IsPrivate": false,
    "MetadataId": "417129e1-207c-e511-80d2-00155d2a68d2",
    "HasChanged": null
}

```

Respuesta:

HTTP

HTTP/1.1 204 No Content
 OData-Version: 4.0

Consulte también

- [Usar la API web con metadatos de Microsoft Dataverse](#)
- [Crear y actualizar definiciones de columna mediante la API web](#)
- [Consultar definiciones de tabla con la API web](#)
- [Recuperar definiciones de tablas por nombre o Metadatald](#)
- [Relaciones de tabla de modelos usando la API web](#)
- [Trabajar con definiciones de tablas mediante SDK para .NET](#)
- [Definiciones de columna \(atributo\)](#)
- [Ejemplo de operaciones de metadatos de API web](#)
- [Ejemplo de operaciones de metadatos de API web \(C#\)](#)

Nota

¿Puede indicarnos sus preferencias de idioma de documentación? [Realice una breve encuesta.](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad](#)).

Crear y actualizar definiciones de columna mediante la API web

Artículo • 12/12/2023

Puede realizar las mismas operaciones en las definiciones de columna utilizando la API web que con el SDK para .NET. Este artículo se centra en el trabajo con definiciones de columnas de tablas (metadatos) utilizando la API web.

Crear columnas

Puede crear columnas de tabla (atributos de entidad) al mismo tiempo que crea la definición de tabla al incluir la definición JSON de los atributos en la matriz `Attributes` para la entidad que publica además del atributo de cadena que sirve como atributo de nombre principal. Si desea agregar atributos a una entidad que ya está creada, puede enviar una solicitud `POST` incluyendo su definición JSON a la propiedad de navegación valorada como colección `Attributes` de la entidad.

Este ejemplo muestra cómo crear diferentes tipos de columnas

- [Crear una columna string](#)
- [Crear una columna dinero](#)
- [Crear una columna datetime](#)
- [Crear una columna booleana](#)
- [Crear una columna de búsqueda de clientes](#)
- [Crear una columna decimal](#)
- [Crear una columna de numero entero](#)
- [Crear una columna de memorándum](#)
- [Crear una columna de elección](#)
- [Crear una columna de opciones de selección múltiple](#)
- [Crear una columna de numero entero grande](#)

Crear una columna string

El siguiente ejemplo usará estas propiedades para crear una columna de cadena.

 Expandir tabla

Propiedades de <code>StringAttributeMetadata</code>	Valores
<code>SchemaName</code>	<code>new_BankName</code>

DisplayName's de StringAttributeMetadata	Valores
Description	Nombre del banco
RequiredLevel	None
MaxLength	100
FormatName	Text

El siguiente ejemplo crea una columna de cadena usando las propiedades y la agrega a la tabla `sample_bankaccount`.

El URI para el atributo se devuelve en la respuesta.

Solicitud:

HTTP

```

POST [Organization
URI]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/Attributes HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
Accept: application/json
Content-Type: application/json; charset=utf-8
OData-MaxVersion: 4.0
OData-Version: 4.0

{
  "AttributeType": "String",
  "AttributeTypeName": {
    "Value": "StringType"
  },
  "Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
      {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Type the name of the bank",
        "LanguageCode": 1033
      }
    ]
  },
  "DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
      {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Bank Name",
        "LanguageCode": 1033
      }
    ]
  }
}

```

```

},
"RequiredLevel": {
  "Value": "None",
  "CanBeChanged": true,
  "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
},
"SchemaName": "new_BankName",
"@odata.type": "Microsoft.Dynamics.CRM.StringAttributeMetadata",
"FormatName": {
  "Value": "Text"
},
"MaxLength": 100
}

```

Respuesta:

HTTP

```

HTTP/1.1 204 No Content
OData-Version: 4.0
OData-EntityId: [Organization
URI]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/Attributes(f01bef16-287c-e511-80d2-00155d2a68d2)

```

Crear una columna dinero

El siguiente ejemplo usará estas propiedades para crear un atributo monetario.

[] Expandir tabla

Propiedades de	Valores
MoneyAttributeMetadata	
SchemaName	new_Balance
DisplayName	Saldo
Description	Especifique el importe de saldo.
RequiredLevel	None
PrecisionSource	2 Nota: para obtener información sobre los valores válidos para PrecisionSource, vea MoneyType . El valor 2 indica que el nivel de precisión decimal coincidirá con TransactionCurrency.CurrencyPrecision , que está asociada con el registro actual.

El siguiente ejemplo crea un atributo monetario usando las propiedades y la agrega a la tabla `sample_bankaccount`. El URI para el atributo se devuelve en la respuesta.

Solicitud:

HTTP

```
POST [Organization
URI]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/Attributes
HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
Accept: application/json
Content-Type: application/json; charset=utf-8
OData-MaxVersion: 4.0
OData-Version: 4.0

{
  "AttributeType": "Money",
  "AttributeTypeName": {
    "Value": "MoneyType"
  },
  "Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
      {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Enter the balance amount",
        "LanguageCode": 1033
      }
    ]
  },
  "DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
      {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Balance",
        "LanguageCode": 1033
      }
    ]
  },
  "RequiredLevel": {
    "Value": "None",
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
  },
  "SchemaName": "new_Balance",
  "@odata.type": "Microsoft.Dynamics.CRM.MoneyAttributeMetadata",
  "PrecisionSource": 2
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 No Content
OData-Version: 4.0
OData-EntityId: [Organization
URI]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/Attributes(f11bef16-287c-e511-80d2-00155d2a68d2)
```

Crear una columna datetime

El siguiente ejemplo usará estas propiedades para crear un atributo de fecha y hora.

 Expandir tabla

Propiedades de DateTimeAttributeMetadata	Valores
SchemaName	new_Checkeddate
DisplayName	Date
Description	Fecha en que se confirmó por última vez el saldo de cuenta.
RequiredLevel	None
Format	DateOnly Nota: para las opciones válidas para esta propiedad, consulte DateTimeFormat EnumType

El siguiente ejemplo crea un atributo datetime usando las propiedades y la agrega a la tabla `sample_bankaccount`. El URI para el atributo se devuelve en la respuesta.

Solicitud:

HTTP

```
POST [Organization
URI]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/Attributes HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
Accept: application/json
Content-Type: application/json; charset=utf-8
OData-MaxVersion: 4.0
OData-Version: 4.0

{
  "AttributeType": "DateTime",
  "AttributeTypeName": {
    "Value": "DateTimeType"
}
```

```

},
"Description": {
  "@odata.type": "Microsoft.Dynamics.CRM.Label",
  "LocalizedLabels": [
    {
      "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
      "Label": "The date the account balance was last confirmed",
      "LanguageCode": 1033
    }
  ]
},
"DisplayName": {
  "@odata.type": "Microsoft.Dynamics.CRM.Label",
  "LocalizedLabels": [
    {
      "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
      "Label": "Date",
      "LanguageCode": 1033
    }
  ]
},
"RequiredLevel": {
  "Value": "None",
  "CanBeChanged": true,
  "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
},
"SchemaName": "new_Checkeddate",
"@odata.type": "Microsoft.Dynamics.CRM.DateTimeAttributeMetadata",
"Format": "DateOnly"
}

```

Respuesta:

HTTP

```

HTTP/1.1 204 No Content
OData-Version: 4.0
OData-EntityId: [Organization
URI]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/Attri
butes(fe1bef16-287c-e511-80d2-00155d2a68d2)

```

Crear una columna booleana

El siguiente ejemplo usará estas propiedades para crear una columna booleana.

[] Expandir tabla

Propiedades de BooleanAttributeMetadata	Valores
SchemaName	new_Boolean

DisplayNames de BooleanAttributeMetadata	Valores de muestra
Description	Atributo booleano
RequiredLevel	None
OptionSet.TrueOption	VERDADERO
OptionSet.FalseOption	False

El siguiente ejemplo crea un atributo booleano mediante las propiedades y lo agrega a la entidad con el valor `LogicalName` de `new_bankaccount`. El URI para el atributo se devuelve en la respuesta.

Solicitud:

HTTP

```

POST [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='new_bankaccount')/Attribut
es HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanAttributeMetadata",
    "AttributeType": "Boolean",
    "AttributeTypeName": {
        "Value": "BooleanType"
    },
    "DefaultValue": false,
    "OptionSet": {
        "TrueOption": {
            "Value": 1,
            "Label": {
                "@odata.type": "Microsoft.Dynamics.CRM.Label",
                "LocalizedLabels": [
                    {
                        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                        "Label": "True",
                        "LanguageCode": 1033,
                        "IsManaged": false
                    }
                ]
            }
        },
        "FalseOption": {
            "Value": 0,
            "Label": {

```

```

"@odata.type": "Microsoft.Dynamics.CRM.Label",
"LocalizedLabels": [
    {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "False",
        "LanguageCode": 1033,
        "IsManaged": false
    }
]
},
"OptionsetType": "Boolean"
},
"Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Boolean Attribute",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ]
},
"DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Sample Boolean",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ]
},
"RequiredLevel": {
    "Value": "None",
    "CanBeChanged": false,
    "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
},
"SchemaName": "new_Boolean"
}

```

Respuesta:

HTTP

```

HTTP/1.1 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='new_bankaccount')/Attribut
es(38d20735-6817-ed11-b83e-00224837179f)

```

Crear una columna de búsqueda de clientes

A diferencia de otros atributos, un atributo de búsqueda de cliente se crea mediante la acción [CreateCustomerRelationships](#).

Los parámetros de esta acción requieren la definición del atributo de búsqueda y un par de relaciones de uno a varios. Un atributo de búsqueda de cliente tiene dos relaciones de uno a varios: una para la entidad de cuenta y la otra para la entidad de contacto.

El siguiente ejemplo usará estas propiedades para crear un atributo de búsqueda de clientes.

 Expandir tabla

Propiedades del atributo de búsqueda de clientes	Valores
SchemaName	new_CustomerId
DisplayName	Customer
Description	Atributo de búsqueda de clientes de ejemplo

El ejemplo crea un atributo de búsqueda de clientes `new_CustomerId`, y lo agrega a la entidad personalizada: `new_bankaccount`. La respuesta es un [CreateCustomerRelationshipsResponse ComplexType](#)".

Solicitud:

HTTP

```
POST [Organization URI]/api/data/v9.2/CreateCustomerRelationships HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
Accept: application/json
Content-Type: application/json; charset=utf-8

{
    "OneToManyRelationships": [
        {
            "SchemaName": "new_bankaccount_customer_account",
            "ReferencedEntity": "account",
            "ReferencingEntity": "new_bankaccount"
        },
        {
            "SchemaName": "new_bankaccount_customer_contact",
            "ReferencedEntity": "contact",
            "ReferencingEntity": "new_bankaccount"
        }
    ]
}
```

```

    "Lookup": {
        "AttributeType": "Lookup",
        "AttributeName": {
            "Value": "LookupType"
        },
        "Description": {
            "@odata.type": "Microsoft.Dynamics.CRM.Label",
            "LocalizedLabels": [
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Sample Customer Lookup Attribute",
                "LanguageCode": 1033
            ],
            "UserLocalizedLabel": {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Sample Customer Lookup Attribute",
                "LanguageCode": 1033
            }
        },
        "DisplayName": {
            "@odata.type": "Microsoft.Dynamics.CRM.Label",
            "LocalizedLabels": [
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Customer",
                "LanguageCode": 1033
            ],
            "UserLocalizedLabel": {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Customer",
                "LanguageCode": 1033
            }
        },
        "SchemaName": "new_CustomerId",
        "@odata.type": "Microsoft.Dynamics.CRM.ComplexLookupAttributeMetadata"
    }
}

```

Respuesta:

HTTP

```

HTTP/1.1 200 OK
Content-Type: application/json; odata.metadata=minimal
OData-Version: 4.0

{
    "@odata.context": "[Organization
URI]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.CreateCustomerRelationsh
ipsResponse",
    "RelationshipIds": [
        "a7d261bc-3580-e611-80d7-00155d2a68de", "aed261bc-3580-e611-80d7-
00155d2a68de"
    ],
}

```

```
"AttributeId": "39a5d94c-e8a2-4a41-acc0-8487242d455e"
```

```
}
```

Crear una columna decimal

El siguiente ejemplo usará estas propiedades para crear una columna decimal.

 Expandir tabla

Propiedades de DecimalAttributeMetadata	Valores
SchemaName	sample_Decimal
DisplayName	Muestra decimal
Description	Atributo decimal
RequiredLevel	None
.MaxValue	100.0
.MinValue	0.0
Precision	1

El siguiente ejemplo crea un atributo decimal usando las propiedades y la agrega a la tabla `sample_bankaccount`.

El URI para el atributo se devuelve en la respuesta.

Solicitud:

HTTP

```
POST [Organization  
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/Attributes  
MSCRM.SolutionUniqueName: examplesolution  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json  
Content-Type: application/json; charset=utf-8  
Content-Length: 1370  
  
{  
    "@odata.type": "Microsoft.Dynamics.CRM.DecimalAttributeMetadata",  
    "AttributeType": "Decimal",  
}
```

```

    "AttributeTypeName": {
        "Value": "DecimalType"
    },
    ".MaxValue": 100.0,
    ".MinValue": 0.0,
    "Precision": 1,
    "Description": {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [
            {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Decimal Attribute",
                "LanguageCode": 1033,
                "IsManaged": false
            }
        ],
        "UserLocalizedLabel": {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Decimal Attribute",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    },
    "DisplayName": {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [
            {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Sample Decimal",
                "LanguageCode": 1033,
                "IsManaged": false
            }
        ],
        "UserLocalizedLabel": {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Sample Decimal",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    },
    "RequiredLevel": {
        "Value": "None",
        "CanBeChanged": false,
        "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
    },
    "SchemaName": "sample_Decimal"
}

```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/Attri
butes(dba9c0df-4c05-ee11-8f6e-000d3a993550)
```

Crear una columna de numero entero

El siguiente ejemplo usará estas propiedades para crear una columna de números enteros.

 Expandir tabla

Propiedades de IntegerAttributeMetadata	Valores
SchemaName	sample_Integer
DisplayName	Entero de muestra
Description	Atributo entero
RequiredLevel	None
.MaxValue	100
.MinValue	0
Format	None

El siguiente ejemplo crea una columna de números enteros usando las propiedades y la agrega a la tabla `sample_bankaccount`.

El URI para la columna se devuelve en la respuesta.

Solicitud:

HTTP

```
POST [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/Attri
butes
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
Content-Type: application/json; charset=utf-8
Content-Length: 1392
```

```
{
    "@odata.type": "Microsoft.Dynamics.CRM.IntegerAttributeMetadata",
    "AttributeType": "Integer",
    "AttributeName": {
        "Value": "IntegerType"
    },
    ".MaxValue": 100,
    ".MinValue": 0,
    "Format": "None",
    "SourceTypeMask": 0,
    "Description": {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [
            {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Integer Attribute",
                "LanguageCode": 1033,
                "IsManaged": false
            }
        ],
        "UserLocalizedLabel": {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Integer Attribute",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    },
    "DisplayName": {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [
            {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Sample Integer",
                "LanguageCode": 1033,
                "IsManaged": false
            }
        ],
        "UserLocalizedLabel": {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Sample Integer",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    },
    "RequiredLevel": {
        "Value": "None",
        "CanBeChanged": false,
        "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
    },
    "SchemaName": "sample_Integer"
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/Attributes(17aac0df-4c05-ee11-8f6e-000d3a993550)
```

Crear una columna de memorándum

El siguiente ejemplo usará estas propiedades para crear una columna de memorándum.

 Expandir tabla

Propiedades de MemoAttributeMetadata	Valores
SchemaName	sample_Memo
DisplayName	Memorándum de muestra
Description	Atributo de memorando
RequiredLevel	None
MaxLength	500
Format	TextArea

El siguiente ejemplo crea una columna de memorando usando las propiedades y la agrega a la tabla `sample_bankaccount`.

El URI para el atributo se devuelve en la respuesta.

Solicitud:

HTTP

```
POST [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/Attributes
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
Content-Type: application/json; charset=utf-8
Content-Length: 1384

{
```

```

"@odata.type": "Microsoft.Dynamics.CRM.MemoAttributeMetadata",
"AttributeType": "Memo",
"AttributeName": {
    "Value": "MemoType"
},
"Format": "TextArea",
"ImeMode": "Disabled",
"MaxLength": 500,
"IsLocalizable": false,
"Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Memo Attribute",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Memo Attribute",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Sample Memo",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Sample Memo",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"RequiredLevel": {
    "Value": "None",
    "CanBeChanged": false,
    "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
},
"SchemaName": "sample_Memo"
}

```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/Attributes(b12d3cee-4c05-ee11-8f6e-000d3a993550)
```

Crear una columna de elección

El siguiente ejemplo usará estas propiedades para crear una columna de elección local.

 Expandir tabla

Propiedades de PicklistAttributeMetadata	Valores
SchemaName	sample.Choice
DisplayName	Elección de muestra
Description	Elegir un atributo
RequiredLevel	None
OptionSet	valor: 727000000, etiqueta: Bravo valor: 727000001, etiqueta: Delta valor: 727000002, etiqueta: Alpha valor: 727000003, etiqueta: Charlie valor: 727000004, etiqueta: Foxtrot

El siguiente ejemplo crea una columna local usando las propiedades y la agrega a la tabla `sample_bankaccount`.

El URI para el atributo se devuelve en la respuesta.

Nota

Para ver un ejemplo que muestra cómo crear una columna de opciones con un conjunto de opciones global, consulte [Crear una columna de opciones usando un conjunto de opciones global](#)

Solicitud:

HTTP

```
POST [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/Attributes
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
Content-Type: application/json; charset=utf-8
Content-Length: 4524

{
    "@odata.type": "Microsoft.Dynamics.CRM.PicklistAttributeMetadata",
    "AttributeType": "Picklist",
    "AttributeName": {
        "Value": "PicklistType"
    },
    "SourceTypeMask": 0,
    "OptionSet": {
        "@odata.type": "Microsoft.Dynamics.CRM.OptionSetMetadata",
        "Options": [
            {
                "Value": 727000000,
                "Label": {
                    "@odata.type": "Microsoft.Dynamics.CRM.Label",
                    "LocalizedLabels": [
                        {
                            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                            "Label": "Bravo",
                            "LanguageCode": 1033,
                            "IsManaged": false
                        }
                    ],
                    "UserLocalizedLabel": {
                        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                        "Label": "Bravo",
                        "LanguageCode": 1033,
                        "IsManaged": false
                    }
                }
            }
        ],
        "Value": 727000001,
        "Label": {
            "@odata.type": "Microsoft.Dynamics.CRM.Label",
            "LocalizedLabels": [
                {
                    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                    "Label": "Delta",
                    "LanguageCode": 1033,
                    "IsManaged": false
                }
            ],
            "UserLocalizedLabel": {
```

```
    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
    "Label": "Delta",
    "LanguageCode": 1033,
    "IsManaged": false
  }
}
},
{
  "Value": 727000002,
  "Label": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
      {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Alpha",
        "LanguageCode": 1033,
        "IsManaged": false
      }
    ],
    "UserLocalizedLabel": {
      "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
      "Label": "Alpha",
      "LanguageCode": 1033,
      "IsManaged": false
    }
  }
},
{
  "Value": 727000003,
  "Label": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
      {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Charlie",
        "LanguageCode": 1033,
        "IsManaged": false
      }
    ],
    "UserLocalizedLabel": {
      "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
      "Label": "Charlie",
      "LanguageCode": 1033,
      "IsManaged": false
    }
  }
},
{
  "Value": 727000004,
  "Label": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
      {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Foxtrot",
        "LanguageCode": 1033,
        "IsManaged": false
      }
    ]
  }
}
```

```
        "LanguageCode": 1033,
        "IsManaged": false
    }
],
"UserLocalizedLabel": {
    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
    "Label": "Foxtrot",
    "LanguageCode": 1033,
    "IsManaged": false
}
}
},
"IsGlobal": false,
"OptionsetType": "Picklist"
},
"Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Choice Attribute",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Choice Attribute",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Sample Choice",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Sample Choice",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"RequiredLevel": {
    "Value": "None",
    "CanBeChanged": false,
    "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
},
```

```
        "SchemaName": "sample_Choice"  
    }
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent  
OData-Version: 4.0  
OData-EntityId: [Organization  
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/Attri  
butes(eed205fb-4c05-ee11-8f6e-000d3a993550)
```

Crear una columna de opciones de selección múltiple

El siguiente ejemplo usará estas propiedades para crear una columna de elección local de selección múltiple.

 Expandir tabla

Propiedades de MultiSelectPicklistAttributeMetadata	Valores
SchemaName	sample_Choice
DisplayName	Elección de muestra
Description	Elegir un atributo
RequiredLevel	None
OptionSet	valor: 727000000, etiqueta: Entrante valor: 727000001, etiqueta:Entree valor: 727000002, etiqueta:Postre

El siguiente ejemplo crea una columna local de elección múltiple usando las propiedades y la agrega a la tabla `sample_bankaccount`.

El URI para el atributo se devuelve en la respuesta.

Solicitud:

HTTP

```
POST [Organization  
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/Attri  
butes  
MSCRM.SolutionUniqueName: examplesolution
```

```
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
Content-Type: application/json; charset=utf-8
Content-Length: 3404

{
    "@odata.type": "Microsoft.Dynamics.CRM.MultiSelectPicklistAttributeMetadata",
    "AttributeType": "Virtual",
    "AttributeName": {
        "Value": "MultiSelectPicklistType"
    },
    "SourceTypeMask": 0,
    "OptionSet": {
        "@odata.type": "Microsoft.Dynamics.CRM.OptionSetMetadata",
        "Options": [
            {
                "Value": 727000000,
                "Label": {
                    "@odata.type": "Microsoft.Dynamics.CRM.Label",
                    "LocalizedLabels": [
                        {
                            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                            "Label": "Appetizer",
                            "LanguageCode": 1033,
                            "IsManaged": false
                        }
                    ],
                    "UserLocalizedLabel": {
                        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                        "Label": "Appetizer",
                        "LanguageCode": 1033,
                        "IsManaged": false
                    }
                }
            }
        ],
        "Value": 727000001,
        "Label": {
            "@odata.type": "Microsoft.Dynamics.CRM.Label",
            "LocalizedLabels": [
                {
                    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                    "Label": "Entree",
                    "LanguageCode": 1033,
                    "IsManaged": false
                }
            ],
            "UserLocalizedLabel": {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Entree",
                "LanguageCode": 1033,
                "IsManaged": false
            }
        }
    }
}
```

```
        }
    },
},
{
    "Value": 727000002,
    "Label": {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [
            {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Dessert",
                "LanguageCode": 1033,
                "IsManaged": false
            }
        ],
        "UserLocalizedLabel": {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Dessert",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    }
},
    "IsGlobal": false,
    "OptionsetType": "Picklist"
},
"Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "MultiSelect Choice Attribute",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "MultiSelect Choice Attribute",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Sample MultiSelect Choice",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
```

```

    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
    "Label": "Sample MultiSelect Choice",
    "LanguageCode": 1033,
    "IsManaged": false
  }
},
"RequiredLevel": {
  "Value": "None",
  "CanBeChanged": false,
  "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
},
"SchemaName": "sample_MultiSelectChoice"
}

```

Respuesta:

HTTP

```

HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/Attri
butes(afd63201-4d05-ee11-8f6e-000d3a993550)

```

Crear una columna de numero entero grande

El siguiente ejemplo usará estas propiedades para crear una columna de números enteros grandes.

[] Expandir tabla

Propiedades BigIntAttributeMetadata	Valores
SchemaName	sample_BigInt
DisplayName	Muestra bigint
Description	Atributo de bigint
RequiredLevel	None

El siguiente ejemplo crea una columna de números enteros usando las propiedades y la agrega a la tabla `sample_bankaccount`.

El URI para la columna se devuelve en la respuesta.

Solicitar

HTTP

```
POST [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/Attributes
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
Authorization: Bearer <access token>
Content-Type: application/json; charset=utf-8
Content-Length: 1301

{
  "AttributeType": "BigInt",
  "AttributeName": {
    "Value": "BigIntType"
  },
  "@odata.type": "Microsoft.Dynamics.CRM.BigIntAttributeMetadata",
  "Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
      {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "BigInt Attribute",
        "LanguageCode": 1033,
        "IsManaged": false
      }
    ],
    "UserLocalizedLabel": {
      "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
      "Label": "BigInt Attribute",
      "LanguageCode": 1033,
      "IsManaged": false
    }
  },
  "DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
      {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Sample BigInt",
        "LanguageCode": 1033,
        "IsManaged": false
      }
    ],
    "UserLocalizedLabel": {
      "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
      "Label": "Sample BigInt",
      "LanguageCode": 1033,
      "IsManaged": false
    }
  }
},
```

```
"RequiredLevel": {  
    "Value": "None",  
    "CanBeChanged": false,  
    "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"  
},  
"SchemaName": "sample_BigInt"  
}
```

Response

HTTP

```
HTTP/1.1 204 NoContent  
OData-Version: 4.0  
OData-EntityId: [Organization  
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/Attri  
butes(28b90018-1d99-ee11-be37-000d3a993223)
```

Actualizar una columna

Como se menciona en [Actualizar definiciones de tablas](#), las entidades del modelo de datos se actualizan mediante el método HTTP `PUT` con la definición JSON completa del elemento actual. Este patrón se aplica tanto a los atributos de entidad como a las entidades. Al igual con las entidades, puede la opción de sobrescribir etiquetas mediante el encabezado `MSCRM.MergeLabels` con el valor establecido en `false`, y debe publicar personalizaciones antes de que estén activas en el sistema.

Usando el atributo booleano creado anteriormente en [Crear una columna booleana](#), debemos, en primer lugar, recuperar el atributo completo.

Solicitud:

HTTP

```
GET [Organization  
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='new_bankaccount')/Attribut  
es(LogicalName='new_boolean')/Microsoft.Dynamics.CRM.BooleanAttributeMetadat  
a HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json
```

Respuesta:

HTTP

HTTP/1.1 OK

OData-Version: 4.0

```
{  
    "@odata.context": "[Organization  
Uri]/api/data/v9.2/$metadata#EntityDefinitions('new_bankaccount')/Attributes  
/Microsoft.Dynamics.CRM.BooleanAttributeMetadata/$entity",  
    "MetadataId": "38d20735-6817-ed11-b83e-00224837179f",  
    "HasChanged": null,  
    "AttributeOf": null,  
    "AttributeType": "Boolean",  
    "ColumnNumber": 35,  
    "DeprecatedVersion": null,  
    "IntroducedVersion": "1.0.0.0",  
    "EntityLogicalName": "new_bankaccount",  
    "IsCustomAttribute": true,  
    "IsPrimaryId": false,  
    "IsValidODataAttribute": true,  
    "IsPrimaryName": false,  
    "IsValidForCreate": true,  
    "IsValidForRead": true,  
    "IsValidForUpdate": true,  
    "CanBeSecuredForRead": true,  
    "CanBeSecuredForCreate": true,  
    "CanBeSecuredForUpdate": true,  
    "IsSecured": false,  
    "IsRetrievable": false,  
    "IsFilterable": false,  
    "IsSearchable": false,  
    "IsManaged": false,  
    "LinkedAttributeId": null,  
    "LogicalName": "new_boolean",  
    "IsValidForForm": true,  
    "IsRequiredForForm": false,  
    "IsValidForGrid": true,  
    "SchemaName": "new_Boolean",  
    "ExternalName": null,  
    "IsLogical": false,  
    "IsDataSourceSecret": false,  
    "InheritsFrom": null,  
    "CreatedOn": "2022-08-08T22:19:49Z",  
    "ModifiedOn": "2022-08-08T22:19:49Z",  
    "SourceType": 0,  
    "AutoNumberFormat": null,  
    "DefaultValue": false,  
    "FormulaDefinition": "",  
    "SourceTypeMask": 0,  
    "AttributeTypeName": {  
        "Value": "BooleanType"  
    },  
    "Description": {  
        "LocalizedLabels": [  
            {  
                "Label": "Boolean Attribute",  
            }  
        ]  
    }  
}
```

```
        "LanguageCode": 1033,
        "IsManaged": false,
        "MetadataId": "2b5545d2-a59b-4503-8846-95665283b41f",
        "HasChanged": null
    }
],
"UserLocalizedLabel": {
    "Label": "Boolean Attribute",
    "LanguageCode": 1033,
    "IsManaged": false,
    "MetadataId": "2b5545d2-a59b-4503-8846-95665283b41f",
    "HasChanged": null
}
},
"DisplayName": {
    "LocalizedLabels": [
        {
            "Label": "Sample Boolean",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "e6b8d06c-067e-4ab0-a9a8-32f520b25e36",
            "HasChanged": null
        }
    ],
    "UserLocalizedLabel": {
        "Label": "Sample Boolean",
        "LanguageCode": 1033,
        "IsManaged": false,
        "MetadataId": "e6b8d06c-067e-4ab0-a9a8-32f520b25e36",
        "HasChanged": null
    }
},
"IsAuditEnabled": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyauditsettings"
},
"IsGlobalFilterEnabled": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyglobalfiltersettings"
},
"IsSortableEnabled": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyissortablesettings"
},
"IsCustomizable": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "iscustomizable"
},
"IsRenameable": {
    "Value": true,
    "CanBeChanged": true,
```

```

        "ManagedPropertyLogicalName": "isrenameable"
    },
    "IsValidForAdvancedFind": {
        "Value": true,
        "CanBeChanged": true,
        "ManagedPropertyLogicalName": "canmodifysearchsettings"
    },
    "RequiredLevel": {
        "Value": "None",
        "CanBeChanged": false,
        "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
    },
    "CanModifyAdditionalSettings": {
        "Value": true,
        "CanBeChanged": true,
        "ManagedPropertyLogicalName": "canmodifyadditionalsettings"
    },
    "Settings": []
}

```

Después, cambie las propiedades que desee cambiar.

[+] Expandir tabla

Propiedades de BooleanAttributeMetadata	Valores antiguos	Valores nuevos
DisplayName	Booleano de muestra	Booleano de muestra actualizado
Description	Atributo booleano	Atributo booleano actualizado
RequiredLevel	None	ApplicationRequired

ⓘ Nota

Si desea actualizar las opciones, debe enviar una solicitud diferente. Más información: [Actualizar opciones](#).

Ahora puede enviar la solicitud `PUT` con las propiedades modificadas:

Solicitud:

HTTP

```

PUT [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='new_bankaccount')/Attribut
es(LogicalName='new_boolean') HTTP/1.1

```

```
MSCRM.SolutionUniqueName: examplesolution
MSCRM.MergeLabels: true
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanAttributeMetadata",
    "MetadataId": "e160ff9b-6f17-ed11-b83e-00224837179f",
    "HasChanged": null,
    "AttributeOf": null,
    "AttributeType": "Boolean",
    "ColumnNumber": 35,
    "DeprecatedVersion": null,
    "IntroducedVersion": "1.0.0.0",
    "EntityLogicalName": "new_bankaccount",
    "IsCustomAttribute": true,
    "IsPrimaryId": false,
    "IsValidODataAttribute": true,
    "IsPrimaryName": false,
    "IsValidForCreate": true,
    "IsValidForRead": true,
    "IsValidForUpdate": true,
    "CanBeSecuredForRead": true,
    "CanBeSecuredForCreate": true,
    "CanBeSecuredForUpdate": true,
    "IsSecured": false,
    "IsRetrievable": false,
    "IsFilterable": false,
    "IsSearchable": false,
    "IsManaged": false,
    "LinkedAttributeId": null,
    "LogicalName": "new_boolean",
    "IsValidForForm": true,
    "IsRequiredForForm": false,
    "IsValidForGrid": true,
    "SchemaName": "new_Boolean",
    "ExternalName": null,
    "IsLogical": false,
    "IsDataSourceSecret": false,
    "InheritsFrom": null,
    "CreatedOn": "2022-08-08T23:12:55Z",
    "ModifiedOn": "2022-08-08T23:12:55Z",
    "SourceType": 0,
    "AutoNumberFormat": null,
    "DefaultValue": false,
    "FormulaDefinition": "",
    "SourceTypeMask": 0,
    "AttributeTypeName": {
        "Value": "BooleanType"
    },
    "Description": {
        "LocalizedLabels": [
            {
                "Label": "New Boolean"
            }
        ]
    }
}
```

```
        "Label": "Boolean Attribute Updated",
        "LanguageCode": 1033,
        "IsManaged": false,
        "MetadataId": "b370f0dd-329f-434e-8b9f-b6eb47d0141f",
        "HasChanged": null
    }
],
"UserLocalizedLabel": {
    "Label": "Boolean Attribute Updated",
    "LanguageCode": 1033,
    "IsManaged": false,
    "MetadataId": "b370f0dd-329f-434e-8b9f-b6eb47d0141f",
    "HasChanged": null
}
},
"DisplayName": {
    "LocalizedLabels": [
        {
            "Label": "Sample Boolean Updated",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "63a9e5f1-e730-40cb-a410-495367d5856d",
            "HasChanged": null
        }
    ],
    "UserLocalizedLabel": {
        "Label": "Sample Boolean Updated",
        "LanguageCode": 1033,
        "IsManaged": false,
        "MetadataId": "63a9e5f1-e730-40cb-a410-495367d5856d",
        "HasChanged": null
    }
},
"IsAuditEnabled": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyauditsettings"
},
"IsGlobalFilterEnabled": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyglobalfiltersettings"
},
"IsSortableEnabled": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyissortablesettings"
},
"IsCustomizable": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "iscustomizable"
},
"IsRenameable": {
    "Value": true,
```

```
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "isrenameable"
},
"IsValidForAdvancedFind": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifysearchsettings"
},
"RequiredLevel": {
    "Value": "ApplicationRequired",
    "CanBeChanged": false,
    "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
},
"CanModifyAdditionalSettings": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyadditionalsettings"
},
"Settings": []
}
```

Respuesta:

HTTP

```
HTTP/1.1 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='new_bankaccount')/Attribut
es(LogicalName='new_boolean')
```

Consulte también

[Usar la API web con metadatos de Microsoft Dataverse](#)

[Crear y actualizar definiciones de tabla mediante la API web](#)

[Consultar definiciones de tabla con la API web](#)

[Recuperar definiciones de tablas por nombre o Metadatald](#)

[Relaciones de tabla de modelos usando la API web](#)

[Trabajar con definiciones de tablas mediante SDK para .NET](#)

[Definiciones de columna \(atributo\)](#)

[Ejemplo de operaciones de metadatos de API web](#)

[Ejemplo de operaciones de metadatos de API web \(C#\)](#)

 Nota

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta. [↗](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales (**declaración de privacidad** [↗](#)).

Crear y actualizar relaciones de tabla mediante la API web

Artículo • 24/08/2023

La API web admite trabajar con definiciones de relaciones (metadatos). Los conceptos descritos en [Definiciones de relación de tabla](#) también se aplican a la API web.

Idoneidad para relaciones

Antes de crear una tabla (entidad) entre entidades, debe comprobar si la tabla puede participar en la relación. Puede usar los acciones que aparecen en la siguiente tabla para determinar su idoneidad. Estas acciones corresponden a los mensajes de servicio de la organización descritos en [Elegibilidad de relación de tabla](#).

Acción	Descripción
CanBeReferenced Action	Comprueba si la entidad especificada puede ser la entidad principal (uno) en una relación de uno a varios.
CanBeReferencing Action	Comprueba si la entidad especificada puede ser la entidad de referencia (varios) en una relación de uno a varios.
CanManyToMany Action	Comprueba si la entidad puede participar en una relación de varios a varios.
GetValidManyToMany Function	Devuelve el conjunto de entidades que pueden participar en una relación de varios a varios.
GetValidReferencedEntities Function	Devuelve el conjunto de entidades que son válidas como entidad principal (uno) de la entidad especificada en una relación de uno a varios.
GetValidReferencingEntities Function	Devuelve el conjunto de entidades que son válidas como la entidad relacionada (varias) de la entidad especificada en una relación uno a varios.

Crear nueva relación de uno a varios

Al crear una relación de uno a varios, la define mediante [OneToManyRelationshipMetadata EntityType](#) el. Esta definición incluye el atributo de búsqueda, que se define utilizando [LookupAttributeMetadata EntityType](#) y también requiere propiedades complejas utilizando [AssociatedMenuConfiguration](#)

[ComplexType](#), [CascadeConfiguration](#) [ComplexType](#), [Label](#) [ComplexType](#) y [LocalizedLabel](#) [ComplexType](#). El atributo de búsqueda se establece como la propiedad de navegación de un solo valor Lookup del objeto [OneToManyRelationshipMetadata](#) y se crea a la vez mediante *inserción profunda*. Más información: [Crear filas de tablas relacionadas en una sola operación](#) y [Metadatos de relación de tablas](#)

Si desea aplicar un nombre de propiedad de navegación personalizado para una relación de uno a varios puede establecer valores para las propiedades

[ReferencingEntityPropertyName](#) y [ReferencedEntityPropertyName](#).

Una vez que haya generado el JSON necesario para definir la relación y el atributo de búsqueda, [POST](#) el JSON en el conjunto de entidades [RelationshipDefinitions](#). Debe incluir el valor de la propiedad [@odata.type](#) de

[Microsoft.Dynamics.CRM.OneToManyRelationshipMetadata](#) para aclarar el tipo de relación que está creando porque este mismo conjunto de entidades se usa para crear relaciones de varios a varios. El uri para la relación resultante se devuelve en la respuesta.

Solicitud:

HTTP

```
POST [Organization URI]/api/data/v9.2/RelationshipDefinitions HTTP/1.1
Accept: application/json
Content-Type: application/json; charset=utf-8
OData-MaxVersion: 4.0
OData-Version: 4.0

{
  "SchemaName": "new_contact_new_bankaccount",
  "@odata.type": "Microsoft.Dynamics.CRM.OneToManyRelationshipMetadata",
  "AssociatedMenuConfiguration": {
    "Behavior": "UseCollectionName",
    "Group": "Details",
    "Label": {
      "@odata.type": "Microsoft.Dynamics.CRM.Label",
      "LocalizedLabels": [
        {
          "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
          "Label": "Bank Accounts",
          "LanguageCode": 1033
        }
      ],
      "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Bank Accounts",
        "LanguageCode": 1033
      }
    },
    "Order": 10000
  }
}
```

```
},
"CascadeConfiguration": {
  "Assign": "Cascade",
  "Delete": "Cascade",
  "Merge": "Cascade",
  "Reparent": "Cascade",
  "Share": "Cascade",
  "Unshare": "Cascade"
},
"ReferencedAttribute": "contactid",
"ReferencedEntity": "contact",
"ReferencingEntity": "new_bankaccount",
"Lookup": {
  "AttributeType": "Lookup",
  "AttributeName": {
    "Value": "LookupType"
  },
  "Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
      {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "The owner of the account",
        "LanguageCode": 1033
      }
    ],
    "UserLocalizedLabel": {
      "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
      "Label": "The owner of the account",
      "LanguageCode": 1033
    }
  },
  "DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
      {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Account Owner",
        "LanguageCode": 1033
      }
    ],
    "UserLocalizedLabel": {
      "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
      "Label": "Account Owner",
      "LanguageCode": 1033
    }
  },
  "RequiredLevel": {
    "Value": "ApplicationRequired",
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
  },
  "SchemaName": "new_AccountOwner",
  "@odata.type": "Microsoft.Dynamics.CRM.LookupAttributeMetadata"
```

```
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 No Content
OData-Version: 4.0
OData-EntityId: [Organization
URI]/api/data/v9.2/RelationshipDefinitions(d475020f-5d7c-e511-80d2-
00155d2a68d2)
```

Crear nueva relación de varios a varios

Si desea aplicar un nombre de propiedad de navegación personalizado para una relación de varios a varios puede establecer valores para las propiedades `Entity1NavigationPropertyName` y `Entity2NavigationPropertyName`.

Una vez que haya generado el JSON necesario para definir la relación, `POST` el JSON en el conjunto de entidades `RelationshipDefinitions`. Debe incluir el valor de la propiedad `@odata.type` de `Microsoft.Dynamics.CRM.ManyToManyRelationshipMetadata` para aclarar el tipo de relación que está creando porque este mismo conjunto de entidades se usa para crear relaciones de uno a varios. El URI de la relación resultante se devuelve en la respuesta.

Solicitud:

HTTP

```
POST [Organization URI]/api/data/v9.2/RelationshipDefinitions HTTP/1.1
Accept: application/json
Content-Type: application/json; charset=utf-8
OData-MaxVersion: 4.0
OData-Version: 4.0

{
  "SchemaName": "new_accounts_campaigns",
  "@odata.type": "Microsoft.Dynamics.CRM.ManyToManyRelationshipMetadata",
  "Entity1AssociatedMenuConfiguration": {
    "Behavior": "UseLabel",
    "Group": "Details",
    "Label": {
      "@odata.type": "Microsoft.Dynamics.CRM.Label",
      "LocalizedLabels": [
        {
          "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
          "Label": "New Accounts-Campaigns"
        }
      ]
    }
  }
}
```

```

        "Label": "Account",
        "LanguageCode": 1033
    }
],
"UserLocalizedLabel": {
    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
    "Label": "Account",
    "LanguageCode": 1033
}
},
"Order": 10000
},
"Entity1LogicalName": "account",
"Entity2AssociatedMenuConfiguration": {
    "Behavior": "UseLabel",
    "Group": "Details",
    "Label": {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [
            {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Campaign",
                "LanguageCode": 1033
            }
        ],
        "UserLocalizedLabel": {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Campaign",
            "LanguageCode": 1033
        }
    },
    "Order": 10000
},
"Entity2LogicalName": "campaign",
"IntersectEntityName": "new_accounts_campaigns"
}

```

Respuesta:

HTTP

```

HTTP/1.1 204 No Content
OData-Version: 4.0
OData-EntityId: [Organization
URI]/api/data/v9.2/RelationshipDefinitions(420245fa-c77c-e511-80d2-
00155d2a68d2)

```

Cree Relaciones para admitir una búsqueda de varias tablas

Las columnas de tipo de búsqueda de varias tablas permiten al usuario utilizar una tabla específica que tiene varias tablas de relaciones uno a varios (1:M) en otras tablas del entorno. Una sola columna de tipo de búsqueda puede hacer referencia a otras tablas. Un valor de búsqueda enviado a la columna de tipo de tabla múltiple se comparará con un registro en cualquiera de las tablas relacionadas.

Más información: [Utilice columnas de búsqueda de varias tablas](#)

Actualizar relaciones

Como se discutió en [Actualizar definiciones de tabla](#), puede actualizar relaciones usando el método HTTP PUT para reemplazar la definición existente con los cambios que desea aplicar. No puede editar propiedades individuales utilizando el método HTTP PATCH tal como se puede hacer con las tablas de datos comerciales. Como ocurre con entidades y atributos, debe incluir un encabezado `MSCRM.MergeLabels` con el valor establecido como `true` para evitar sobrescribir etiquetas localizadas no incluidas en la actualización y debe publicar personalizaciones antes de que estén activas en el sistema.

Eliminar relaciones

Para eliminar una relación mediante web API, use el método HTTP DELETE con el identificador uniforme de recursos para la relación.

Consultar también

[Utilizar la API web con definiciones de tabla](#)

[Consultar definiciones de tabla con la API web](#)

[Recuperar definiciones de tablas por nombre o Metadatald](#)

[Modelar tablas y columnas usando la API web](#)

[Ejemplo de operaciones de metadatos de API web](#)

[Ejemplo de operaciones de metadatos de API web \(C#\)](#)

ⓘ Nota

¿Puede indicarnos sus preferencias de idioma de documentación? [Realice una breve encuesta.](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad](#)).

Búsquedas en varias tablas

Artículo • 25/08/2023

Las columnas de tipo de búsqueda de varias tablas permiten al usuario utilizar una tabla específica que tiene varias tablas de relaciones uno a varios (1:N) en otras tablas del entorno. Una sola columna de tipo de búsqueda puede hacer referencia a otras tablas. Un valor de búsqueda enviado a la columna de tipo de tabla múltiple se compara con un registro en cualquiera de las tablas relacionadas. Las búsquedas de tablas múltiples se pueden crear con tablas locales y tablas virtuales como tablas referenciadas.

Los tipos de tablas múltiples están integrados actualmente en Microsoft Dataverse como tipos estáticos como Cliente, que se conecta a Cuenta y Contacto. Esta nueva característica ofrece a los usuarios el poder de definir cualquier otra búsqueda de múltiples tablas que puedan necesitar.

ⓘ Nota

En este momento, los usuarios pueden crear y modificar búsquedas personalizadas de múltiples tablas a través del SDK o las API web. La compatibilidad con la interfaz de usuario interactiva vendrá en una versión futura.

Ejemplos

Supongamos que aloja medios para usuarios en una biblioteca. Tiene muchos MediaObjects diferentes, muchos de ellos tienen el mismo nombre, pero están en diferentes formatos, como Books, Audio y Video. La creación de una búsqueda de varias tablas llamada `new_Media` que tiene 1: N Relaciones a `new_Books`, `new_Audio`, y `new_Video` da como resultado una tabla de búsqueda `new_Media` que proporciona identificaciones rápidas de registros almacenados en tablas específicas.

Tabla de búsqueda new_Media

PrimaryID	PrimaryName	RelatedID	Related Name
<media1>	MediaObjectOne	<books1>	Content1
<media2>	MediaObjectTwo	<audio1>	Content1
<media3>	MediaObjectThree	<video1>	Content3

PrimaryID	PrimaryName	RelatedID	Related Name
<media4>	MediaObjectFour	<audio2>	Content3

new_Books table

PrimaryID	PrimaryName	CallNumber
<books1>	Content1	1ww-3452
<books2>	Content2	a4e-87hw

Tabla new_Audio

PrimaryID	PrimaryName	AudioFormat
<audio1>	Content1	mp4
<audio2>	Content3	wma

Tabla new_Video

PrimaryID	PrimaryName	VideoFormat
<video1>	Content3	wmv
<video2>	Content2	avi

La búsqueda de medios puede devolver registros en todas las tablas en la búsqueda polimórfica.

- Una búsqueda en medios con el nombre Content1 recuperaría registros de <books1> y <audio1>
- Una búsqueda en medios de Content3 recuperaría registros de <audio2> y <video1>

Ejemplo de API web

A continuación se muestra una publicación HTTP para un atributo de búsqueda polimórfica.

HTTP

```
POST [Organization URI]/api/data/v9.2/CreatePolymorphicLookupAttribute  
HTTP/1.1
```

```
Accept: application/json  
Content-Type: application/json; charset=utf-8  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
  
{  
    "OneToManyRelationships": [  
        {  
            "SchemaName": "new_media_new_book",  
            "ReferencedEntity": "new_book",  
            "ReferencingEntity": "new_media"  
        },  
        {  
            "SchemaName": "new_media_new_video",  
            "ReferencedEntity": "new_video",  
            "ReferencingEntity": "new_media"  
        },  
        {  
            "SchemaName": "new_media_new_audio",  
            "ReferencedEntity": "new_audio",  
            "ReferencingEntity": "new_media",  
            "CascadeConfiguration": {  
                "Assign": "NoCascade",  
                "Delete": "RemoveLink",  
                "Merge": "NoCascade",  
                "Reparent": "NoCascade",  
                "Share": "NoCascade",  
                "Unshare": "NoCascade"  
            }  
        }  
    ],  
  
    "Lookup": {  
        "AttributeType": "Lookup",  
        "AttributeName": {  
            "Value": "LookupType"  
        },  
  
        "Description": {  
            "@odata.type": "Microsoft.Dynamics.CRM.Label",  
            "LocalizedLabels": [  
                {  
                    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",  
                    "Label": "Media Polymorphic Lookup",  
                    "LanguageCode": 1033  
                }  
            ],  
  
            "UserLocalizedLabel": {  
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",  
                "Label": " Media Polymorphic Lookup Attribute",  
            }  
        }  
    }  
}
```

```

        "LanguageCode": 1033
    }
},
"DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "MediaPolymorphicLookup",
            "LanguageCode": 1033
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "MediaPolymorphicLookup",
        "LanguageCode": 1033
    }
},
"SchemaName": "new_mediaPolymorphicLookup",
"@odata.type": "Microsoft.Dynamics.CRM.ComplexLookupAttributeMetadata"
}
}

```

La respuesta de la publicación HTTP se muestra a continuación y contiene el ID del atributo polimórfico y todas las relaciones creadas.

JSON

```
{
    "@odata.context":
        "http://<organization
URL>/api/data/v9.1/$metadata#Microsoft.Dynamics.CRM.CreatePolymorphicLookupA
ttributeResponse",

    "RelationshipIds": [
        "77d4c6e9-0397-eb11-a81c-000d3a6cfaba",
        "7ed4c6e9-0397-eb11-a81c-000d3a6cfaba",
        "85d4c6e9-0397-eb11-a81c-000d3a6cfaba"
    ],
    "AttributeId": "d378dd3e-42f4-4bd7-95c7-0ee546c7de40"
}
```

Utilice las API de búsqueda de varias tablas

La siguiente tabla enumera las operaciones relevantes para las definiciones de tabla y atributo.

Operación (method)	Descripción	Formato de URL
Create (POST)	Nueva API	[OrganizationUrl]/api/data/v9.2 /CreatePolymorphicLookupAttribute
Atributo Recuperar (GET)	API existentes	[OrganizationUrl]/api/data/v9.2 /EntityDefinitions(<EntityId>)/Attributes(<Attributeld>)
Relación Recuperar (GET)	API existentes	[OrganizationUrl]/api/data/v9.2 /RelationshipDefinitions(<RelationshipId>)
Agregar relación (POST)	Agregar una relación a una existente búsqueda polimórfica atributo	[OrganizationUrl]/api/data/v9.2 /RelationshipDefinitions
Relación Quitar (DELETE)	API existentes	[OrganizationUrl]/api/data/v9.2 /RelationshipDefinitions(<RelationshipId>)
Quitar atributo (DELETE)	API existentes	[OrganizationUrl]/api/data/v9.2 /EntityDefinitions(<EntityId>)/Attributes(<Attributeld>)

La siguiente tabla enumera las operaciones relevantes para los datos de tabla y atributo.

Operation (method)	Descripción	Formato de URL
Create (POST)	Vea el ejemplo <code>new_checkouts</code> siguiente	[OrganizationUrl]/api/data/v9.2 /<entitySetName>
Retrieve (GET)	Agregue el siguiente encabezado para obtener anotaciones: Tipo de contenido: aplicación/json Preferir: odata.include-annotations = "*"	[OrganizationUrl]/api/data/v9.2 /<entitySetName> (<recordId>)

A continuación se muestra una solicitud de ejemplo que crea un nuevo conjunto de entidades con dos filas.

HTTP

```
POST [OrganizationUrl]/api/data/v9.1/new_checkouts
```

HTTP

```
{
  "new_name": "c1",
  new_CheckedOutItem_new_book@odata.bind: "/new_books(387a2c9b-ecc6-ea11-a81e-000d3af68bd7)"
}

{
  "new_name": "c2",
  new_CheckedOutItem_new_device@odata.bind: "/new_devices(6472e7ba-ecc6-ea11-a81e-000d3af68bd7)"
}
```

Crear búsqueda polimórfica (ejemplo de carga)

HTTP

```
POST [OrganizationUrl]/api/data/v9.2/CreatePolymorphicLookupAttribute
```

HTTP

```
{
  "OneToManyRelationships": [
    {
      "SchemaName": "new_checkout_poly_new_book",
      "ReferencedEntity": "new_book",
      "ReferencingEntity": "new_checkout"
    },
    {
      "SchemaName": "new_checkout_poly_new_device",
      "ReferencedEntity": "new_device",
      "ReferencingEntity": "new_checkout"
    },
    {
      "SchemaName": "new_checkout_poly_new_dvd",
      "ReferencedEntity": "new_dvd",
      "ReferencingEntity": "new_checkout",
      "CascadeConfiguration": {
        "Assign": "NoCascade",
        "Delete": "RemoveLink",
        "Merge": "NoCascade",
        "Reparent": "NoCascade",
        "Share": "NoCascade",
        "Unshare": "NoCascade"
      }
    }
  ],
  "Lookup": {
    "AttributeType": "Lookup",
    "AttributeTypeName": {
      "Value": "LookupType"
    }
  }
}
```

```

},
"Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Checkedout item Polymorphic Lookup Attribute",
            "LanguageCode": 1033
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Checkedout item Polymorphic Lookup Attribute",
        "LanguageCode": 1033
    }
},
"DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Checkedout item",
            "LanguageCode": 1033
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Checkedout item",
        "LanguageCode": 1033
    }
},
"SchemaName": "new_CheckedoutItem",
"@odata.type": "Microsoft.Dynamics.CRM.ComplexLookupAttributeMetadata"
}
}

```

Agregar relación a la búsqueda polimórfica existente (ejemplo de carga)

HTTP

POST [OrganizationUrl]/api/data/v9.2/RelationshipDefinitions

HTTP

```
{
    "SchemaName": "new_checkout_poly_new_researchresource",
    "@odata.type": "Microsoft.Dynamics.CRM.OneToManyRelationshipMetadata",
    "CascadeConfiguration": {
        "Assign": "NoCascade",
        "Delete": "NoCascade"
    }
}
```

```

    "Delete": "RemoveLink",
    "Merge": "NoCascade",
    "Reparent": "NoCascade",
    "Share": "NoCascade",
    "Unshare": "NoCascade"
},
"ReferencedEntity": "new_researchresource",
"ReferencingEntity": "new_checkout",
"Lookup": {
    "AttributeType": "Lookup",
    "AttributeTypeName": { "Value": "LookupType" },
    "Description": {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [
            {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Checkout Polymorphic Lookup Attribute",
                "LanguageCode": 1033
            }
        ],
        "UserLocalizedLabel": {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Checkout Polymorphic Lookup Attribute",
            "LanguageCode": 1033
        }
    },
    "DisplayName": {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [
            {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Checkout item",
                "LanguageCode": 1033
            }
        ],
        "UserLocalizedLabel": {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Checkout item",
            "LanguageCode": 1033
        }
    },
    "SchemaName": "new_CheckedoutItem",
    "@odata.type": "Microsoft.Dynamics.CRM.LookupAttributeMetadata"
}
}

```

Consulte también

[Utilizar la API web con definiciones de tabla](#)

[Crear y actualizar relaciones de tablas](#)

[Consultar definiciones de tabla con la API web](#)

[Recuperar definiciones de tablas por nombre o Metadatald](#)

[Modelar tablas y columnas usando la API web](#)

[Ejemplo de operaciones de metadatos de API web](#)

[Ejemplo de operaciones de metadatos de API web \(C#\)](#)

Crear y actualizar selecciones (conjuntos de opciones) mediante la API web

Artículo • 25/08/2023

Normalmente, usa los conjuntos de opciones *global* para establecer columnas de tabla de modo que diferentes columnas puedan compartir el mismo conjunto de opciones, que se mantienen en una ubicación. A diferencia de los conjuntos de opciones *local* que se definen solo para una columna específica, puede reutilizar conjuntos de opciones globales. Estos valores también se usan en los parámetros de solicitudes de manera similar a una enumeración.

ⓘ Nota

Solo puede cambiar un conjunto de opciones administrado existente si es el editor. Para realizar un cambio, como la opción de cambio de nombre o la opción de eliminación, en estos conjuntos de opciones, se debe realizar una actualización a la solución que agregó el conjunto de opciones. Más información: [Actualizar una solución](#)

Cuando define un conjunto de opciones global mediante una solicitud `POST` en `[URI de organización]/api/data/v9.2/GlobalOptionSetDefinitions`, se recomienda dejar que el sistema asigne un valor. Deje que el sistema asigne el valor pasando un valor `null` cuando cree la nueva instancia `OptionMetadata`. Al definir una opción, contiene un determinada prefijo de valor de opción específico del contexto del editor establecido para la solución en la que se crea el conjunto de opciones. Este prefijo ayuda a reducir la posibilidad de crear conjuntos de opciones duplicados para una solución administrada y en cualquier conjunto de opciones que se definan en entornos donde está instalada la solución administrada. Para obtener más información, consulte [Combinar opciones del conjunto de opciones](#).

Mensajes

La siguiente tabla enumera los mensajes que se pueden usar con los conjuntos de opciones globales.

Publicación	Operación de API web
<code>CreateOptionSet</code>	Use la solicitud <code>POST</code> para <code>URI de organización/api/data/v9.2/GlobalOptionSetDefinitions</code> .

Publicación	Operación de API web
DeleteOptionSet	Use la solicitud <code>DELETE</code> para <i>URI de organización</i> /api/data/v9.2/GlobalOptionSetDefinitions(Name='<name>').
RetrieveAllOptionSets	Use la solicitud <code>GET</code> para <i>URI de organización</i> /api/data/v9.2/GlobalOptionSetDefinitions.
RetrieveOptionSet	Use la solicitud <code>GET</code> para <i>URI de organización</i> /api/data/v9.2/GlobalOptionSetDefinitions(Name='<name>').

La siguiente tabla enumera los mensajes que se pueden usar con los conjuntos de opciones globales y locales.

Publicación	Operación de API web
DeleteOptionValue	Acción DeleteOptionValue
Elimina uno de los valores de un conjunto de opciones globales.	Ejemplo: Opción de eliminación
InsertOptionValue	Acción InsertOptionValue
Inserta una opción en un conjunto de opciones globales.	Ejemplo: Opciones de inserción
InsertStatusValue	Acción InsertStatusValue
Inserta una nueva opción en el conjunto de opciones global usado en la columna <code>Status</code> .	Ejemplo: Insertar un nuevo valor de estado
OrderOption	Acción OrderOption
Cambia el orden relativo de las opciones de un conjunto de opciones.	Ejemplo: Opciones de pedido
UpdateOptionSet	Use la solicitud <code>PUT</code> con <code>OptionSetMetadataBase EntityType</code> para <i>[Organization URI] /api/data/v9.2/GlobalOptionSetDefinitions(metadataid)</i> Solo aquellas propiedades definidas por <code>OptionSetMetadataBase</code> se pueden actualizar. Estas propiedades no incluyen las opciones. Utilice otras acciones para realizar cambios en las opciones.
UpdateOptionValue	Acción UpdateOptionValue
Actualiza una opción	Ejemplo: Opciones de actualización

Publicación	Operación de API web
de un conjunto de opciones.	
UpdateStateValue Inserta una nueva opción en el conjunto de opciones usado en la columna <code>Status</code> .	Acción UpdateStateValue

Ejemplos

- Creación de un conjunto de opciones global
- Cree una columna de elección usando un conjunto de opciones global
- Insertar opciones
- Opciones de actualización
- Ordenar opciones
- Opción Eliminar
- Insertar valor de estado

Creación de un conjunto de opciones global

El siguiente ejemplo usará estas propiedades para crear una opción global.

Propiedades de OptionSetMetadata	Valores
<code>Name</code>	<code>sample_colors</code>
<code>DisplayName</code>	Colores
<code>Description</code>	Opción de color
<code>OptionSetType</code>	<code>Picklist</code>
<code>Options</code>	<code>valor: 727000000, etiqueta:Rojo</code> <code>valor: 727000001, etiqueta:Amarillo</code> <code>valor: 727000002, etiqueta:Verde</code>

El siguiente ejemplo crea una opción global usando las propiedades.

El URI para la opción global se devuelve en la respuesta. También puede hacer referencia a esta opción global con el nombre:

```
GlobalOptionSetDefinitions(Name='sample_colors').
```

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/GlobalOptionSetDefinitions
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
Content-Type: application/json; charset=utf-8
Content-Length: 2769

{
    "@odata.type": "Microsoft.Dynamics.CRM.OptionSetMetadata",
    "Options": [
        {
            "Value": 727000000,
            "Label": {
                "@odata.type": "Microsoft.Dynamics.CRM.Label",
                "LocalizedLabels": [
                    {
                        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                        "Label": "Red",
                        "LanguageCode": 1033,
                        "IsManaged": false
                    }
                ],
                "UserLocalizedLabel": {
                    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                    "Label": "Red",
                    "LanguageCode": 1033,
                    "IsManaged": false
                }
            }
        },
        {
            "Value": 727000001,
            "Label": {
                "@odata.type": "Microsoft.Dynamics.CRM.Label",
                "LocalizedLabels": [
                    {
                        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                        "Label": "Yellow",
                        "LanguageCode": 1033,
                        "IsManaged": false
                    }
                ],
                "UserLocalizedLabel": {
                    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                    "Label": "Yellow",
                    "LanguageCode": 1033,
                    "IsManaged": false
                }
            }
        }
    ]
}
```

```
        }
    },
{
    "Value": 727000002,
    "Label": {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [
            {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Green",
                "LanguageCode": 1033,
                "IsManaged": false
            }
        ],
        "UserLocalizedLabel": {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Green",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    }
},
"Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Color Choice",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Color Choice",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Colors",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Colors",
        "LanguageCode": 1033,
        "IsManaged": false
    }
}
```

```
        },
        "Name": "sample_colors",
        "OptionsetType": "Picklist"
    }
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/GlobalOptionSetDefinitions(1d733907-4d05-ee11-8f6e-
000d3a993550)
```

Cree una columna de elección usando un conjunto de opciones global

El siguiente ejemplo usará estas propiedades para crear una columna de elección usando una opción global.

Propiedades de atributo de lista de selección	Valores
SchemaName	sample_Colors
DisplayName	Colores de muestra
Description	Atributo de lista de selección global de colores
RequiredLevel	None
GlobalOptionSet	Esta propiedad de navegación de un solo valor debe establecerse utilizando la sintaxis <code>@odata.bind</code> con una referencia a la opción global. Este ejemplo usa <code>MetadataId</code> como clave, pero también podría usar clave alternativa con <code>Name: GlobalOptionSetDefinitions(Name='sample_colors')</code>

El siguiente ejemplo crea una columna local usando las propiedades y la agrega a la tabla `sample_bankaccount`.

El URI para el atributo se devuelve en la respuesta.

Solicitud:

HTTP

```
POST [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/Attributes
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
Content-Type: application/json; charset=utf-8
Content-Length: 1465

{
    "@odata.type": "Microsoft.Dynamics.CRM.PicklistAttributeMetadata",
    "AttributeType": "Picklist",
    "AttributeName": {
        "Value": "PicklistType"
    },
    "SourceTypeMask": 0,
    "GlobalOptionSet@odata.bind": "/GlobalOptionSetDefinitions(1d733907-4d05-
ee11-8f6e-000d3a993550)",
    "Description": {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [
            {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Colors Global Picklist Attribute",
                "LanguageCode": 1033,
                "IsManaged": false
            }
        ],
        "UserLocalizedLabel": {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Colors Global Picklist Attribute",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    },
    "DisplayName": {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [
            {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Sample Colors",
                "LanguageCode": 1033,
                "IsManaged": false
            }
        ],
        "UserLocalizedLabel": {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Sample Colors",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    },
}
```

```
"RequiredLevel": {  
    "Value": "None",  
    "CanBeChanged": false,  
    "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"  
},  
"SchemaName": "sample_Colors"  
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent  
OData-Version: 4.0  
OData-EntityId: [Organization  
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/Attri  
butes(25733907-4d05-ee11-8f6e-000d3a993550)
```

Insertar opciones

El siguiente ejemplo usa la acción [InsertOptionValue](#) para agregar una nueva opción con el valor `727000005` y la etiqueta **Echo** a la columna de elección local creada por [Cree una columna de elección](#).

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/InsertOptionValue  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json  
Content-Type: application/json; charset=utf-8  
Content-Length: 612  
  
{  
    "AttributeLogicalName": "sample_choice",  
    "EntityLogicalName": "sample_bankaccount",  
    "Value": 727000005,  
    "Label": {  
        "@odata.type": "Microsoft.Dynamics.CRM.Label",  
        "LocalizedLabels": [  
            {  
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",  
                "Label": "Echo",  
                "LanguageCode": 1033,  
                "IsManaged": false  
            }  
        ]  
    }  
}
```

```
],
  "UserLocalizedLabel": {
    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
    "Label": "Echo",
    "LanguageCode": 1033,
    "IsManaged": false
  }
},
"SolutionUniqueName": "examplesolution"
}
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.InsertOptionValueRespons
e",
  "NewOptionValue": 727000005
}
```

Opciones de actualización

Para actualizar opciones individuales, debe usar la [acción Actualizar valor de opción](#). El siguiente ejemplo actualiza `TrueOption` del ejemplo de columna booleana en [Crear una columna booleana](#) y cambia la etiqueta para que sea `Up` en lugar de `True`. Debido a que este es un conjunto de opciones 'local', usa `AttributeLogicalName` y `EntityLogicalName`. Para un conjunto de opciones globales, use el parámetro `OptionSetName` en su lugar.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/UpdateOptionValue HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
  "AttributeLogicalName": "new_boolean",
  "EntityLogicalName": "new_bankaccount",
  "Value": 1,
  "Label": {
```

```
"@odata.type": "Microsoft.Dynamics.CRM.Label",
"LocalizedLabels": [
  {
    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
    "Label": "Up",
    "LanguageCode": 1033,
    "IsManaged": false
  }
],
"UserLocalizedLabel": {
  "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
  "Label": "Up",
  "LanguageCode": 1033,
  "IsManaged": false
},
"MergeLabels": true
}
```

Respuesta:

HTTP

HTTP/1.1 204 NoContent

OData-Version: 4.0

Ordenar opciones

Este ejemplo siguiente muestra cómo reordenar las opciones en un conjunto de opciones local con la acción `OrderOption`. La propiedad `Value` contiene los valores de la opción en el orden deseado.

Para usar esto con un conjunto de opciones global, especifique el parámetro

`OptionSetName` en lugar de `EntityLogicalName` y `AttributeLogicalName`.

El parámetro `SolutionUniqueName` aplica los cambios como parte de la solución especificada.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/OrderOption
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
Content-Type: application/json; charset=utf-8
```

```
Content-Length: 253

{
  "EntityLogicalName": "sample_bankaccount",
  "AttributeLogicalName": "sample_choice",
  "Values": [
    727000002,
    727000000,
    727000003,
    727000001,
    727000005,
    727000004
  ],
  "SolutionUniqueName": "examplesolution"
}
```

Respuesta:

HTTP

HTTP/1.1 204 NoContent

OData-Version: 4.0

Opción Eliminar

El siguiente ejemplo muestra cómo eliminar una opción para una columna de opción local mediante la acción [DeleteOptionValue](#).

Para usar esto con un conjunto de opciones global, especifique el parámetro `OptionSetName` en lugar de `EntityLogicalName` y `AttributeLogicalName`.

El parámetro `SolutionUniqueName` aplica los cambios como parte de la solución especificada.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/DeleteOptionValue
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
Content-Type: application/json; charset=utf-8
Content-Length: 116

{
  "AttributeLogicalName": "sample_choice",
```

```
    "EntityLogicalName": "sample_bankaccount",
    "Value": 727000004
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
```

Insertar valor de estado

El siguiente ejemplo muestra cómo agregar una opción para una columna de estado mediante la acción [InsertStatusValue](#).

Utilice el parámetro `StateCode` para especificar a qué opción de código de estado se aplica el valor de estado. El parámetro `SolutionUniqueName` aplica los cambios como parte de la solución especificada.

La propiedad `NewOptionValue` devuelta por [InsertStatusValueResponse ComplexType](#) contiene el valor asignado a la opción.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/InsertStatusValue
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
Content-Type: application/json; charset=utf-8
Content-Length: 609

{
    "AttributeLogicalName": "statuscode",
    "EntityLogicalName": "sample_bankaccount",
    "Label": {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [
            {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Frozen",
                "LanguageCode": 1033,
                "IsManaged": false
            }
        ],
    }
}
```

```
"UserLocalizedLabel": {  
    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",  
    "Label": "Frozen",  
    "LanguageCode": 1033,  
    "IsManaged": false  
},  
    "StateCode": 1,  
    "SolutionUniqueName": "examplesolution"  
}
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK  
OData-Version: 4.0  
  
{  
    "@odata.context": "[Organization  
Uri]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.InsertStatusValueRespons  
e",  
    "NewOptionValue": 727000000  
}
```

Consulte también

[Personalizar opciones](#)

[Información general para crear y editar opciones globales](#)

[Crear una opción](#)

ⓘ Nota

¿Puede indicarnos sus preferencias de idioma de documentación? [Realice una breve encuesta.](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad](#)).

Ejemplo de operaciones de metadatos de API web

Artículo • 24/08/2023

Esta colección de solicitudes y respuestas http demuestra cómo realizar operaciones seleccionadas que modifican el esquema de Dataverse, o metadatos, usando la [muestra de operaciones de metadatos de API web \(C#\)](#)

En este artículo se describe un conjunto común de operaciones implementadas por cada ejemplo en este grupo. Este artículo describe las solicitudes y respuestas HTTP y la salida de texto que realiza cada muestra sin los detalles específicos del idioma. Consulte las descripciones específicas del idioma y los ejemplos individuales para obtener más información acerca de cómo se realizan estas operaciones.

Demostraciones

Este ejemplo se divide en las siguientes secciones, que contienen operaciones de datos de consulta de la API web de Dataverse que se describen minuciosamente en los artículos conceptuales asociados especificados.

Sección de código	Artículos conceptuales y de referencia asociados
Sección 0: Crear un editor y una solución	Crear una fila de tabla publisher EntityType solucion EntityType
Sección 1: Crear, recuperar y actualizar una tabla	Crear y actualizar definiciones de tablas EntityMetadata EntityType
Sección 2: Crear, recuperar y actualizar columnas <ul style="list-style-type: none">- Columna booleana<ul style="list-style-type: none">- Valores de opción de actualización- Columna DateTime- Columna decimal- Columnas de enteros- Columna de notas- Columna de moneda- Columna de lista de selección<ul style="list-style-type: none">- Agregar una opción al conjunto de opciones local<ul style="list-style-type: none">- Opciones para reordenar la columna de opciones- Eliminar valor de opción local	Crear columnas Recuperar atributos Acción InsertOptionValue Acción OrderOption Acción DeleteOptionValue Acción InsertStatusValue

Sección de código	Artículos conceptuales y de referencia asociados
<ul style="list-style-type: none"> - Columna de lista de selección múltiple - Insertar valor de estado 	
Sección 3: Crear y usar OptionSet global	Crear y actualizar opciones (conjuntos de opciones)
Sección 4: Crear relación con el cliente	CreateCustomerRelationships Action
Sección 5: Crear y recuperar una relación de uno a muchos	Idoneidad para relaciones Crear nueva relación de uno a varios Consultar metadatos de relaciones
Sección 6: Crear y recuperar una relación de muchos a uno	Crear nueva relación de uno a varios Consultar metadatos de relaciones
Sección 7: Crear y recuperar una relación de muchos a muchos	Crear nueva relación de varios a varios Consultar metadatos de relaciones
Sección 8: Exportar una solución administrada	Exportar soluciones
Sección 9: Eliminar registros de muestra	Eliminación básica
Sección 10: Importar y eliminar una solución administrada	Importar soluciones

! Nota

Para razones de brevedad, se han omitido los encabezados HTTP menos pertinentes. Las URL de los registros variarán con la dirección de la organización base y los id. establecidos por el servidor de Dataverse.

Sección 0: Crear un editor y una solución

1. Cree primero el editor, ya que la solución debe estar relacionada con él. Todos los elementos creados o modificados en esta muestra utilizan los valores Publisher `customizationprefix` y `customizationoptionvalueprefix`.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/publishers HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
```

```
If-None-Match: null  
Accept: application/json  
  
{  
  "friendlyname": "Example Publisher",  
  "uniquename": "examplepublisher",  
  "description": "An example publisher for samples",  
  "customizationprefix": "sample",  
  "customizationoptionvalueprefix": 72700  
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent  
OData-Version: 4.0  
OData-EntityId: [Organization Uri]/api/data/v9.2/publishers(a78ab7fc-102a-ed11-9db1-00224804f8e2)
```

Salida de la consola:

```
Created publisher Example Publisher
```

2. A continuación, cree la solución relacionada con el editor.

(!) Nota

Muchos de los elementos creados o actualizados en este ejemplo utilizarán el valor `uniquename` de esta solución con el encabezado de solicitud `MSCRM.SolutionUniqueName` para que los cambios se incluyan como parte de esta solución. Algunas acciones tienen un parámetro `SolutionUniqueName` que hace lo mismo. Al final de este ejemplo, esta solución se exportará y contendrá las definiciones de todos los elementos creados y modificados en este ejemplo.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/solutions HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0
```

```
If-None-Match: null  
Accept: application/json  
  
{  
  "friendlyname": "Example Solution",  
  "uniquename": "examplesolution",  
  "description": "An example solution for samples",  
  "version": "1.0.0.0",  
  "publisherid@odata.bind": "publishers(a78ab7fc-102a-ed11-9db1-00224804f8e2)"  
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent  
OData-Version: 4.0  
OData-EntityId: [Organization Uri]/api/data/v9.2/solutions(5472b902-112a-ed11-9db1-00224804f8e2)
```

Salida de la consola:

```
Created solution Example Solution
```

Sección 1: Crear, recuperar y actualizar una tabla

1. Cree la tabla `sample_BankAccount`.

Estas propiedades son requeridas: `SchemaName`, `DisplayName`, `DisplayCollectionName`, `HasNotes`, `HasActivities` y `PrimaryNameAttribute`, que deben incluir el valor `LogicalName` de la columna de nombre principal.

La tabla también debe incluir una columna `StringAttributeMetadata` en la colección `Attributes` que sea la columna de nombre principal de la tabla. Esa definición de columna debe tener valores `SchemaName`, `MaxLength` y `DisplayName`, y `IsPrimaryName` debe establecerse en verdadero.

ⓘ Nota

El encabezado de solicitud `MSCRM.SolutionUniqueName: examplesolution` asocia esta tabla a la solución. El valor `SchemaName` (`sample_BankAccount`) incluye el prefijo de personalización del editor.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/EntityDefinitions HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
  "@odata.type": "Microsoft.Dynamics.CRM.EntityMetadata",
  "Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
      {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "A table to store information about customer bank
accounts",
        "LanguageCode": 1033,
        "IsManaged": false
      }
    ],
    "UserLocalizedLabel": {
      "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
      "Label": "A table to store information about customer bank
accounts",
      "LanguageCode": 1033,
      "IsManaged": false
    }
  },
  "DisplayCollectionName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
      {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Bank Accounts",
        "LanguageCode": 1033,
        "IsManaged": false
      }
    ],
    "UserLocalizedLabel": {
      "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
      "Label": "Bank Accounts",
      "LanguageCode": 1033,
      "IsManaged": false
    }
  }
}
```

```
},
"DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Bank Account",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Bank Account",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"HasActivities": false,
"HasNotes": false,
"OwnershipType": "UserOwned",
"PrimaryNameAttribute": "sample_name",
"SchemaName": "sample_BankAccount",
"Attributes": [
    {
        "@odata.type": "Microsoft.Dynamics.CRM.StringAttributeMetadata",
        "AttributeType": "String",
        "AttributeTypeName": {
            "Value": "StringType"
        },
        "MaxLength": 100,
        "Description": {
            "@odata.type": "Microsoft.Dynamics.CRM.Label",
            "LocalizedLabels": [
                {
                    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                    "Label": "The primary attribute for the Bank Account entity.",
                    "LanguageCode": 1033,
                    "IsManaged": false
                }
            ],
            "UserLocalizedLabel": {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "The primary attribute for the Bank Account entity.",
                "LanguageCode": 1033,
                "IsManaged": false
            }
        }
    },
    {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [
            {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Account Name",
            }
        ]
    }
],
```

```

        "LanguageCode": 1033,
        "IsManaged": false
    },
],
"UserLocalizedLabel": {
    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
    "Label": "Account Name",
    "LanguageCode": 1033,
    "IsManaged": false
}
},
"IsPrimaryName": true,
"RequiredLevel": {
    "Value": "None",
    "CanBeChanged": false,
    "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
},
"SchemaName": "sample_Name"
}
]
}

```

Respuesta:

HTTP

```

HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/EntityDefinitions(5872b902-112a-ed11-9db1-
00224804f8e2)

```

Salida de la consola:

```

Sending request to create the sample_BankAccount table...
Created sample_BankAccount table.

```

2. Recupere la definición de tabla `sample_BankAccount`.

- Esta operación de recuperación no incluye ninguna `$select` para filtrar las propiedades devueltas porque estos datos se modifican y se envían para actualizar la definición de la tabla usando `PUT`, que sobrescribe el valor existente.
- Esta consulta tampoco incluye `$expand` para incluir datos relacionados, como atributos, porque los datos relacionados deben actualizarse por separado.

(!) Nota

Esta solicitud y otras en esta muestra usan el encabezado `Consistency: Strong`. Utilice este encabezado cuando recupere los cambios de definición de metadatos directamente después de aplicarlos. Los cambios de metadatos se almacenan en caché por razones de rendimiento y una solicitud de un elemento recién creado puede devolver un 404 porque aún no se ha almacenado en caché. El almacenamiento en caché puede tardar 30 segundos. Este encabezado obligará al servidor a leer la última versión, incluidos sus cambios. Al usar este encabezado, niega la ganancia de rendimiento que proporciona el almacenamiento en caché, por lo que solo debe usarlo en escenarios como este ejemplo donde está recuperando los cambios que acaba de realizar. Más información: [Encabezados HTTP > Otros encabezados](#).

Solicitud:

HTTP

```
GET [Organization  
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')  
HTTP/1.1  
Consistency: Strong  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK  
OData-Version: 4.0  
  
{  
"@odata.context": "[Organization  
Uri]/api/data/v9.2/$metadata#EntityDefinitions/$entity",  
"ActivityTypeMask": 0,  
"AutoRouteToOwnerQueue": false,  
"CanTriggerWorkflow": true,  
"EntityHelpUrlEnabled": false,  
"EntityHelpUrl": null,  
"IsDocumentManagementEnabled": false,  
"IsOneNoteIntegrationEnabled": false,  
"IsInteractionCentricEnabled": false,
```

```
"IsKnowledgeManagementEnabled": false,
"IsSLAEnabled": false,
"IsBPFEntity": false,
"IsDocumentRecommendationsEnabled": false,
"IsMSTeamsIntegrationEnabled": false,
"SettingOf": null,
"DataProviderId": null,
"DataSourceId": null,
"AutoCreateAccessTeams": false,
"IsActivity": false,
"IsActivityParty": false,
"IsRetrieveAuditEnabled": false,
"IsRetrieveMultipleAuditEnabled": false,
"IsArchivalEnabled": false,
"IsAvailableOffline": false,
"IsChildEntity": false,
"IsAIRUpdated": false,
"IconLargeName": null,
"IconMediumName": null,
"IconSmallName": null,
"IconVectorName": null,
"IsCustomEntity": true,
"IsBusinessProcessEnabled": false,
"SyncToExternalSearchIndex": false,
"IsOptimisticConcurrencyEnabled": true,
"ChangeTrackingEnabled": false,
"IsImportable": true,
"IsIntersect": false,
"IsManaged": false,
".IsEnabledForCharts": true,
".IsEnabledForTrace": false,
"IsValidForAdvancedFind": true,
"DaysSinceRecordLastModified": 0,
"MobileOfflineFilters": "",
"IsReadingPaneEnabled": true,
"IsQuickCreateEnabled": false,
"LogicalName": "sample_bankaccount",
"ObjectTypeCode": 10393,
"OwnershipType": "UserOwned",
"PrimaryNameAttribute": "sample_name",
"PrimaryImageAttribute": null,
"PrimaryIdAttribute": "sample_bankaccountid",
"RecurrenceBaseEntityLogicalName": null,
"ReportViewName": "Filteredsample_BankAccount",
"SchemaName": "sample_BankAccount",
"IntroducedVersion": "1.0.0.0",
"IsStateModelAware": true,
"EnforceStateTransitions": false,
"ExternalName": null,
"EntityColor": null,
"LogicalCollectionName": "sample_bankaccounts",
"ExternalCollectionName": null,
"CollectionSchemaName": "sample_BankAccounts",
"EntitySetName": "sample_bankaccounts",
".IsEnabledForExternalChannels": false,
```

```
"IsPrivate": false,
"UsesBusinessDataLabelTable": false,
"IsLogicalEntity": false,
"HasNotes": false,
"HasActivities": false,
"HasFeedback": false,
"IsSolutionAware": false,
"CreatedOn": "2022-09-01T16:13:40Z",
"ModifiedOn": "2022-09-01T16:13:40Z",
"HasEmailAddresses": false,
"OwnerId": null,
"OwnerIdType": 8,
"OwningBusinessUnit": null,
"MetadataId": "5872b902-112a-ed11-9db1-00224804f8e2",
"HasChanged": null,
"Description": {
    "LocalizedLabels": [
        {
            "Label": "A table to store information about customer bank accounts",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "daf026b7-dfde-4b7b-8e52-91f31b098a9d",
            "HasChanged": null
        }
    ],
    "UserLocalizedLabel": {
        "Label": "A table to store information about customer bank accounts",
        "LanguageCode": 1033,
        "IsManaged": false,
        "MetadataId": "daf026b7-dfde-4b7b-8e52-91f31b098a9d",
        "HasChanged": null
    }
},
"DisplayCollectionName": {
    "LocalizedLabels": [
        {
            "Label": "Bank Accounts",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "5c598c79-b89d-4679-8a40-9562d0a1e4fb",
            "HasChanged": null
        }
    ],
    "UserLocalizedLabel": {
        "Label": "Bank Accounts",
        "LanguageCode": 1033,
        "IsManaged": false,
        "MetadataId": "5c598c79-b89d-4679-8a40-9562d0a1e4fb",
        "HasChanged": null
    }
},
"DisplayName": {
    "LocalizedLabels": [
```

```
{
    "Label": "Bank Account",
    "LanguageCode": 1033,
    "IsManaged": false,
    "MetadataId": "4e4c3fdc-7711-4b43-8eba-9155bb7100c0",
    "HasChanged": null
}
],
"UserLocalizedLabel": {
    "Label": "Bank Account",
    "LanguageCode": 1033,
    "IsManaged": false,
    "MetadataId": "4e4c3fdc-7711-4b43-8eba-9155bb7100c0",
    "HasChanged": null
}
},
"IsAuditEnabled": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyauditsettings"
},
"IsValidForQueue": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyqueuesettings"
},
"IsConnectionsEnabled": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyconnectionsettings"
},
"IsCustomizable": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "iscustomizable"
},
"IsRenameable": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "isrenameable"
},
"IsMappable": {
    "Value": true,
    "CanBeChanged": false,
    "ManagedPropertyLogicalName": "ismappable"
},
"IsDuplicateDetectionEnabled": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifieduplicatedetectionsettings"
},
"CanCreateAttributes": {
    "Value": true,
    "CanBeChanged": false,
    "ManagedPropertyLogicalName": "cancreateattributes"
```

```
},
  "CanCreateForms": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "cancreateforms"
  },
  "CanCreateViews": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "cancreateviews"
  },
  "CanCreateCharts": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "cancreatecharts"
  },
  "CanBeRelatedEntityInRelationship": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canberelatedentityinrelationship"
  },
  "CanBePrimaryEntityInRelationship": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canbeprimaryentityinrelationship"
  },
  "CanBeInManyToMany": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canbeinmanytomany"
  },
  "CanBeInCustomEntityAssociation": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canbeincustomentityassociation"
  },
  "CanEnableSyncToExternalSearchIndex": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canenablesynctoexternalsearchindex"
  },
  "CanModifyAdditionalSettings": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyadditionalsettings"
  },
  "CanChangeHierarchicalRelationship": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canchangehierarchicalrelationship"
  },
  "CanChangeTrackingBeEnabled": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canchangetrackingbeenabled"
```

```
},
  "IsMailMergeEnabled": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifymailmergesettings"
  },
  "IsVisibleInMobile": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifymobilevisibility"
  },
  "IsVisibleInMobileClient": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifymobileclientvisibility"
  },
  "IsReadOnlyInMobileClient": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifymobileclientreadonly"
  },
  "IsOfflineInMobileClient": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifymobileclientoffline"
  },
  "Privileges": [
    {
      "CanBeBasic": true,
      "CanBeDeep": true,
      "CanBeGlobal": true,
      "CanBeLocal": true,
      "CanBeEntityReference": false,
      "CanBeParentEntityReference": false,
      "Name": "prvCreatesample_BankAccount",
      "PrivilegeId": "44f00701-716e-4584-8bab-cb0d263c070b",
      "PrivilegeType": "Create"
    },
    {
      "CanBeBasic": true,
      "CanBeDeep": true,
      "CanBeGlobal": true,
      "CanBeLocal": true,
      "CanBeEntityReference": false,
      "CanBeParentEntityReference": false,
      "Name": "prvReadsample_BankAccount",
      "PrivilegeId": "9cad3243-d0fe-467e-a731-c8b3416a6252",
      "PrivilegeType": "Read"
    },
    {
      "CanBeBasic": true,
      "CanBeDeep": true,
      "CanBeGlobal": true,
      "CanBeLocal": true,
      "CanBeEntityReference": false,
```

```
        "CanBeParentEntityReference": false,
        "Name": "prvWritesample_BankAccount",
        "PrivilegeId": "dc5465ed-223f-4b13-a272-fff25e5b5270",
        "PrivilegeType": "Write"
    },
    {
        "CanBeBasic": true,
        "CanBeDeep": true,
        "CanBeGlobal": true,
        "CanBeLocal": true,
        "CanBeEntityReference": false,
        "CanBeParentEntityReference": false,
        "Name": "prvDeletesample_BankAccount",
        "PrivilegeId": "9a409df2-ca4a-4ad9-8218-df88424dd7a0",
        "PrivilegeType": "Delete"
    },
    {
        "CanBeBasic": true,
        "CanBeDeep": true,
        "CanBeGlobal": true,
        "CanBeLocal": true,
        "CanBeEntityReference": false,
        "CanBeParentEntityReference": false,
        "Name": "prvAssignsample_BankAccount",
        "PrivilegeId": "73bf7dd3-f532-4468-abfe-84bbf0eae058",
        "PrivilegeType": "Assign"
    },
    {
        "CanBeBasic": true,
        "CanBeDeep": true,
        "CanBeGlobal": true,
        "CanBeLocal": true,
        "CanBeEntityReference": false,
        "CanBeParentEntityReference": false,
        "Name": "prvSharesample_BankAccount",
        "PrivilegeId": "292f6e27-9603-4835-882d-e28c175432ed",
        "PrivilegeType": "Share"
    },
    {
        "CanBeBasic": true,
        "CanBeDeep": true,
        "CanBeGlobal": true,
        "CanBeLocal": true,
        "CanBeEntityReference": false,
        "CanBeParentEntityReference": false,
        "Name": "prvAppendsample_BankAccount",
        "PrivilegeId": "42401aa6-6447-4fdc-9679-bcb89b62bd76",
        "PrivilegeType": "Append"
    },
    {
        "CanBeBasic": true,
        "CanBeDeep": true,
        "CanBeGlobal": true,
        "CanBeLocal": true,
        "CanBeEntityReference": false,
```

```

        "CanBeParentEntityReference": false,
        "Name": "prvAppendTosample_BankAccount",
        "PrivilegeId": "847ba62d-2f33-4208-87e6-52532b331f60",
        "PrivilegeType": "AppendTo"
    }
],
"Settings": []
}

```

Salida de la consola:

El ejemplo muestra el JSON recuperado del servidor.

3. Actualice la tabla `sample_BankAccount`. Los únicos valores que se modifican son `HasActivities` y `Description`, pero debe enviar la definición completa con `PUT`.

Solicitud:

HTTP

```

PUT [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')
HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
MSCRM.MergeLabels: true
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
"@odata.type": "Microsoft.Dynamics.CRM.EntityMetadata",
"ActivityTypeMask": 0,
"AutoCreateAccessTeams": false,
"AutoRouteToOwnerQueue": false,
"CanBeInCustomEntityAssociation": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canbeincustomentityassociation"
},
"CanBeInManyToMany": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canbeinmanytomany"
},
"CanBePrimaryEntityInRelationship": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canbeprimaryentityinrelationship"
}
}

```

```
},
  "CanBeRelatedEntityInRelationship": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canberelatedentityinrelationship"
  },
  "CanChangeHierarchicalRelationship": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canchangehierarchicalrelationship"
  },
  "CanChangeTrackingBeEnabled": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canchangetrackingbeenabled"
  },
  "CanCreateAttributes": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": true,
    "CanBeChanged": false,
    "ManagedPropertyLogicalName": "cancreateattributes"
  },
  "CanCreateCharts": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "cancreatecharts"
  },
  "CanCreateForms": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "cancreateforms"
  },
  "CanCreateViews": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "cancreateviews"
  },
  "CanEnableSyncToExternalSearchIndex": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canenablesynctoexternalsearchindex"
  },
  "CanModifyAdditionalSettings": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyadditionalsettings"
  }
},
```

```
"CanTriggerWorkflow": true,
"ChangeTrackingEnabled": false,
"CollectionSchemaName": "sample_BankAccounts",
"DaysSinceRecordLastModified": 0,
"Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Contains information about customer bank accounts",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Contains information about customer bank accounts",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"DisplayCollectionName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Bank Accounts",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "5c598c79-b89d-4679-8a40-9562d0a1e4fb"
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Bank Accounts",
        "LanguageCode": 1033,
        "IsManaged": false,
        "MetadataId": "5c598c79-b89d-4679-8a40-9562d0a1e4fb"
    }
},
"DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Bank Account",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "4e4c3fdc-7711-4b43-8eba-9155bb7100c0"
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Bank Account",
        "LanguageCode": 1033,
```

```
        "IsManaged": false,
        "MetadataId": "4e4c3fdc-7711-4b43-8eba-9155bb7100c0"
    },
},
"EnforceStateTransitions": false,
"EntityHelpUrlEnabled": false,
"EntitySetName": "sample_bankaccounts",
"HasActivities": true,
"HasFeedback": false,
"HasNotes": false,
"IntroducedVersion": "1.0.0.0",
"IsActivity": false,
"IsActivityParty": false,
"IsAIRUpdated": false,
"IsAuditEnabled": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyauditsettings"
},
"IsAvailableOffline": false,
"IsBPFEentity": false,
"IsBusinessProcessEnabled": false,
"IsChildEntity": false,
"IsConnectionsEnabled": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyconnectionsettings"
},
"IsCustomEntity": true,
"IsCustomizable": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "iscustomizable"
},
"IsDocumentManagementEnabled": false,
"IsDocumentRecommendationsEnabled": false,
"IsDuplicateDetectionEnabled": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifieduplicatedetectionsettings"
},
"IsEnabledForCharts": true,
"IsEnabledForExternalChannels": false,
"IsEnabledForTrace": false,
"IsImportable": true,
"IsInteractionCentricEnabled": false,
"IsIntersect": false,
"IsKnowledgeManagementEnabled": false,
"IsLogicalEntity": false,
"IsMailMergeEnabled": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
```

```
        "Value": false,
        "CanBeChanged": true,
        "ManagedPropertyLogicalName": "canmodifymailmergesettings"
    },
    "IsManaged": false,
    "IsMappable": {
        "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
        "Value": true,
        "CanBeChanged": false,
        "ManagedPropertyLogicalName": "ismappable"
    },
    "IsMSTeamsIntegrationEnabled": false,
    "IsOfflineInMobileClient": {
        "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
        "Value": false,
        "CanBeChanged": true,
        "ManagedPropertyLogicalName": "canmodifymobileclientoffline"
    },
    "IsOneNoteIntegrationEnabled": false,
    "IsOptimisticConcurrencyEnabled": true,
    "IsPrivate": false,
    "IsQuickCreateEnabled": false,
    "IsReadingPaneEnabled": true,
    "IsReadOnlyInMobileClient": {
        "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
        "Value": false,
        "CanBeChanged": true,
        "ManagedPropertyLogicalName": "canmodifymobileclientreadonly"
    },
    "IsRenameable": {
        "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
        "Value": true,
        "CanBeChanged": true,
        "ManagedPropertyLogicalName": "isrenameable"
    },
    "IsSLAEnabled": false,
    "IsSolutionAware": false,
    "IsStateModelAware": true,
    "IsValidForAdvancedFind": true,
    "IsValidForQueue": {
        "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
        "Value": false,
        "CanBeChanged": true,
        "ManagedPropertyLogicalName": "canmodifyqueuesettings"
    },
    "IsVisibleInMobile": {
        "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
        "Value": false,
        "CanBeChanged": true,
        "ManagedPropertyLogicalName": "canmodifymobilevisibility"
    },
    "IsVisibleInMobileClient": {
        "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
        "Value": false,
        "CanBeChanged": true,
```

```
    "ManagedPropertyLogicalName": "canmodifymobileclientvisibility"
},
"LogicalCollectionName": "sample_bankaccounts",
"LogicalName": "sample_bankaccount",
"MobileOfflineFilters": "",
"ObjectTypeCode": 10393,
"OwnershipType": "UserOwned",
"PrimaryIdAttribute": "sample_bankaccountid",
"PrimaryNameAttribute": "sample_name",
"Privileges": [
{
    "@odata.type":
"Microsoft.Dynamics.CRM.SecurityPrivilegeMetadata",
    "CanBeBasic": true,
    "CanBeDeep": true,
    "CanBeGlobal": true,
    "CanBeLocal": true,
    "CanBeEntityReference": false,
    "CanBeParentEntityReference": false,
    "Name": "prvCreatesample_BankAccount",
    "PrivilegeId": "44f00701-716e-4584-8bab-cb0d263c070b",
    "PrivilegeType": "Create"
},
{
    "@odata.type":
"Microsoft.Dynamics.CRM.SecurityPrivilegeMetadata",
    "CanBeBasic": true,
    "CanBeDeep": true,
    "CanBeGlobal": true,
    "CanBeLocal": true,
    "CanBeEntityReference": false,
    "CanBeParentEntityReference": false,
    "Name": "prvReadsample_BankAccount",
    "PrivilegeId": "9cad3243-d0fe-467e-a731-c8b3416a6252",
    "PrivilegeType": "Read"
},
{
    "@odata.type":
"Microsoft.Dynamics.CRM.SecurityPrivilegeMetadata",
    "CanBeBasic": true,
    "CanBeDeep": true,
    "CanBeGlobal": true,
    "CanBeLocal": true,
    "CanBeEntityReference": false,
    "CanBeParentEntityReference": false,
    "Name": "prvWritessample_BankAccount",
    "PrivilegeId": "dc5465ed-223f-4b13-a272-fff25e5b5270",
    "PrivilegeType": "Write"
},
{
    "@odata.type":
"Microsoft.Dynamics.CRM.SecurityPrivilegeMetadata",
    "CanBeBasic": true,
    "CanBeDeep": true,
    "CanBeGlobal": true,
```

```
    "CanBeLocal": true,
    "CanBeEntityReference": false,
    "CanBeParentEntityReference": false,
    "Name": "prvDeletesample_BankAccount",
    "PrivilegeId": "9a409df2-ca4a-4ad9-8218-df88424dd7a0",
    "PrivilegeType": "Delete"
},
{
    "@odata.type":
"Microsoft.Dynamics.CRM.SecurityPrivilegeMetadata",
    "CanBeBasic": true,
    "CanBeDeep": true,
    "CanBeGlobal": true,
    "CanBeLocal": true,
    "CanBeEntityReference": false,
    "CanBeParentEntityReference": false,
    "Name": "prvAssignsample_BankAccount",
    "PrivilegeId": "73bf7dd3-f532-4468-abfe-84bbf0eae058",
    "PrivilegeType": "Assign"
},
{
    "@odata.type":
"Microsoft.Dynamics.CRM.SecurityPrivilegeMetadata",
    "CanBeBasic": true,
    "CanBeDeep": true,
    "CanBeGlobal": true,
    "CanBeLocal": true,
    "CanBeEntityReference": false,
    "CanBeParentEntityReference": false,
    "Name": "prvSharesample_BankAccount",
    "PrivilegeId": "292f6e27-9603-4835-882d-e28c175432ed",
    "PrivilegeType": "Share"
},
{
    "@odata.type":
"Microsoft.Dynamics.CRM.SecurityPrivilegeMetadata",
    "CanBeBasic": true,
    "CanBeDeep": true,
    "CanBeGlobal": true,
    "CanBeLocal": true,
    "CanBeEntityReference": false,
    "CanBeParentEntityReference": false,
    "Name": "prvAppendsample_BankAccount",
    "PrivilegeId": "42401aa6-6447-4fdc-9679-bcb89b62bd76",
    "PrivilegeType": "Append"
},
{
    "@odata.type":
"Microsoft.Dynamics.CRM.SecurityPrivilegeMetadata",
    "CanBeBasic": true,
    "CanBeDeep": true,
    "CanBeGlobal": true,
    "CanBeLocal": true,
    "CanBeEntityReference": false,
    "CanBeParentEntityReference": false,
```

```
        "Name": "prvAppendTosample_BankAccount",
        "PrivilegeId": "847ba62d-2f33-4208-87e6-52532b331f60",
        "PrivilegeType": "AppendTo"
    },
],
"ReportViewName": "Filteredsample_BankAccount",
"SchemaName": "sample_BankAccount",
"Settings": [],
"SyncToExternalSearchIndex": false,
"UsesBusinessDataLabelTable": false,
"MetadataId": "5872b902-112a-ed11-9db1-00224804f8e2"
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')
```

Salida de la consola:

```
Sending request to update the sample_BankAccount table...
Updated the Bank Account table
```

Sección 2: Crear, recuperar y actualizar columnas

Esta sección crea y recupera un grupo seleccionado de definiciones de columna. Cada uno de estos tipos se derivan de [AttributeMetadata EntityType](#) por lo que comparten la mayoría de las mismas propiedades comunes. Sin embargo, cada tipo derivado tiene algunas propiedades especiales.

Columna booleana

1. Cree una columna booleana usando [BooleanAttributeMetadata EntityType](#). A pesar del nombre, las columnas booleanas tienen una propiedad `OptionSet` al igual que las columnas de elección. Sin embargo, siempre tienen solo dos opciones: `TrueOption` con valor 1 y `FalseOption` con valor 0.

Solicitud:

HTTP

```
POST [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanAttributeMetadata",
    "AttributeType": "Boolean",
    "AttributeName": {
        "Value": "BooleanType"
    },
    "DefaultValue": false,
    "OptionSet": {
        "@odata.type": "Microsoft.Dynamics.CRM.BooleanOptionSetMetadata",
        "TrueOption": {
            "Value": 1,
            "Label": {
                "@odata.type": "Microsoft.Dynamics.CRM.Label",
                "LocalizedLabels": [
                    {
                        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                        "Label": "True",
                        "LanguageCode": 1033,
                        "IsManaged": false
                    }
                ],
                "UserLocalizedLabel": {
                    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                    "Label": "True",
                    "LanguageCode": 1033,
                    "IsManaged": false
                }
            }
        },
        "FalseOption": {
            "Value": 0,
            "Label": {
                "@odata.type": "Microsoft.Dynamics.CRM.Label",
                "LocalizedLabels": [
                    {
                        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                        "Label": "False",
                        "LanguageCode": 1033,
                        "IsManaged": false
                    }
                ],
            }
        }
    }
},
```

```

    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "False",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
{
},
},
"OptionsetType": "Boolean"
},
"Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Boolean Attribute",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Boolean Attribute",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Sample Boolean",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Sample Boolean",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"RequiredLevel": {
    "Value": "None",
    "CanBeChanged": false,
    "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
},
"SchemaName": "sample_Boolean"
}

```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(73f33b3d-112a-ed11-9db1-00224804f8e2)
```

2. Recupere la columna booleana que incluye `$expand=OptionSet` para que se puedan recuperar las opciones.

ⓘ Nota

La URL de esta solicitud incluye

`/Microsoft.Dynamics.CRM.BooleanAttributeMetadata`, que realiza una operación de conversión necesaria para devolver cualquier propiedad que no esté definida en **AttributeMetadata EntityType**. Sin esto, no es posible la expansión de `OptionSet`.

Solicitud:

HTTP

```
GET [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(LogicalName='sample_boolean')/Microsoft.Dynamics.CRM.Boolean
AttributeMetadata?$expand=OptionSet HTTP/1.1
Consistency: Strong
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
"@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#EntityDefinitions('sample_bankaccount')/At
tributes/Microsoft.Dynamics.CRM.BooleanAttributeMetadata(OptionSet())/$
entity",
"MetadataId": "73f33b3d-112a-ed11-9db1-00224804f8e2",
"HasChanged": null,
```

```
"AttributeOf": null,
"AttributeType": "Boolean",
"ColumnNumber": 35,
"DeprecatedVersion": null,
"IntroducedVersion": "1.0.0.0",
"EntityLogicalName": "sample_bankaccount",
"IsCustomAttribute": true,
"IsPrimaryId": false,
"IsValidODataAttribute": true,
"IsPrimaryName": false,
"IsValidForCreate": true,
"IsValidForRead": true,
"IsValidForUpdate": true,
"CanBeSecuredForRead": true,
"CanBeSecuredForCreate": true,
"CanBeSecuredForUpdate": true,
"IsSecured": false,
"IsRetrievable": false,
"IsFilterable": false,
"IsSearchable": false,
"IsManaged": false,
"LinkedAttributeId": null,
"LogicalName": "sample_boolean",
"IsValidForForm": true,
"IsRequiredForForm": false,
"IsValidForGrid": true,
"SchemaName": "sample_Boolean",
"ExternalName": null,
"IsLogical": false,
"IsDataSourceSecret": false,
"InheritsFrom": null,
"CreatedOn": "2022-09-01T16:15:08Z",
"ModifiedOn": "2022-09-01T16:15:08Z",
"SourceType": 0,
"AutoNumberFormat": null,
"DefaultValue": false,
"FormulaDefinition": "",
"SourceTypeMask": 0,
"AttributeTypeName": {
    "Value": "BooleanType"
},
"Description": {
    "LocalizedLabels": [
        {
            "Label": "Boolean Attribute",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "ea50b52d-53e4-4f8d-82ce-8f74a7554800",
            "HasChanged": null
        }
    ],
    "UserLocalizedLabel": {
        "Label": "Boolean Attribute",
        "LanguageCode": 1033,
        "IsManaged": false,
```

```
        "MetadataId": "ea50b52d-53e4-4f8d-82ce-8f74a7554800",
        "HasChanged": null
    },
},
"DisplayName": {
    "LocalizedLabels": [
        {
            "Label": "Sample Boolean",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "9e4daa21-8774-4de9-b467-d046389459dc",
            "HasChanged": null
        }
    ],
    "UserLocalizedLabel": {
        "Label": "Sample Boolean",
        "LanguageCode": 1033,
        "IsManaged": false,
        "MetadataId": "9e4daa21-8774-4de9-b467-d046389459dc",
        "HasChanged": null
    }
},
"IsAuditEnabled": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyauditsettings"
},
"IsGlobalFilterEnabled": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyglobalfiltersettings"
},
"IsSortableEnabled": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyissortablesettings"
},
"IsCustomizable": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "iscustomizable"
},
"IsRenameable": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "isrenameable"
},
"IsValidForAdvancedFind": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifysearchsettings"
},
"RequiredLevel": {
    "Value": "None",
    "CanBeChanged": false,
```

```
        "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
    },
    "CanModifyAdditionalSettings": {
        "Value": true,
        "CanBeChanged": true,
        "ManagedPropertyLogicalName": "canmodifyadditionalsettings"
    },
    "Settings": [],
    "OptionSet": {
        "MetadataId": "74f33b3d-112a-ed11-9db1-00224804f8e2",
        "HasChanged": null,
        "IsCustomOptionSet": true,
        "IsGlobal": false,
        "IsManaged": false,
        "Name": "sample_bankaccount_sample_boolean",
        "ExternalTypeName": null,
        "OptionsetType": "Boolean",
        "IntroducedVersion": "1.0.0.0",
        "Description": {
            "LocalizedLabels": [
                {
                    "Label": "Boolean Attribute",
                    "LanguageCode": 1033,
                    "IsManaged": false,
                    "MetadataId": "76f33b3d-112a-ed11-9db1-00224804f8e2",
                    "HasChanged": null
                }
            ],
            "UserLocalizedLabel": {
                "Label": "Boolean Attribute",
                "LanguageCode": 1033,
                "IsManaged": false,
                "MetadataId": "76f33b3d-112a-ed11-9db1-00224804f8e2",
                "HasChanged": null
            }
        },
        "DisplayName": {
            "LocalizedLabels": [
                {
                    "Label": "Sample Boolean",
                    "LanguageCode": 1033,
                    "IsManaged": false,
                    "MetadataId": "75f33b3d-112a-ed11-9db1-00224804f8e2",
                    "HasChanged": null
                }
            ],
            "UserLocalizedLabel": {
                "Label": "Sample Boolean",
                "LanguageCode": 1033,
                "IsManaged": false,
                "MetadataId": "75f33b3d-112a-ed11-9db1-00224804f8e2",
                "HasChanged": null
            }
        },
        "IsCustomizable": {
```

```
        "Value": true,
        "CanBeChanged": true,
        "ManagedPropertyLogicalName": "iscustomizable"
    },
    "TrueOption": {
        "Value": 1,
        "Color": null,
        "IsManaged": false,
        "ExternalValue": "",
        "ParentValues": [],
        "MetadataId": null,
        "HasChanged": null,
        "Label": {
            "LocalizedLabels": [
                {
                    "Label": "True",
                    "LanguageCode": 1033,
                    "IsManaged": false,
                    "MetadataId": "12049c5f-e99d-453f-8315-3933512539a1",
                    "HasChanged": null
                }
            ],
            "UserLocalizedLabel": {
                "Label": "True",
                "LanguageCode": 1033,
                "IsManaged": false,
                "MetadataId": "12049c5f-e99d-453f-8315-3933512539a1",
                "HasChanged": null
            }
        },
        "Description": {
            "LocalizedLabels": [],
            "UserLocalizedLabel": null
        }
    },
    "FalseOption": {
        "Value": 0,
        "Color": null,
        "IsManaged": false,
        "ExternalValue": "",
        "ParentValues": [],
        "MetadataId": null,
        "HasChanged": null,
        "Label": {
            "LocalizedLabels": [
                {
                    "Label": "False",
                    "LanguageCode": 1033,
                    "IsManaged": false,
                    "MetadataId": "e3d4c2b1-ad54-4d3a-8e01-f759da0e476f",
                    "HasChanged": null
                }
            ],
            "UserLocalizedLabel": {
                "Label": "False",

```

```
        "LanguageCode": 1033,
        "IsManaged": false,
        "MetadataId": "e3d4c2b1-ad54-4d3a-8e01-f759da0e476f",
        "HasChanged": null
    }
},
"Description": {
    "LocalizedLabels": [],
    "UserLocalizedLabel": null
}
}
}
```

Salida de la consola:

```
Original Option Labels:
True Option Label:'True' Value: 1
False Option Label:'False' Value: 0
```

3. Actualice la columna booleana. Los únicos cambios son en las propiedades

`DisplayName`, `Description` y `RequiredLevel`, pero se incluye toda la definición porque se usa `PUT`.

ⓘ Nota

A pesar de que la propiedad `OptionSet` está incluida en esta carga útil, los cambios en las opciones no se aplicarán porque no se consideran parte de la definición de la columna. Deben actualizarse por separado y este ejemplo le mostrará cómo hacerlo en los siguientes pasos.

Solicitud:

HTTP

```
PUT [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(LogicalName='sample_boolean') HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
MSCRM.MergeLabels: true
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

```
{
  "@odata.type": "Microsoft.Dynamics.CRM.BooleanAttributeMetadata",
  "AttributeType": "Boolean",
  "AttributeName": {
    "Value": "BooleanType"
  },
  "DefaultValue": false,
  "FormulaDefinition": "",
  "SourceTypeMask": 0,
  "OptionSet": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanOptionSetMetadata",
    "TrueOption": {
      "Value": 1,
      "Label": {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [
          {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "True",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "12049c5f-e99d-453f-8315-3933512539a1"
          }
        ],
        "UserLocalizedLabel": {
          "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
          "Label": "True",
          "LanguageCode": 1033,
          "IsManaged": false,
          "MetadataId": "12049c5f-e99d-453f-8315-3933512539a1"
        }
      }
    },
    "UserLocalizedLabel": {
      "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
      "Label": "True",
      "LanguageCode": 1033,
      "IsManaged": false,
      "MetadataId": "12049c5f-e99d-453f-8315-3933512539a1"
    }
  },
  "Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [],
    "UserLocalizedLabel": {
      "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
      "LanguageCode": 0,
      "IsManaged": false
    }
  },
  "IsManaged": false,
  "ExternalValue": ""
},
  "FalseOption": {
    "Value": 0,
    "Label": {
      "@odata.type": "Microsoft.Dynamics.CRM.Label",
      "LocalizedLabels": [
        {
          "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
          "Label": "False",
          "LanguageCode": 1033,
          "IsManaged": false,
          "MetadataId": "e3d4c2b1-ad54-4d3a-8e01-f759da0e476f"
        }
      ]
    }
  }
}
```

```
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "False",
        "LanguageCode": 1033,
        "IsManaged": false,
        "MetadataId": "e3d4c2b1-ad54-4d3a-8e01-f759da0e476f"
    }
},
{
    "Description": {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [],
        "UserLocalizedLabel": {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "LanguageCode": 0,
            "IsManaged": false
        }
    },
    "IsManaged": false,
    "ExternalValue": ""
},
{
    "OptionsetType": "Boolean",
    "Description": {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [
            {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Boolean Attribute",
                "LanguageCode": 1033,
                "IsManaged": false,
                "MetadataId": "76f33b3d-112a-ed11-9db1-00224804f8e2"
            }
        ],
        "UserLocalizedLabel": {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Boolean Attribute",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "76f33b3d-112a-ed11-9db1-00224804f8e2"
        }
    }
},
{
    "DisplayName": {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [
            {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Sample Boolean",
                "LanguageCode": 1033,
                "IsManaged": false,
                "MetadataId": "75f33b3d-112a-ed11-9db1-00224804f8e2"
            }
        ],
        "UserLocalizedLabel": {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Sample Boolean",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "75f33b3d-112a-ed11-9db1-00224804f8e2"
        }
    }
}
```

```
        "Label": "Sample Boolean",
        "LanguageCode": 1033,
        "IsManaged": false,
        "MetadataId": "75f33b3d-112a-ed11-9db1-00224804f8e2"
    }
},
"IntroducedVersion": "1.0.0.0",
"IsCustomizable": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "iscustomizable"
},
"IsCustomOptionSet": true,
"IsGlobal": false,
"IsManaged": false,
"Name": "sample_bankaccount_sample_boolean",
"MetadataId": "74f33b3d-112a-ed11-9db1-00224804f8e2"
},
"CanBeSecuredForCreate": true,
"CanBeSecuredForRead": true,
"CanBeSecuredForUpdate": true,
"CanModifyAdditionalSettings": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyadditionalsettings"
},
"ColumnNumber": 35,
"Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Boolean Attribute Updated",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Boolean Attribute Updated",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Sample Boolean Updated",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ]
}
```

```
],
  "UserLocalizedLabel": {
    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
    "Label": "Sample Boolean Updated",
    "LanguageCode": 1033,
    "IsManaged": false
  }
},
"EntityLogicalName": "sample_bankaccount",
"IntroducedVersion": "1.0.0.0",
"IsAuditEnabled": {
  "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
  "Value": true,
  "CanBeChanged": true,
  "ManagedPropertyLogicalName": "canmodifyauditsettings"
},
"IsCustomAttribute": true,
"IsCustomizable": {
  "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
  "Value": true,
  "CanBeChanged": true,
  "ManagedPropertyLogicalName": "iscustomizable"
},
"IsDataSourceSecret": false,
"IsFilterable": false,
"IsGlobalFilterEnabled": {
  "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
  "Value": false,
  "CanBeChanged": true,
  "ManagedPropertyLogicalName": "canmodifyglobalfiltersettings"
},
"IsLogical": false,
"IsManaged": false,
"IsPrimaryId": false,
"IsPrimaryName": false,
"IsRenameable": {
  "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
  "Value": true,
  "CanBeChanged": true,
  "ManagedPropertyLogicalName": "isrenameable"
},
"IsRequiredForForm": false,
"IsRetrievable": false,
"IsSearchable": false,
"IsSecured": false,
"IsSortableEnabled": {
  "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
  "Value": false,
  "CanBeChanged": true,
  "ManagedPropertyLogicalName": "canmodifyissortablesettings"
},
"IsValidForAdvancedFind": {
  "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
  "Value": true,
  "CanBeChanged": true,
```

```
        "ManagedPropertyLogicalName": "canmodifysearchsettings"
    },
    "IsValidForCreate": true,
    "IsValidForForm": true,
    "IsValidForGrid": true,
    "IsValidForRead": true,
    "IsValidForUpdate": true,
    "LogicalName": "sample_boolean",
    "RequiredLevel": {
        "Value": "ApplicationRequired",
        "CanBeChanged": false,
        "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
    },
    "SchemaName": "sample_Boolean",
    "SourceType": 0,
    "MetadataId": "73f33b3d-112a-ed11-9db1-00224804f8e2"
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(LogicalName='sample_boolean')
```

Salida de la consola:

```
Updated Boolean Column properties
```

Valores de opción de actualización

Actualice cada una de las opciones booleanas usando [UpdateOptionValue Action](#).

ⓘ Nota

Aquí estamos aplicando cambios a las opciones en un atributo booleano, pero usará [UpdateOptionValue](#) para opciones en cualquier tipo de columna que las use excepto columnas `status`, donde debe utilizar [UpdateStateValue Action](#).

1. Cambie el valor de la etiqueta `TrueOption` a 'Arriba'.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/UpdateOptionValue HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
  "AttributeLogicalName": "sample_boolean",
  "EntityLogicalName": "sample_bankaccount",
  "Value": 1,
  "Label": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
      {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Up",
        "LanguageCode": 1033,
        "IsManaged": false
      }
    ],
    "UserLocalizedLabel": {
      "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
      "Label": "Up",
      "LanguageCode": 1033,
      "IsManaged": false
    }
  },
  "MergeLabels": true
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
```

2. Cambie el valor de la etiqueta `FalseOption` a 'Abajo'.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/UpdateOptionValue HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
```

```
Accept: application/json
```

```
{
  "AttributeLogicalName": "sample_boolean",
  "EntityLogicalName": "sample_bankaccount",
  "Value": 0,
  "Label": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
      {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Down",
        "LanguageCode": 1033,
        "IsManaged": false
      }
    ],
    "UserLocalizedLabel": {
      "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
      "Label": "Down",
      "LanguageCode": 1033,
      "IsManaged": false
    }
  },
  "MergeLabels": true
}
```

Respuesta:

```
HTTP
```

```
HTTP/1.1 204 NoContent
```

```
OData-Version: 4.0
```

Salida de la consola:

```
Updated option labels
```

3. Recupere los valores de opción modificados para la columna booleana usando la misma consulta que antes:

Salida de la consola:

```
Updated Option Labels:
Updated True Option Label:'Up' Value: 1
Updated False Option Label:'Down' Value: 0
```

Columna DateTime

1. Cree una columna DateTime mediante [DateTimeAttributeMetadata EntityType](#).

Solicitud:

HTTP

```
POST [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
  "@odata.type": "Microsoft.Dynamics.CRM.DateTimeAttributeMetadata",
  "AttributeType": "DateTime",
  "AttributeName": {
    "Value": "DateTimeType"
  },
  "Format": "DateOnly",
  "ImeMode": "Disabled",
  "DateTimeBehavior": {
    "Value": "DateOnly"
  },
  "Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
      {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "DateTime Attribute",
        "LanguageCode": 1033,
        "IsManaged": false
      }
    ],
    "UserLocalizedLabel": {
      "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
      "Label": "DateTime Attribute",
      "LanguageCode": 1033,
      "IsManaged": false
    }
  },
  "DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
      {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Sample DateTime",
        "LanguageCode": 1033,
        "IsManaged": false
      }
    ]
  }
}
```

```
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Sample DateTime",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"RequiredLevel": {
    "Value": "None",
    "CanBeChanged": false,
    "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
},
"SchemaName": "sample_DateTime"
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(f1db3d43-112a-ed11-9db1-00224804f8e2)
```

Salida de la consola:

```
Created DateTime column with id:f1db3d43-112a-ed11-9db1-00224804f8e2
```

2. Recupere los valores seleccionados de la columna DateTime.

Solicitud:

HTTP

```
GET [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(LogicalName='sample_datetime')/Microsoft.Dynamics.CRM.DateTime
meAttributeMetadata?$select=SchemaName,Format,DateTimeBehavior HTTP/1.1
Consistency: Strong
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#EntityDefinitions('sample_bankaccount')/At
tributes/Microsoft.Dynamics.CRM.DateTimeAttributeMetadata(SchemaName,Fo
rmat,DateTimeBehavior)/$entity",
  "SchemaName": "sample_DateTime",
  "Format": "DateOnly",
  "MetadataId": "f1db3d43-112a-ed11-9db1-00224804f8e2",
  "DateTimeBehavior": {
    "Value": "DateOnly"
  }
}
```

Salida de la consola:

```
Retrieved Datetime column properties:
  DateTime Format:'DateOnly'
  DateTime DateTimeBehavior:'DateOnly'
```

Columna decimal

1. Cree una columna Decimal usando [DecimalAttributeMetadata EntityType](#).

Solicitud:

HTTP

```
POST [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
  "@odata.type": "Microsoft.Dynamics.CRM.DecimalAttributeMetadata",
  "AttributeType": "Decimal",
  "AttributeTypeName": {
```

```

    "Value": "DecimalType"
},
".MaxValue": 100.0,
".MinValue": 0.0,
"Precision": 1,
"Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Decimal Attribute",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Decimal Attribute",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Sample Decimal",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Sample Decimal",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"RequiredLevel": {
    "Value": "None",
    "CanBeChanged": false,
    "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
},
"SchemaName": "sample_Decimal"
}

```

Respuesta:

HTTP

HTTP/1.1 204 NoContent
OData-Version: 4.0

```
OData-EntityId: [Organization  
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/  
Attributes(f2db3d43-112a-ed11-9db1-00224804f8e2)
```

Salida de la consola:

```
Created Decimal column with id:f2db3d43-112a-ed11-9db1-00224804f8e2
```

2. Recupere los valores seleccionados de la columna Decimal.

Solicitud:

HTTP

```
GET [Organization  
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/  
Attributes(LogicalName='sample_decimal')/Microsoft.Dynamics.CRM.Decimal  
AttributeMetadata?$select=SchemaName,.MaxValue,.MinValue,Precision  
HTTP/1.1  
Consistency: Strong  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK  
OData-Version: 4.0  
  
{  
"@odata.context": "[Organization  
Uri]/api/data/v9.2/$metadata#EntityDefinitions('sample_bankaccount')/At  
tributes/Microsoft.Dynamics.CRM.DecimalAttributeMetadata(SchemaName,Max  
Value,MinValue,Precision)/$entity",  
"SchemaName": "sample_Decimal",  
".MaxValue": 100,  
".MinValue": 0,  
"Precision": 1,  
"MetadataId": "f2db3d43-112a-ed11-9db1-00224804f8e2"  
}
```

Salida de la consola:

```
Retrieved Decimal column properties:  
Decimal MaxValue:100  
Decimal MinValue:0  
Decimal Precision:1
```

Columnas de enteros

1. Cree una columna de enteros usando [IntegerAttributeMetadata EntityType](#).

Solicitud:

HTTP

```
POST [Organization  
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/  
Attributes HTTP/1.1  
MSCRM.SolutionUniqueName: examplesolution  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json  
  
{  
    "@odata.type": "Microsoft.Dynamics.CRM.IntegerAttributeMetadata",  
    "AttributeType": "Integer",  
    "AttributeTypeName": {  
        "Value": "IntegerType"  
    },  
    ".MaxValue": 100,  
    ".MinValue": 0,  
    "Format": "None",  
    "SourceTypeMask": 0,  
    "Description": {  
        "@odata.type": "Microsoft.Dynamics.CRM.Label",  
        "LocalizedLabels": [  
            {  
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",  
                "Label": "Integer Attribute",  
                "LanguageCode": 1033,  
                "IsManaged": false  
            }  
        ],  
        "UserLocalizedLabel": {  
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",  
            "Label": "Integer Attribute",  
            "LanguageCode": 1033,  
            "IsManaged": false  
        }  
    },  
    "DisplayName": {  
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
```

```
"LocalizedLabels": [
  {
    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
    "Label": "Sample Integer",
    "LanguageCode": 1033,
    "IsManaged": false
  }
],
"UserLocalizedLabel": {
  "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
  "Label": "Sample Integer",
  "LanguageCode": 1033,
  "IsManaged": false
}
},
"RequiredLevel": {
  "Value": "None",
  "CanBeChanged": false,
  "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
},
"SchemaName": "sample_Integer"
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(f5db3d43-112a-ed11-9db1-00224804f8e2)
```

Salida de la consola:

```
Created Integer column with id:f5db3d43-112a-ed11-9db1-00224804f8e2
```

2. Recupere los valores seleccionados de la columna de enteros.

Solicitud:

HTTP

```
GET [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(LogicalName='sample_integer')/Microsoft.Dynamics.CRM.Integer
AttributeMetadata?$select=SchemaName,.MaxValue,.MinValue,Format HTTP/1.1
Consistency: Strong
```

```
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#EntityDefinitions('sample_bankaccount')/At
tributes/Microsoft.Dynamics.CRM.IntegerAttributeMetadata(SchemaName,Max
Value,MinValue,Format)/$entity",
  "SchemaName": "sample_Integer",
  ".MaxValue": 100,
  ".MinValue": 0,
  "Format": "None",
  "MetadataId": "f5db3d43-112a-ed11-9db1-00224804f8e2"
}
```

Salida de la consola:

```
Retrieved Integer column properties:
Integer MaxValue:100
Integer MinValue:0
Integer Format:None
```

Columna de notas

1. Cree una columna de notas usando [MemoAttributeMetadata EntityType](#).

Solicitud:

HTTP

```
POST [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

```
{
  "@odata.type": "Microsoft.Dynamics.CRM.MemoAttributeMetadata",
  "AttributeType": "Memo",
  "AttributeName": {
    "Value": "MemoType"
  },
  "Format": "TextArea",
  "ImeMode": "Disabled",
  "MaxLength": 500,
  "IsLocalizable": false,
  "Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
      {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Memo Attribute",
        "LanguageCode": 1033,
        "IsManaged": false
      }
    ],
    "UserLocalizedLabel": {
      "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
      "Label": "Memo Attribute",
      "LanguageCode": 1033,
      "IsManaged": false
    }
  },
  "DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
      {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Sample Memo",
        "LanguageCode": 1033,
        "IsManaged": false
      }
    ],
    "UserLocalizedLabel": {
      "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
      "Label": "Sample Memo",
      "LanguageCode": 1033,
      "IsManaged": false
    }
  },
  "RequiredLevel": {
    "Value": "None",
    "CanBeChanged": false,
    "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
  },
  "SchemaName": "sample_Memo"
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(f6db3d43-112a-ed11-9db1-00224804f8e2)
```

Salida de la consola:

```
Created Memo column with id:f6db3d43-112a-ed11-9db1-00224804f8e2
```

2. Recupere los valores seleccionados de la columna de notas.

Solicitud:

HTTP

```
GET [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(LogicalName='sample_memo')/Microsoft.Dynamics.CRM.MemoAttributeMetadata?$select=SchemaName,Format,ImeMode,MaxLength HTTP/1.1
Consistency: Strong
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
"@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#EntityDefinitions('sample_bankaccount')/Attributes/Microsoft.Dynamics.CRM.MemoAttributeMetadata(SchemaName,Format
,ImeMode,MaxLength)/$entity",
"SchemaName": "sample_Memo",
"Format": "TextArea",
"ImeMode": "Disabled",
"MaxLength": 500,
"MetadataId": "f6db3d43-112a-ed11-9db1-00224804f8e2"
}
```

Salida de la consola:

```
Retrieved Memo column properties:  
Memo Format:TextArea  
Memo ImeMode:Disabled  
Memo MaxLength:500
```

Columna de moneda

1. Crear una columna de moneda mediante [MoneyAttributeMetadata EntityType](#).

Solicitud:

HTTP

```
POST [Organization  
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/  
Attributes HTTP/1.1  
MSCRM.SolutionUniqueName: examplesolution  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json  
  
{  
    "@odata.type": "Microsoft.Dynamics.CRM.MoneyAttributeMetadata",  
    "AttributeType": "Money",  
    "AttributeTypeName": {  
        "Value": "MoneyType"  
    },  
    "ImeMode": "Disabled",  
    ".MaxValue": 1000.0,  
    ".MinValue": 0.0,  
    "Precision": 1,  
    "PrecisionSource": 1,  
    "SourceTypeMask": 0,  
    "IsBaseCurrency": false,  
    "Description": {  
        "@odata.type": "Microsoft.Dynamics.CRM.Label",  
        "LocalizedLabels": [  
            {  
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",  
                "Label": "Money Attribute",  
                "LanguageCode": 1033,  
                "IsManaged": false  
            }  
        ],  
        "UserLocalizedLabel": {  
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",  
            "Label": "Moneda",  
            "LanguageCode": 1033,  
            "IsManaged": false  
        }  
    }  
}
```

```

        "Label": "Money Attribute",
        "LanguageCode": 1033,
        "IsManaged": false
    },
},
"DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Sample Money",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Sample Money",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"RequiredLevel": {
    "Value": "None",
    "CanBeChanged": false,
    "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
},
"SchemaName": "sample_Money"
}

```

Respuesta:

HTTP

```

HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(fddb3d43-112a-ed11-9db1-00224804f8e2)

```

Salida de la consola:

```
Created Money column with id:fddb3d43-112a-ed11-9db1-00224804f8e2
```

2. Recupere los valores seleccionados de la columna de moneda.

Solicitud:

HTTP

```
GET [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(LogicalName='sample_money')/Microsoft.Dynamics.CRM.MoneyAttr
ibuteMetadata?
$select=SchemaName,.MaxValue,.MinValue,Precision,PrecisionSource,ImeMode
HTTP/1.1
Consistency: Strong
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#EntityDefinitions('sample_bankaccount')/At
tributes/Microsoft.Dynamics.CRM.MoneyAttributeMetadata(SchemaName,MaxVa
lue,MinValue,Precision,PrecisionSource,ImeMode)/$entity",
  "SchemaName": "sample_Money",
  ".MaxValue": 1000.0,
  ".MinValue": 0.0,
  "Precision": 1,
  "PrecisionSource": 1,
  "ImeMode": "Disabled",
  "MetadataId": "fdedb3d43-112a-ed11-9db1-00224804f8e2"
}
```

Salida de la consola:

```
Retrieved Money column properties:
Money MaxValue:1000
Money MinValue:0
Money Precision:1
Money PrecisionSource:1
Money ImeMode:Disabled
```

Columna de lista de selección

1. Cree una columna de opciones (lista de selección) usando [PicklistAttributeMetadata EntityType](#) con un conjunto de opciones local.

Solicitud:

HTTP

```
POST [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
"@odata.type": "Microsoft.Dynamics.CRM.PicklistAttributeMetadata",
"AttributeType": "Picklist",
"AttributeName": {
    "Value": "PicklistType"
},
"SourceTypeMask": 0,
"OptionSet": {
    "@odata.type": "Microsoft.Dynamics.CRM.OptionSetMetadata",
    "Options": [
        {
            "Value": 727000000,
            "Label": {
                "@odata.type": "Microsoft.Dynamics.CRM.Label",
                "LocalizedLabels": [
                    {
                        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                        "Label": "Bravo",
                        "LanguageCode": 1033,
                        "IsManaged": false
                    }
                ],
                "UserLocalizedLabel": {
                    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                    "Label": "Bravo",
                    "LanguageCode": 1033,
                    "IsManaged": false
                }
            }
        },
        {
            "Value": 727000001,
            "Label": {
                "@odata.type": "Microsoft.Dynamics.CRM.Label",
                "LocalizedLabels": [
                    {
                        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                        "Label": "Bravo"
                    }
                ]
            }
        }
    ]
}
```

```
        "Label": "Delta",
        "LanguageCode": 1033,
        "IsManaged": false
    }
],
"UserLocalizedLabel": {
    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
    "Label": "Delta",
    "LanguageCode": 1033,
    "IsManaged": false
}
},
{
"Value": 727000002,
"Label": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Alpha",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Alpha",
        "LanguageCode": 1033,
        "IsManaged": false
    }
}
},
{
"Value": 727000003,
"Label": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Charlie",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Charlie",
        "LanguageCode": 1033,
        "IsManaged": false
    }
}
},
{
"Value": 727000004,
```

```
"Label": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Foxtrot",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Foxtrot",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"IsGlobal": false,
"OptionsetType": "Picklist"
},
"Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Choice Attribute",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Choice Attribute",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Sample Choice",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Sample Choice",
        "LanguageCode": 1033,
        "IsManaged": false
    }
}
```

```
},
"RequiredLevel": {
    "Value": "None",
    "CanBeChanged": false,
    "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
},
"SchemaName": "sample_Choice"
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(4a154e49-112a-ed11-9db1-00224804f8e2)
```

Salida de la consola:

```
Created Choice column with id:4a154e49-112a-ed11-9db1-00224804f8e2
```

2. Recuperar opciones de la columna de elección usando

```
$select=SchemaName&$expand=OptionSet.
```

Solicitud:

HTTP

```
GET [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(LogicalName='sample_choice')/Microsoft.Dynamics.CRM.Picklist
AttributeMetadata?$select=SchemaName&$expand=OptionSet HTTP/1.1
Consistency: Strong
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0
```

```
{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#EntityDefinitions('sample_bankaccount')/At
tributes/Microsoft.Dynamics.CRM.PicklistAttributeMetadata(SchemaName,Op
tionSet())/$entity",
  "SchemaName": "sample_Choice",
  "MetadataId": "4a154e49-112a-ed11-9db1-00224804f8e2",
  "OptionSet": {
    "MetadataId": "4b154e49-112a-ed11-9db1-00224804f8e2",
    "HasChanged": null,
    "IsCustomOptionSet": true,
    "IsGlobal": false,
    "IsManaged": false,
    "Name": "sample_bankaccount_sample_choice",
    "ExternalTypeName": null,
    "OptionsetType": "Picklist",
    "IntroducedVersion": "1.0.0.0",
    "ParentOptionSetName": null,
    "Description": {
      "LocalizedLabels": [
        {
          "Label": "Choice Attribute",
          "LanguageCode": 1033,
          "IsManaged": false,
          "MetadataId": "4d154e49-112a-ed11-9db1-00224804f8e2",
          "HasChanged": null
        }
      ],
      "UserLocalizedLabel": {
        "Label": "Choice Attribute",
        "LanguageCode": 1033,
        "IsManaged": false,
        "MetadataId": "4d154e49-112a-ed11-9db1-00224804f8e2",
        "HasChanged": null
      }
    },
    "DisplayName": {
      "LocalizedLabels": [
        {
          "Label": "Sample Choice",
          "LanguageCode": 1033,
          "IsManaged": false,
          "MetadataId": "4c154e49-112a-ed11-9db1-00224804f8e2",
          "HasChanged": null
        }
      ],
      "UserLocalizedLabel": {
        "Label": "Sample Choice",
        "LanguageCode": 1033,
        "IsManaged": false,
        "MetadataId": "4c154e49-112a-ed11-9db1-00224804f8e2",
        "HasChanged": null
      }
    }
  }
},
```

```
"IsCustomizable": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "iscustomizable"
},
"Options": [
    {
        "Value": 727000000,
        "Color": null,
        "IsManaged": false,
        "ExternalValue": "",
        "ParentValues": [],
        "MetadataId": null,
        "HasChanged": null,
        "Label": {
            "LocalizedLabels": [
                {
                    "Label": "Bravo",
                    "LanguageCode": 1033,
                    "IsManaged": false,
                    "MetadataId": "bc8d1815-75b7-4c13-b618-7959aaf4abb6",
                    "HasChanged": null
                }
            ],
            "UserLocalizedLabel": {
                "Label": "Bravo",
                "LanguageCode": 1033,
                "IsManaged": false,
                "MetadataId": "bc8d1815-75b7-4c13-b618-7959aaf4abb6",
                "HasChanged": null
            }
        },
        "Description": {
            "LocalizedLabels": [],
            "UserLocalizedLabel": null
        }
    },
    {
        "Value": 727000001,
        "Color": null,
        "IsManaged": false,
        "ExternalValue": "",
        "ParentValues": [],
        "MetadataId": null,
        "HasChanged": null,
        "Label": {
            "LocalizedLabels": [
                {
                    "Label": "Delta",
                    "LanguageCode": 1033,
                    "IsManaged": false,
                    "MetadataId": "c3613791-85a0-41ac-8575-91aca4bb91e8",
                    "HasChanged": null
                }
            ],
            "UserLocalizedLabel": {
                "Label": "Delta",
                "LanguageCode": 1033,
                "IsManaged": false,
                "MetadataId": "c3613791-85a0-41ac-8575-91aca4bb91e8",
                "HasChanged": null
            }
        }
    }
],
```

```
        "UserLocalizedLabel": {
            "Label": "Delta",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "c3613791-85a0-41ac-8575-91aca4bb91e8",
            "HasChanged": null
        }
    },
    "Description": {
        "LocalizedLabels": [],
        "UserLocalizedLabel": null
    }
},
{
    "Value": 727000002,
    "Color": null,
    "IsManaged": false,
    "ExternalValue": "",
    "ParentValues": [],
    "MetadataId": null,
    "HasChanged": null,
    "Label": {
        "LocalizedLabels": [
            {
                "Label": "Alpha",
                "LanguageCode": 1033,
                "IsManaged": false,
                "MetadataId": "8db04562-9ec3-4014-a170-0482bbb94e44",
                "HasChanged": null
            }
        ],
        "UserLocalizedLabel": {
            "Label": "Alpha",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "8db04562-9ec3-4014-a170-0482bbb94e44",
            "HasChanged": null
        }
    },
    "Description": {
        "LocalizedLabels": [],
        "UserLocalizedLabel": null
    }
},
{
    "Value": 727000003,
    "Color": null,
    "IsManaged": false,
    "ExternalValue": "",
    "ParentValues": [],
    "MetadataId": null,
    "HasChanged": null,
    "Label": {
        "LocalizedLabels": [
            {

```

```
        "Label": "Charlie",
        "LanguageCode": 1033,
        "IsManaged": false,
        "MetadataId": "d00dc11e-ed91-478b-ac78-86b6784326ad",
        "HasChanged": null
    }
],
"UserLocalizedLabel": {
    "Label": "Charlie",
    "LanguageCode": 1033,
    "IsManaged": false,
    "MetadataId": "d00dc11e-ed91-478b-ac78-86b6784326ad",
    "HasChanged": null
}
},
"Description": {
    "LocalizedLabels": [],
    "UserLocalizedLabel": null
}
},
{
    "Value": 727000004,
    "Color": null,
    "IsManaged": false,
    "ExternalValue": "",
    "ParentValues": [],
    "MetadataId": null,
    "HasChanged": null,
    "Label": {
        "LocalizedLabels": [
            {
                "Label": "Foxtrot",
                "LanguageCode": 1033,
                "IsManaged": false,
                "MetadataId": "36a565b7-cd21-4505-812b-5567c28eec23",
                "HasChanged": null
            }
        ],
        "UserLocalizedLabel": {
            "Label": "Foxtrot",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "36a565b7-cd21-4505-812b-5567c28eec23",
            "HasChanged": null
        }
    },
    "Description": {
        "LocalizedLabels": [],
        "UserLocalizedLabel": null
    }
}
]
}
```

Salida de la consola:

```
Retrieved Choice column options:  
    Value:727000000 Label:Bravo  
    Value:727000001 Label:Delta  
    Value:727000002 Label:Alpha  
    Value:727000003 Label:Charlie  
    Value:727000004 Label:Foxtrot
```

Agregar una opción al conjunto de opciones local

1. Agregue una opción a la columna de elección usando [InsertOptionValue Action](#).

⚠️ Nota

`InsertOptionValue` y las siguientes acciones para trabajar con opciones tiene un parámetro `SolutionUniqueName` para que establezca el nombre único de la solución en lugar de usar el encabezado de solicitud

`MSCRM.SolutionUniqueName`.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/InsertOptionValue HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json  
  
{  
    "AttributeLogicalName": "sample_choice",  
    "EntityLogicalName": "sample_bankaccount",  
    "Value": 727000005,  
    "Label": {  
        "@odata.type": "Microsoft.Dynamics.CRM.Label",  
        "LocalizedLabels": [  
            {  
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",  
                "Label": "Echo",  
                "LanguageCode": 1033,  
                "IsManaged": false  
            }  
        ],  
        "UserLocalizedLabel": {
```

```
    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
    "Label": "Echo",
    "LanguageCode": 1033,
    "IsManaged": false
  },
},
"SolutionUniqueName": "examplesolution"
}
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.InsertOptionValueRe
sponse",
  "NewOptionValue": 727000005
}
```

Salida de la consola:

```
Added new option with label 'Echo'
```

2. Recupere las opciones de la columna de elección nuevamente usando la misma consulta que antes:

Salida de la consola:

```
The option values for the picklist:
  Value: 727000000, Label:Bravo
  Value: 727000001, Label:Delta
  Value: 727000002, Label:Alpha
  Value: 727000003, Label:Charlie
  Value: 727000004, Label:Foxtrot
  Value: 727000005, Label:Echo
```

Opciones para reordenar la columna de opciones

1. Reordene las opciones de la columna de opciones mediante [OrderOption Action](#).

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/OrderOption HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
  "EntityLogicalName": "sample_bankaccount",
  "AttributeLogicalName": "sample_choice",
  "Values": [
    727000002,
    727000000,
    727000003,
    727000001,
    727000005,
    727000004
  ],
  "SolutionUniqueName": "examplesolution"
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
```

Salida de la consola:

```
Options re-ordered.
```

2. Recupere las opciones de la columna Elección nuevamente usando la misma consulta que antes para ver las opciones en el nuevo orden.

Salida de la consola:

```
The option values for the picklist in the new order:
Value: 727000002, Label:Alpha
Value: 727000000, Label:Bravo
Value: 727000003, Label:Charlie
Value: 727000001, Label:Delta
```

```
Value: 727000005, Label:Echo  
Value: 727000004, Label:Foxtrot
```

Eliminar valor de opción local

1. Elimine una opción mediante [DeleteOptionValue Action](#).

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/DeleteOptionValue HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json  
  
{  
    "AttributeLogicalName": "sample_choice",  
    "EntityLogicalName": "sample_bankaccount",  
    "Value": 727000004  
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent  
OData-Version: 4.0
```

Salida de la consola:

```
Deleting a local option value...  
Local OptionSet option value deleted.
```

Columna de lista de selección múltiple

1. Cree una columna de opciones de selección múltiple usando [MultiSelectPicklistAttributeMetadata EntityType](#).

Solicitud:

HTTP

```
POST [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
"@odata.type":
"Microsoft.Dynamics.CRM.MultiSelectPicklistAttributeMetadata",
"AttributeType": "Virtual",
"AttributeName": {
    "Value": "MultiSelectPicklistType"
},
"SourceTypeMask": 0,
"OptionSet": {
    "@odata.type": "Microsoft.Dynamics.CRM.OptionSetMetadata",
    "Options": [
        {
            "Value": 727000000,
            "Label": {
                "@odata.type": "Microsoft.Dynamics.CRM.Label",
                "LocalizedLabels": [
                    {
                        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                        "Label": "Appetizer",
                        "LanguageCode": 1033,
                        "IsManaged": false
                    }
                ],
                "UserLocalizedLabel": {
                    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                    "Label": "Appetizer",
                    "LanguageCode": 1033,
                    "IsManaged": false
                }
            }
        },
        {
            "Value": 727000001,
            "Label": {
                "@odata.type": "Microsoft.Dynamics.CRM.Label",
                "LocalizedLabels": [
                    {
                        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                        "Label": "Entree",
                        "LanguageCode": 1033,
                        "IsManaged": false
                    }
                ],
                "UserLocalizedLabel": {
                    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                    "Label": "Entree",
                    "LanguageCode": 1033,
                    "IsManaged": false
                }
            }
        }
    ]
}
```

```
        "Label": "Entree",
        "LanguageCode": 1033,
        "IsManaged": false
    }
}
},
{
"Value": 727000002,
"Label": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Dessert",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Dessert",
        "LanguageCode": 1033,
        "IsManaged": false
    }
}
}
],
"IsGlobal": false,
"OptionsetType": "Picklist"
},
"Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "MultiSelect Choice Attribute",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "MultiSelect Choice Attribute",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Sample MultiSelect Choice",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ]
}
```

```

        },
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Sample MultiSelect Choice",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"RequiredLevel": {
    "Value": "None",
    "CanBeChanged": false,
    "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
},
"SchemaName": "sample_MultiSelectChoice"
}

```

Respuesta:

HTTP

```

HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(2c1c3050-112a-ed11-9db1-00224804f8e2)

```

Salida de la consola:

```

Creating a MultiSelect Choice column...
Created MultiSelect Choice column with id:2c1c3050-112a-ed11-9db1-
00224804f8e2

```

2. Recupere las opciones de la columna de selección múltiple usando `GET`

```

EntityDefinitions(LogicalName='sample_bankaccount')/Attributes(LogicalName='sa
mple_multiselectchoice')/Microsoft.Dynamics.CRM.MultiSelectPicklistAttributeMe
tadata?$select=SchemaName&$expand=OptionSet

```

Salida de la consola:

```

The option values for the multi-select choice column:
Value: 727000000, Label:Appetizer

```

```
Value: 727000001, Label:Entree  
Value: 727000002, Label:Dessert
```

Insertar valor de estado

Use [InsertStatusValue Action](#) para agregar una nueva opción a una columna `statuscode`. Debe especificar para qué `StateCode` es válido.

ⓘ Nota

Observe que el valor devuelto se aplica al valor del editor `customizationoptionvalueprefix` (72700) automáticamente.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/InsertStatusValue HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json  
  
{  
    "AttributeLogicalName": "statuscode",  
    "EntityLogicalName": "sample_bankaccount",  
    "Label": {  
        "@odata.type": "Microsoft.Dynamics.CRM.Label",  
        "LocalizedLabels": [  
            {  
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",  
                "Label": "Frozen",  
                "LanguageCode": 1033,  
                "IsManaged": false  
            }  
        ],  
        "UserLocalizedLabel": {  
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",  
            "Label": "Frozen",  
            "LanguageCode": 1033,  
            "IsManaged": false  
        }  
    },  
    "StateCode": 1,  
    "SolutionUniqueName": "examplesolution"  
}
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.InsertStatusValueRespons
e",
    "NewOptionValue": 727000000
}
```

Salida de la consola:

```
Created new Status value:727000000
```

Sección 3: Crear y usar OptionSet global

1. Cree una opción global (conjunto de opciones). Este conjunto de opciones se llama `sample_colors` y contiene opciones para rojo, amarillo y verde.

Nota

La solicitud especifica los valores de opción utilizando el editor `customizationoptionvalueprefix`.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/GlobalOptionSetDefinitions
HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
    "@odata.type": "Microsoft.Dynamics.CRM.OptionSetMetadata",
    "Options": [
        {
            "Name": "Red"
        },
        {
            "Name": "Yellow"
        },
        {
            "Name": "Green"
        }
    ]
}
```

```
"Value": 727000000,
"Label": {
"@odata.type": "Microsoft.Dynamics.CRM.Label",
"LocalizedLabels": [
{
"@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
"Label": "Red",
"LanguageCode": 1033,
"IsManaged": false
}
],
"UserLocalizedLabel": {
"@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
"Label": "Red",
"LanguageCode": 1033,
"IsManaged": false
}
}
},
{
"Value": 727000001,
"Label": {
"@odata.type": "Microsoft.Dynamics.CRM.Label",
"LocalizedLabels": [
{
"@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
"Label": "Yellow",
"LanguageCode": 1033,
"IsManaged": false
}
],
"UserLocalizedLabel": {
"@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
"Label": "Yellow",
"LanguageCode": 1033,
"IsManaged": false
}
}
},
{
"Value": 727000002,
"Label": {
"@odata.type": "Microsoft.Dynamics.CRM.Label",
"LocalizedLabels": [
{
"@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
"Label": "Green",
"LanguageCode": 1033,
"IsManaged": false
}
],
"UserLocalizedLabel": {
"@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
"Label": "Green",
"LanguageCode": 1033,
```

```

        "IsManaged": false
    }
}
]
,
"Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Color Choice",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Color Choice",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Colors",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Colors",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"Name": "sample_colors",
"OptionsetType": "Picklist"
}

```

Respuesta:

HTTP

```

HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/GlobalOptionSetDefinitions(7cf8c56-112a-ed11-9db1-
00224804f8e2)

```

Salida de la consola:

```
Created a new global option set with id:7cf8c56-112a-ed11-9db1-00224804f8e2
```

2. Recupere las opciones de elección global.

Solicitud:

HTTP

```
GET [Organization  
Uri]/api/data/v9.2/GlobalOptionSetDefinitions(7cf8c56-112a-ed11-9db1-00224804f8e2)/Microsoft.Dynamics.CRM.OptionSetMetadata HTTP/1.1  
Consistency: Strong  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK  
OData-Version: 4.0  
  
{  
    "@odata.context": "[Organization  
Uri]/api/data/v9.2/$metadata#GlobalOptionSetDefinitions/Microsoft.Dynamics.CRM.OptionSetMetadata/$entity",  
    "ParentOptionSetName": null,  
    "IsCustomOptionSet": true,  
    "IsGlobal": true,  
    "IsManaged": false,  
    "Name": "sample_colors",  
    "ExternalTypeName": null,  
    "OptionSetType": "Picklist",  
    "IntroducedVersion": "1.0.0.0",  
    "MetadataId": "7cf8c56-112a-ed11-9db1-00224804f8e2",  
    "HasChanged": null,  
    "Options": [  
        {  
            "Value": 727000000,  
            "Color": null,  
            "IsManaged": false,  
            "ExternalValue": "",  
            "ParentValues": [],  
            "MetadataId": null,
```

```
"HasChanged": null,
"Label": {
"LocalizedLabels": [
{
"Label": "Red",
"LanguageCode": 1033,
"IsManaged": false,
"MetadataId": "2c1fa94f-3714-4615-995b-690158d0d989",
"HasChanged": null
}
],
"UserLocalizedLabel": {
"Label": "Red",
"LanguageCode": 1033,
"IsManaged": false,
"MetadataId": "2c1fa94f-3714-4615-995b-690158d0d989",
"HasChanged": null
}
},
"Description": {
"LocalizedLabels": [],
"UserLocalizedLabel": null
}
},
{
"Value": 727000001,
"Color": null,
"IsManaged": false,
"ExternalValue": "",
"ParentValues": [],
"MetadataId": null,
"HasChanged": null,
"Label": {
"LocalizedLabels": [
{
"Label": "Yellow",
"LanguageCode": 1033,
"IsManaged": false,
"MetadataId": "a499c2fe-c13a-4c1e-b190-db8ae74396f5",
"HasChanged": null
}
],
"UserLocalizedLabel": {
"Label": "Yellow",
"LanguageCode": 1033,
"IsManaged": false,
"MetadataId": "a499c2fe-c13a-4c1e-b190-db8ae74396f5",
"HasChanged": null
}
},
"Description": {
"LocalizedLabels": [],
"UserLocalizedLabel": null
}
},
```

```
{
    "Value": 727000002,
    "Color": null,
    "IsManaged": false,
    "ExternalValue": "",
    "ParentValues": [],
    "MetadataId": null,
    "HasChanged": null,
    "Label": {
        "LocalizedLabels": [
            {
                "Label": "Green",
                "LanguageCode": 1033,
                "IsManaged": false,
                "MetadataId": "8378af2c-4b68-4ea4-ad37-e676f696e1ba",
                "HasChanged": null
            }
        ],
        "UserLocalizedLabel": {
            "Label": "Green",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "8378af2c-4b68-4ea4-ad37-e676f696e1ba",
            "HasChanged": null
        }
    },
    "Description": {
        "LocalizedLabels": [],
        "UserLocalizedLabel": null
    }
},
],
"Description": {
    "LocalizedLabels": [
        {
            "Label": "Color Choice",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "7efd8c56-112a-ed11-9db1-00224804f8e2",
            "HasChanged": null
        }
    ],
    "UserLocalizedLabel": {
        "Label": "Color Choice",
        "LanguageCode": 1033,
        "IsManaged": false,
        "MetadataId": "7efd8c56-112a-ed11-9db1-00224804f8e2",
        "HasChanged": null
    }
},
},
"DisplayName": {
    "LocalizedLabels": [
        {
            "Label": "Colors",
            "LanguageCode": 1033,
```

```

        "IsManaged": false,
        "MetadataId": "7dfd8c56-112a-ed11-9db1-00224804f8e2",
        "HasChanged": null
    }
],
"UserLocalizedLabel": {
    "Label": "Colors",
    "LanguageCode": 1033,
    "IsManaged": false,
    "MetadataId": "7dfd8c56-112a-ed11-9db1-00224804f8e2",
    "HasChanged": null
}
},
"IsCustomizable": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "iscustomizable"
}
}

```

Salida de la consola:

```

List the retrieved options for the colors global option set:
Value: 727000000 Label:Red
Value: 727000001 Label:Yellow
Value: 727000002 Label:Green

```

- Cree una columna de opciones que el conjunto de opciones global. Asocie la definición de columna al conjunto de opciones global usando:

```
"GlobalOptionSet@odata.bind": "/GlobalOptionSetDefinitions(7cf8c56-112a-ed11-9db1-00224804f8e2)".
```

Solicitud:

HTTP

```

POST [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
"@odata.type": "Microsoft.Dynamics.CRM.PicklistAttributeMetadata",
"AttributeType": "Picklist",

```

```

"AttributeType": {
    "Value": "PicklistType"
},
"SourceTypeMask": 0,
"GlobalOptionSet@odata.bind": "/GlobalOptionSetDefinitions(7cf8c56-112a-ed11-9db1-00224804f8e2)",
"Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Colors Global Picklist Attribute",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Colors Global Picklist Attribute",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Sample Colors",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Sample Colors",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"RequiredLevel": {
    "Value": "None",
    "CanBeChanged": false,
    "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
},
"SchemaName": "sample_Colors"
}

```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(81fd8c56-112a-ed11-9db1-00224804f8e2)
```

Salida de la consola:

```
Created Choice column with id:81fd8c56-112a-ed11-9db1-00224804f8e2
using colors global optionset.
```

Sección 4: Crear relación con el cliente

1. Use [CreateCustomerRelationships Action](#) para crear una relación con el cliente.

Esta acción agrega una columna de búsqueda para la tabla `sample_BankAccount` que permite establecer un registro `account` o `contact`.

`CreateCustomerRelationships` tiene un parámetro `Lookup` [ComplexLookupAttributeMetadata ComplexType](#) y un parámetro `OneToManyRelationships` que contienen un par de relaciones definidas mediante [ComplexOneToManyRelationshipMetadata ComplexType](#).

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/CreateCustomerRelationships
HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
  "Lookup": {
    "@odata.type":
      "Microsoft.Dynamics.CRM.ComplexLookupAttributeMetadata",
    "AttributeType": "Lookup",
    "AttributeName": {
      "Value": "LookupType"
    },
    "Format": "None",
    "Targets": [
      "account",
      "contact"
    ]
  }
}
```

```
        "contact"
    ],
    "ColumnNumber": 0,
    "Description": {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [
            {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "The owner of the bank account",
                "LanguageCode": 1033,
                "IsManaged": false
            }
        ],
        "UserLocalizedLabel": {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "The owner of the bank account",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    },
    "DisplayName": {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [
            {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Bank Account owner",
                "LanguageCode": 1033,
                "IsManaged": false
            }
        ],
        "UserLocalizedLabel": {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Bank Account owner",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    },
    "RequiredLevel": {
        "Value": "ApplicationRequired",
        "CanBeChanged": false,
        "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
    },
    "SchemaName": "sample_CustomerId",
    "SourceType": 0
},
"OneToManyRelationships": [
    {
        "@odata.type": "Microsoft.Dynamics.CRM.ComplexOneToManyRelationshipMetadata",
        "ReferencedEntity": "account",
        "ReferencingEntity": "sample_bankaccount",
        "RelationshipBehavior": 0,
        "RelationshipType": "OneToManyRelationship",
        "SchemaName": "sample_BankAccount_Customer_Account",
        "SecurityTypes": "None"
```

```

},
{
  "@odata.type": "Microsoft.Dynamics.CRM.ComplexOneToManyRelationshipMetadata",
  "ReferencedEntity": "contact",
  "ReferencingEntity": "sample_bankaccount",
  "RelationshipBehavior": 0,
  "RelationshipType": "OneToManyRelationship",
  "SchemaName": "sample_BankAccount_Customer_Contact",
  "SecurityTypes": "None"
}
],
"SolutionUniqueName": "examplesolution"
}

```

Respuesta:

HTTP

```

HTTP/1.1 200 OK
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.CreateCustomerRelat
ionshipsResponse",
  "RelationshipIds": [
    "84fd8c56-112a-ed11-9db1-00224804f8e2",
    "8dfd8c56-112a-ed11-9db1-00224804f8e2"
  ],
  "AttributeId": "59478264-16af-4bcc-8baa-b154df0d6767"
}

```

2. Use el `CreateCustomerRelationshipsResponse.AttributeId` Valor para recuperar la propiedad `Targets` de la columna de búsqueda para la relación con el cliente.

Solicitud:

HTTP

```

GET [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(LogicalName='sample_customerid')/Microsoft.Dynamics.CRM.Look
upAttributeMetadata?$select=Targets HTTP/1.1
Consistency: Strong
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#EntityDefinitions('sample_bankaccount')/At
tributes/Microsoft.Dynamics.CRM.LookupAttributeMetadata(Targets)/$entit
y",
  "Targets": [
    "account",
    "contact"
  ],
  "MetadataId": "59478264-16af-4bcc-8baa-b154df0d6767"
}
```

Salida de la consola:

```
The Target values of the Lookup column created:
  account
  contact
```

3. Use el `CreateCustomerRelationshipsResponse.RelationshipIds` valores para recuperar las relaciones para la columna del cliente.

Primero para la relación con `account`:

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/RelationshipDefinitions(84fd8c56-
112a-ed11-9db1-
00224804f8e2)/Microsoft.Dynamics.CRM.OneToManyRelationshipMetadata
HTTP/1.1
Consistency: Strong
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#RelationshipDefinitions/Microsoft.Dynamics
.CRM.OneToManyRelationshipMetadata/$entity",
    "ReferencedAttribute": "accountid",
    "ReferencedEntity": "account",
    "ReferencingAttribute": "sample_customerid",
    "ReferencingEntity": "sample_bankaccount",
    "IsHierarchical": false,
    "EntityKey": null,
    "IsRelationshipAttributeDenormalized": false,
    "ReferencedEntityNavigationPropertyName":
        "sample_BankAccount_Customer_Account",
    "ReferencingEntityNavigationPropertyName": "sample_CustomerId_account",
    "RelationshipBehavior": 1,
    "IsDenormalizedLookup": null,
    "DenormalizedAttributeName": null,
    "IsCustomRelationship": true,
    "IsValidForAdvancedFind": true,
    "SchemaName": "sample_BankAccount_Customer_Account",
    "SecurityTypes": "Append",
    "IsManaged": false,
    "RelationshipType": "OneToManyRelationship",
    "IntroducedVersion": "1.0.0.0",
    "MetadataId": "84fd8c56-112a-ed11-9db1-00224804f8e2",
    "HasChanged": null,
    "AssociatedMenuConfiguration": {
        "Behavior": "UseCollectionName",
        "Group": "Details",
        "Order": 10000,
        "IsCustomizable": true,
        "Icon": null,
        "ViewId": "00000000-0000-0000-0000-000000000000",
        "AvailableOffline": true,
        "MenuItem": null,
        "QueryApi": null,
        "Label": {
            "LocalizedLabels": [],
            "UserLocalizedLabel": null
        }
    },
    "CascadeConfiguration": {
        "Assign": "NoCascade",
        "Delete": "RemoveLink",
        "Archive": "RemoveLink",
        "Merge": "Cascade",
        "Reparent": "NoCascade",
        "Share": "NoCascade",
        "Unshare": "NoCascade"
    }
}
```

```
        "RollupView": "NoCascade"
    },
    "RelationshipAttributes": [],
    "IsCustomizable": {
        "Value": true,
        "CanBeChanged": true,
        "ManagedPropertyLogicalName": "iscustomizable"
    }
}
```

Luego para la relación con `contact`:

Solicitud:

```
HTTP

GET [Organization Uri]/api/data/v9.2/RelationshipDefinitions(8dfd8c56-112a-ed11-9db1-00224804f8e2)/Microsoft.Dynamics.CRM.OneToManyRelationshipMetadata
HTTP/1.1
Consistency: Strong
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

```
HTTP

HTTP/1.1 200 OK
OData-Version: 4.0

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#RelationshipDefinitions/Microsoft.Dynamics
.CRM.OneToManyRelationshipMetadata/$entity",
    "ReferencedAttribute": "contactid",
    "ReferencedEntity": "contact",
    "ReferencingAttribute": "sample_customerid",
    "ReferencingEntity": "sample_bankaccount",
    "IsHierarchical": false,
    "EntityKey": null,
    "IsRelationshipAttributeDenormalized": false,
    "ReferencedEntityNavigationPropertyName":
        "sample_BankAccount_Customer_Contact",
    "ReferencingEntityNavigationPropertyName": "sample_CustomerId_contact",
    "RelationshipBehavior": 1,
    "IsDenormalizedLookup": null,
    "DenormalizedAttributeName": null,
    "IsCustomRelationship": true,
```

```

    "IsValidForAdvancedFind": true,
    "SchemaName": "sample_BankAccount_Customer_Contact",
    "SecurityTypes": "Append",
    "IsManaged": false,
    "RelationshipType": "OneToManyRelationship",
    "IntroducedVersion": "1.0.0.0",
    "MetadataId": "8dfd8c56-112a-ed11-9db1-00224804f8e2",
    "HasChanged": null,
    "AssociatedMenuConfiguration": {
        "Behavior": "UseCollectionName",
        "Group": "Details",
        "Order": 10000,
        "IsCustomizable": true,
        "Icon": null,
        "ViewId": "00000000-0000-0000-0000-000000000000",
        "AvailableOffline": true,
        "MenuItem": null,
        "QueryApi": null,
        "Label": {
            "LocalizedLabels": [],
            "UserLocalizedLabel": null
        }
    },
    "CascadeConfiguration": {
        "Assign": "NoCascade",
        "Delete": "RemoveLink",
        "Archive": "RemoveLink",
        "Merge": "Cascade",
        "Reparent": "NoCascade",
        "Share": "NoCascade",
        "Unshare": "NoCascade",
        "RollupView": "NoCascade"
    },
    "RelationshipAttributes": [],
    "IsCustomizable": {
        "Value": true,
        "CanBeChanged": true,
        "ManagedPropertyLogicalName": "iscustomizable"
    }
}

```

Salida de la consola:

```

The Schema Names of the relationships created:
sample_BankAccount_Customer_Account
sample_BankAccount_Customer_Contact

```

Sección 5: Crear y recuperar una relación de uno a muchos

Antes de crear una relación mediante código, debe confirmar que la relación es válida. Los diseñadores de [Power Apps](#) usan funciones especiales para mostrarle qué combinaciones son válidas. Puede usar las mismas funciones en su código para detectar si una relación en particular se puede crear o no.

Validar la idoneidad de la relación de 1:N

1. [CanBeReferenced Function](#) le indica si una tabla puede ser la tabla principal (uno) en una relación de uno a varios.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/CanBeReferenced HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
  "EntityName": "sample_bankaccount"
}
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.CanBeReferencedRespon
se",
  "CanBeReferenced": true
}
```

Salida de la consola:

The sample_BankAccount table is eligible to be a primary table in a one-to-many relationship.

2. [CanBeReferencing Function](#) indica si la tabla especificada puede ser la tabla de referencia en una relación de uno a varios. La tabla de referencia es la tabla a la que se le ha agregado una columna de búsqueda para que sea "varios" en la relación de uno a varios.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/CanBeReferencing HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
  "EntityName": "contact"
}
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.CanBeReferencingRes
ponse",
  "CanBeReferencing": true
}
```

Salida de la consola:

The contact table is eligible to be a related table in a one-to-many relationship.

Identificar entidades de referencia potenciales

En el contexto de una tabla específica que puede ser la tabla parmary en una relación de uno a muchos, use [GetValidReferencingEntities Function](#) para identificar qué otras tablas pueden ser las relacionadas con él.

Solicitud:

```
HTTP

GET [Organization
Uri]/api/data/v9.2/GetValidReferencingEntities(ReferencedEntityName='sample_
bankaccount') HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

```
HTTP

HTTP/1.1 200 OK
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.GetValidReferencingEntit
iesResponse",
  "EntityNames": [
    "msdyn_slakpi",
    "workflowbinary",
    "apisettings",
    "flowsession",
    "knowledgearticle",
    "socialprofile",
    "goal",
    "metric",
    "goalrollupquery",
    "mailbox",
    "position",
    "channelaccessprofile",
    "externalparty",
    "channelaccessprofilerule",
    "channelaccessprofileruleitem",
    "sample_bankaccount",
    "privilegesremovalsetting",
    "knowledgebaserecord",
    "msdyn_insightsstorevirtualentity",
    "aaduser",
    "sharepointdocument",
    "msfp_unsubscribedrecipient",
    "msdyn_dataflow",
    "flowmachineimage",
```

"queueitem",
"appointment",
"msdyn_federatedarticleincident",
"msfp_surveyresponse",
"msdyn_dataflowrefreshhistory",
"mailmergetemplate",
"contact",
"organizationdatasyncstate",
"bot",
"knowledgearticleviews",
"slaitem",
"msfp_question",
"category",
"connection",
"newprocess",
"msfp_survey",
"emailserverprofile",
"appnotification",
"feedback",
"activityfileattachment",
"organizationdatasyncsubscriptionentity",
"msdyn_nonrelationalds",
"expiredprocess",
"msfp_surveyinvite",
"msfp_alert",
"businessunit",
"msfp_alertrule",
"slakpiinstance",
"email",
"datasyncstate",
"msdyn_entityrefreshhistory",
"msdyn_componentlayerdatasource",
"account",
"kbarticle",
"systemuser",
"task",
"letter",
"reportcategory",
"phonecall",
"actioncard",
"msdyn_kmfederatedsearchconfig",
"featurecontrolsetting",
"translationprocess",
"recurringappointmentmaster",
"externalpartyitem",
"msdyn_aibdatasetfile",
"socialactivity",
"flowmachineimageversion",
"fax",
"msdyn_kbattachment",
"serviceplanmapping",
"msdyn_knowledgearticletemplate",
"msfp_emailtemplate",
"conversationtranscript",
"sharepointsite",

```

    "processstage",
    "msfp_localizedemailtemplate",
    "queue",
    "msdyn_richtextfile",
    "msdyn_serviceconfiguration",
    "team",
    "sharedlinksetting",
    "territory",
    "msdyn_federatedarticle",
    "msdyn_knowledgepersonalfilter",
    "sharepointdocumentlocation",
    "chat",
    "msfp_fileresponse",
    "msfp_satisfactionmetric",
    "msdyn_aibfeedbackloop",
    "msdyn_customcontrolextendedsettings",
    "msfp_surveyreminder",
    "msfp_questionresponse",
    "msfp_project"
]
}

```

Salida de la consola:

```
The contact table is in the list of potential referencing entities for sample_BankAccount.
```

Crear una relación de 1:N

La siguiente solicitud crea una relación de uno a varios entre `sample_BankAccount` y las tablas de contactos con una columna de búsqueda agregada a la tabla `contact`.

Solicitud:

HTTP

```

POST [Organization Uri]/api/data/v9.2/RelationshipDefinitions HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
    "@odata.type": "Microsoft.Dynamics.CRM.OneToManyRelationshipMetadata",
    "AssociatedMenuConfiguration": {
        "Behavior": "UseLabel",
        "Group": "Details",

```

```
"Label": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Cardholders",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Cardholders",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"Order": 10000,
"ViewId": "00000000-0000-0000-0000-000000000000"
},
"CascadeConfiguration": {
    "Assign": "NoCascade",
    "Delete": "RemoveLink",
    "Merge": "Cascade",
    "Reparent": "NoCascade",
    "Share": "NoCascade",
    "Unshare": "NoCascade",
    "RollupView": "NoCascade"
},
"IsHierarchical": false,
"ReferencedAttribute": "sample_bankaccountid",
"ReferencedEntity": "sample_bankaccount",
"ReferencingEntity": "contact",
"Lookup": {
    "@odata.type": "Microsoft.Dynamics.CRM.LookupAttributeMetadata",
    "AttributeType": "Lookup",
    "AttributeTypeName": {
        "Value": "LookupType"
    },
    "Format": "None",
    "Description": {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [
            {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "The bank account this contact has access to.",
                "LanguageCode": 1033,
                "IsManaged": false
            }
        ],
        "UserLocalizedLabel": {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "The bank account this contact has access to.",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    }
}
```

```

        },
        "DisplayName": {
            "@odata.type": "Microsoft.Dynamics.CRM.Label",
            "LocalizedLabels": [
                {
                    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                    "Label": "Bank Account",
                    "LanguageCode": 1033,
                    "IsManaged": false
                }
            ],
            "UserLocalizedLabel": {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Bank Account",
                "LanguageCode": 1033,
                "IsManaged": false
            }
        },
        "SchemaName": "sample_BankAccountId"
    },
    "IsCustomRelationship": false,
    "IsManaged": false,
    "IsValidForAdvancedFind": false,
    "RelationshipType": "OneToManyRelationship",
    "SchemaName": "sample_BankAccount_Contacts",
    "SecurityTypes": "None"
}

```

Respuesta:

HTTP

```

HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/RelationshipDefinitions(991efd5f-112a-ed11-9db1-
00224804f8e2)

```

Salida de la consola:

```

Creating a one-to-many relationship...
Created one-to-many relationship: RelationshipDefinitions(991efd5f-112a-
ed11-9db1-00224804f8e2)

```

Recuperar una relación de 1:N

La siguiente solicitud recupera la relación creada por la solicitud anterior.

ⓘ Nota

Porque `RelationshipDefinitions` contiene definiciones de relación de uno a muchos y de muchos a muchos, debe incluir lo siguiente en la URL para convertir al tipo que desea recuperar: `/Microsoft.Dynamics.CRM.OneToManyRelationshipMetadata`. De lo contrario, el valor devuelto será `RelationshipMetadataBase EntityType` y no incluirá las propiedades específicas del `OneToManyRelationshipMetadata EntityType`.

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/RelationshipDefinitions(991efd5f-112a-ed11-9db1-00224804f8e2)/Microsoft.Dynamics.CRM.OneToManyRelationshipMetadata
HTTP/1.1
Consistency: Strong
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#RelationshipDefinitions/Microsoft.Dynamics.CRM.
OneToManyRelationshipMetadata/$entity",
    "ReferencedAttribute": "sample_bankaccountid",
    "ReferencedEntity": "sample_bankaccount",
    "ReferencingAttribute": "sample_bankaccountid",
    "ReferencingEntity": "contact",
    "IsHierarchical": false,
    "EntityKey": null,
    "IsRelationshipAttributeDenormalized": false,
    "ReferencedEntityNavigationPropertyName": "sample_BankAccount_Contacts",
    "ReferencingEntityNavigationPropertyName": "sample_BankAccountId",
    "RelationshipBehavior": 1,
    "IsDenormalizedLookup": null,
    "DenormalizedAttributeName": null,
    "IsCustomRelationship": true,
```

```
"IsValidForAdvancedFind": false,
"SchemaName": "sample_BankAccount_Contacts",
"SecurityTypes": "Append",
"IsManaged": false,
"RelationshipType": "OneToManyRelationship",
"IntroducedVersion": "1.0.0.0",
"MetadataId": "991efd5f-112a-ed11-9db1-00224804f8e2",
"HasChanged": null,
"AssociatedMenuConfiguration": {
    "Behavior": "UseLabel",
    "Group": "Details",
    "Order": 10000,
    "IsCustomizable": true,
    "Icon": null,
    "ViewId": "00000000-0000-0000-0000-000000000000",
    "AvailableOffline": true,
    "MenuId": null,
    "QueryApi": null,
    "Label": {
        "LocalizedLabels": [
            {
                "Label": "Cardholders",
                "LanguageCode": 1033,
                "IsManaged": false,
                "MetadataId": "a99a420f-6778-4f2f-b42b-64bc84b2c2d2",
                "HasChanged": null
            }
        ],
        "UserLocalizedLabel": {
            "Label": "Cardholders",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "a99a420f-6778-4f2f-b42b-64bc84b2c2d2",
            "HasChanged": null
        }
    }
},
"CascadeConfiguration": {
    "Assign": "NoCascade",
    "Delete": "RemoveLink",
    "Archive": "RemoveLink",
    "Merge": "NoCascade",
    "Reparent": "NoCascade",
    "Share": "NoCascade",
    "Unshare": "NoCascade",
    "RollupView": "NoCascade"
},
"RelationshipAttributes": [],
"IsCustomizable": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "iscustomizable"
}
}
```

Salida de la consola:

```
Retrieved relationship: sample_BankAccount_Contacts
```

Sección 6: Crear y recuperar una relación de muchos a uno

Una relación de varios a uno es simplemente una relación uno a varios vista desde la otra dirección. Los siguientes ejemplos crean una columna de búsqueda denominada `sample_relatedaccountid` en la tabla `sample_BankAccount` que hace referencia a una fila en la tabla `account`.

Crear una relación de N:1

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/RelationshipDefinitions HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
    "@odata.type": "Microsoft.Dynamics.CRM.OneToManyRelationshipMetadata",
    "AssociatedMenuConfiguration": {
        "Behavior": "UseLabel",
        "Group": "Details",
        "Label": {
            "@odata.type": "Microsoft.Dynamics.CRM.Label",
            "LocalizedLabels": [
                {
                    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                    "Label": "Related Bank Accounts",
                    "LanguageCode": 1033,
                    "IsManaged": false
                }
            ],
            "UserLocalizedLabel": {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Related Bank Accounts",
                "LanguageCode": 1033,
                "IsManaged": false
            }
        }
    }
}
```

```
        },
    },
    "Order": 10000,
    "ViewId": "00000000-0000-0000-0000-000000000000"
},
"CascadeConfiguration": {
    "Assign": "NoCascade",
    "Delete": "RemoveLink",
    "Merge": "Cascade",
    "Reparent": "NoCascade",
    "Share": "NoCascade",
    "Unshare": "NoCascade",
    "RollupView": "NoCascade"
},
"IsHierarchical": false,
"ReferencedAttribute": "accountid",
"ReferencedEntity": "account",
"ReferencingEntity": "sample_bankaccount",
"Lookup": {
    "@odata.type": "Microsoft.Dynamics.CRM.LookupAttributeMetadata",
    "AttributeType": "Lookup",
    "AttributeName": {
        "Value": "LookupType"
    },
    "Format": "None",
    "Description": {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [
            {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "An Account related to the bank account.",
                "LanguageCode": 1033,
                "IsManaged": false
            }
        ],
        "UserLocalizedLabel": {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "An Account related to the bank account.",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    },
    "DisplayName": {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [
            {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Related Account",
                "LanguageCode": 1033,
                "IsManaged": false
            }
        ],
        "UserLocalizedLabel": {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Related Account",
            "LanguageCode": 1033
        }
    }
}
```

```
        "LanguageCode": 1033,  
        "IsManaged": false  
    }  
,  
    "SchemaName": "sample_RelatedAccountId"  
},  
"IsCustomRelationship": false,  
"IsManaged": false,  
"IsValidForAdvancedFind": false,  
"RelationshipType": "OneToManyRelationship",  
"SchemaName": "sample_Account_BankAccounts",  
"SecurityTypes": "None"  
}  
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent  
OData-Version: 4.0  
OData-EntityId: [Organization  
Uri]/api/data/v9.2/RelationshipDefinitions(0901c466-112a-ed11-9db1-  
00224804f8e2)
```

Salida de la consola:

```
Created many-to-one relationship: RelationshipDefinitions(0901c466-112a-  
ed11-9db1-00224804f8e2)
```

Recuperar una relación de N:1

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/RelationshipDefinitions(0901c466-112a-  
ed11-9db1-00224804f8e2)/Microsoft.Dynamics.CRM.OneToManyRelationshipMetadata  
HTTP/1.1  
Consistency: Strong  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#RelationshipDefinitions/Microsoft.Dynamics.CRM.
OneToManyRelationshipMetadata/$entity",
    "ReferencedAttribute": "accountid",
    "ReferencedEntity": "account",
    "ReferencingAttribute": "sample_relatedaccountid",
    "ReferencingEntity": "sample_bankaccount",
    "IsHierarchical": false,
    "EntityKey": null,
    "IsRelationshipAttributeDenormalized": false,
    "ReferencedEntityNavigationPropertyName": "sample_Account_BankAccounts",
    "ReferencingEntityNavigationPropertyName": "sample_RelatedAccountId",
    "RelationshipBehavior": 1,
    "IsDenormalizedLookup": null,
    "DenormalizedAttributeName": null,
    "IsCustomRelationship": true,
    "IsValidForAdvancedFind": false,
    "SchemaName": "sample_Account_BankAccounts",
    "SecurityTypes": "Append",
    "IsManaged": false,
    "RelationshipType": "OneToManyRelationship",
    "IntroducedVersion": "1.0.0.0",
    "MetadataId": "0901c466-112a-ed11-9db1-00224804f8e2",
    "HasChanged": null,
    "AssociatedMenuConfiguration": {
        "Behavior": "UseLabel",
        "Group": "Details",
        "Order": 10000,
        "IsCustomizable": true,
        "Icon": null,
        "ViewId": "00000000-0000-0000-0000-000000000000",
        "AvailableOffline": true,
        "MenuItem": null,
        "QueryApi": null,
        "Label": {
            "LocalizedLabels": [
                {
                    "Label": "Related Bank Accounts",
                    "LanguageCode": 1033,
                    "IsManaged": false,
                    "MetadataId": "b2ad2cce-5c7c-46aa-8bbf-7903d33ef019",
                    "HasChanged": null
                }
            ],
            "UserLocalizedLabel": {
                "Label": "Related Bank Accounts",
                "LanguageCode": 1033,
                "IsManaged": false,
            }
        }
    }
}
```

```
        "MetadataId": "b2ad2cce-5c7c-46aa-8bbf-7903d33ef019",
        "HasChanged": null
    }
},
{
    "CascadeConfiguration": {
        "Assign": "NoCascade",
        "Delete": "RemoveLink",
        "Archive": "RemoveLink",
        "Merge": "Cascade",
        "Reparent": "NoCascade",
        "Share": "NoCascade",
        "Unshare": "NoCascade",
        "RollupView": "NoCascade"
    },
    "RelationshipAttributes": [],
    "IsCustomizable": {
        "Value": true,
        "CanBeChanged": true,
        "ManagedPropertyLogicalName": "iscustomizable"
    }
}
```

Salida de la consola:

```
Retrieved relationship: sample_Account_BankAccounts
```

Sección 7: Crear y recuperar una relación de muchos a muchos

Al igual que uno a muchos Relaciones, hay funciones especiales utilizadas por los diseñadores en [Power Apps](#) para evitar combinaciones no válidas al crear muchos a muchos Relaciones.

Validar la idoneidad de la relación de N:N

1. [CanManyToMany Function](#) indica si una tabla puede participar en una relación de varios a varios. Entonces, esta solicitud prueba la tabla `contact`.

Solicitud:

```
HTTP
```

```
POST [Organization Uri]/api/data/v9.2/CanManyToMany HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{{
"EntityName": "contact"
}}
```

Respuesta:

```
HTTP

HTTP/1.1 200 OK
OData-Version: 4.0

{{
"@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.CanManyToManyRespon
se",
"CanManyToMany": true
}}
```

Salida de la consola:

```
The contact table can participate in many-to-many relationships.
```

2. Esta solicitud realiza la misma prueba en la tabla `sample_bankaccount`.

Solicitud:

```
HTTP

POST [Organization Uri]/api/data/v9.2/CanManyToMany HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{{
"EntityName": "sample_bankaccount"
}}
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.CanManyToManyRespon
se",
  "CanManyToMany": true
}
```

Salida de la consola:

```
The sample_bankaccount table can participate in many-to-many
relationships.
```

Identificar entidades potenciales para relaciones de N:N

Use [GetValidManyToMany Function](#) para obtener una lista de tablas que pueden participar en relaciones de varios a varios.

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/GetValidManyToMany HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.GetValidManyToManyRespon
se",
  "EntityNames": [
    "msdyn_slakpi",
    "workflowbinary",
```

```
"apisettings",
"flowsession",
"theme",
"knowledgearticle",
"socialprofile",
"goal",
"position",
"externalparty",
"channelaccessprofileruleitem",
"routingruleitem",
"sample_bankaccount",
"synapsealinkexternaltablestate",
"synapsesdatabase",
"msdyn_aimodel",
"aaduser",
"applicationuser",
"msfp_unsubscribedrecipient",
"msdyn_aiconfiguration",
"msdyn_dataflow",
"flowmachineimage",
"queueitem",
"synapselinkschedule",
"msdyn_federatedarticleincident",
"flowmachine",
"synapsealinkprofile",
"msdyn_dataflowrefreshhistory",
"solutioncomponentrelationshipconfiguration",
"contact",
"organizationdatasyncstate",
"botcomponent",
"bot",
"msdyn_componentlayer",
"msdyn_odatav4ds",
"msfp_question",
"msdyn_aibfile",
"msdyn_solutionhistorydatasource",
"msdyn_solutionhealthruleset",
"newprocess",
"connectionreference",
"msdyn_knowledgemanagementsetting",
"msdyn_pmrecording",
"msfp_survey",
"msdyn_aibdatasetscontainer",
"package",
"msdyn_solutioncomponentsummary",
"msdyn_helppage",
"appnotification",
"organizationdatasyncsubscriptionentity",
"msdyn_aidotrainingboundingbox",
"msdyn_nonrelationalds",
"expiredprocess",
"msdyn_analysisresultdetail",
"msfp_alertrule",
"msdyn_solutioncomponentcountsummary",
"msdyn_kalanguagesetting",
```

"transactioncurrency",
"exportsolutionupload",
"msdyn_pmprocessusersettings",
"datasyncstate",
"msdyn_entityrefreshhistory",
"msdyn_analysisresult",
"msdyn_componentlayerdatasource",
"account",
"kbarticle",
"systemuser",
"task",
"msdyn_analysisjob",
"solutioncomponentconfiguration",
"msdyn_knowledgesearchfilter",
"stagesolutionupload",
"msdyn_pmttemplate",
"phonecall",
"msdyn_solutioncomponentdatasource",
"environmentvariablevalue",
"msdyn_aitemplate",
"userrating",
"synapsealinkprofileentity",
"featurecontrolsetting",
"translationprocess",
"msdyn_pminferredtask",
"customapirequestparameter",
"externalpartyitem",
"msdyn_aibdatasetfile",
"flowmachineingroup",
"flowmachineimageversion",
"msdyn_aibdatasetrecord",
"msdyn_kbattachment",
"msdyn_aifptrainingdocument",
"customapiresponseproperty",
"msdyn_knowledgearticletemplate",
"msdyn_aiodimage",
"msdyn_knowledgesearchinsight",
"msfp_emailtemplate",
"catalog",
"msdyn_knowledgeinteractioninsight",
"conversationtranscript",
"msdyn_pmanalyshistory",
"msdyn_datalakeds",
"canvasappextendedmetadata",
"msfp_localizedemailtemplate",
"msdynce_botcontent",
"queue",
"msdyn_solutionhealthruleargument",
"msdyn_aibfileattacheddata",
"msdyn_richtextfile",
"msdyn_kmpersonalizationsetting",
"msdyn_aiodtrainingimage",
"msdyn_serviceconfiguration",
"msdyn_knowledgearticleimage",
"team",

```

    "territory",
    "msdyn_solutioncomponentcountdatasource",
    "catalogassignment",
    "msdyn_federatedarticle",
    "msdyn_solutionhealthrule",
    "msdyn_solutionhistory",
    "msdyn_knowledgepersonalfilter",
    "organizationdatasyncsubscription",
    "solutioncomponentbatchconfiguration",
    "connector",
    "solutioncomponentattributeconfiguration",
    "synapsealinkprofileentitystate",
    "msdyn_aiodlabel",
    "customapi",
    "msdyn_aibdataset",
    "msfp_fileresponse",
    "environmentvariabledefinition",
    "msdyn_analysiscomponent",
    "msfp_satisfactionmetric",
    "msdyn_tour",
    "msdyn_customcontrolextendedsettings",
    "msfp_surveyreminder",
    "virtualentitymetadata",
    "msfp_questionresponse",
    "msfp_project"
]
}

```

Salida de la consola:

```

Contact is in the list of potential tables for N:N.
sample_BankAccount is in the list of potential tables for N:N.

```

Crear una relación de N:N

Esta solicitud crea una relación de varios a varios entre las tablas `sample_BankAccount` y `Contact`.

Solicitud:

HTTP

```

POST [Organization Uri]/api/data/v9.2/RelationshipDefinitions HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

```

```
{  
    "@odata.type": "Microsoft.Dynamics.CRM.ManyToManyRelationshipMetadata",  
    "Entity1AssociatedMenuConfiguration": {  
        "Behavior": "UseLabel",  
        "Group": "Details",  
        "Label": {  
            "@odata.type": "Microsoft.Dynamics.CRM.Label",  
            "LocalizedLabels": [  
                {  
                    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",  
                    "Label": "Bank Accounts",  
                    "LanguageCode": 1033,  
                    "IsManaged": false  
                }  
            ],  
            "UserLocalizedLabel": {  
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",  
                "Label": "Bank Accounts",  
                "LanguageCode": 1033,  
                "IsManaged": false  
            }  
        },  
        "Order": 10000,  
        "ViewId": "00000000-0000-0000-0000-000000000000"  
    },  
    "Entity1LogicalName": "sample_bankaccount",  
    "Entity2AssociatedMenuConfiguration": {  
        "Behavior": "UseLabel",  
        "Group": "Details",  
        "Label": {  
            "@odata.type": "Microsoft.Dynamics.CRM.Label",  
            "LocalizedLabels": [  
                {  
                    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",  
                    "Label": "Contacts",  
                    "LanguageCode": 1033,  
                    "IsManaged": false  
                }  
            ],  
            "UserLocalizedLabel": {  
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",  
                "Label": "Contacts",  
                "LanguageCode": 1033,  
                "IsManaged": false  
            }  
        },  
        "Order": 10000,  
        "ViewId": "00000000-0000-0000-0000-000000000000"  
    },  
    "Entity2LogicalName": "contact",  
    "IntersectEntityName": "sample_sample_BankAccounts_Contacts",  
    "IsCustomRelationship": false,  
    "IsManaged": false,  
    "IsValidForAdvancedFind": false,  
}
```

```
"RelationshipType": "OneToManyRelationship",
"SchemaName": "sample_sample_BankAccounts_Contacts",
"SecurityTypes": "None"
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/RelationshipDefinitions(55c9f86c-112a-ed11-9db1-
00224804f8e2)
```

Salida de la consola:

```
Created Many-to-Many relationship at: RelationshipDefinitions(55c9f86c-112a-
ed11-9db1-00224804f8e2)
```

Recuperar una relación N:N

Esta solicitud recupera la relación varios a varios creada por la solicitud anterior.

ⓘ Nota

Como se mencionaba antes, como `RelationshipDefinitions` contiene definiciones de relación de uno a muchos y de muchos a muchos, debe incluir lo siguiente en la URL para convertir al tipo que desea recuperar:

`/Microsoft.Dynamics.CRM.ManyToManyRelationshipMetadata`. De lo contrario, el valor devuelto será `RelationshipMetadataBase EntityType` y no incluirá las propiedades específicas del `ManyToManyRelationshipMetadata EntityType`.

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/RelationshipDefinitions(55c9f86c-112a-
ed11-9db1-
00224804f8e2)/Microsoft.Dynamics.CRM.ManyToManyRelationshipMetadata HTTP/1.1
Consistency: Strong
OData-MaxVersion: 4.0
```

```
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

HTTP/1.1 200 OK

OData-Version: 4.0

```
{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#RelationshipDefinitions/Microsoft.Dynamics.CRM.
ManyToManyRelationshipMetadata/$entity",
  "Entity1LogicalName": "sample_bankaccount",
  "Entity2LogicalName": "contact",
  "IntersectEntityName": "sample_sample_bankaccounts_contacts",
  "Entity1IntersectAttribute": "sample_bankaccountid",
  "Entity2IntersectAttribute": "contactid",
  "Entity1NavigationPropertyName": "sample_sample_BankAccounts_Contacts",
  "Entity2NavigationPropertyName": "sample_sample_BankAccounts_Contacts",
  "IsCustomRelationship": true,
  "IsValidForAdvancedFind": false,
  "SchemaName": "sample_sample_BankAccounts_Contacts",
  "SecurityTypes": "None",
  "IsManaged": false,
  "RelationshipType": "ManyToManyRelationship",
  "IntroducedVersion": "1.0.0.0",
  "MetadataId": "55c9f86c-112a-ed11-9db1-00224804f8e2",
  "HasChanged": null,
  "Entity1AssociatedMenuConfiguration": {
    "Behavior": "UseLabel",
    "Group": "Details",
    "Order": 10000,
    "IsCustomizable": true,
    "Icon": null,
    "ViewId": "00000000-0000-0000-0000-000000000000",
    "AvailableOffline": true,
    "MenuItemId": null,
    "QueryApi": null,
    "Label": {
      "LocalizedLabels": [
        {
          "Label": "Bank Accounts",
          "LanguageCode": 1033,
          "IsManaged": false,
          "MetadataId": "271999d1-6fd9-4413-b027-f2a1b231e0a4",
          "HasChanged": null
        }
      ],
      "UserLocalizedLabel": {
        "Label": "Bank Accounts",
        "LanguageCode": 1033
      }
    }
  }
}
```

```

        "LanguageCode": 1033,
        "IsManaged": false,
        "MetadataId": "271999d1-6fd9-4413-b027-f2a1b231e0a4",
        "HasChanged": null
    }
}
},
"Entity2AssociatedMenuConfiguration": {
    "Behavior": "UseLabel",
    "Group": "Details",
    "Order": 10000,
    "IsCustomizable": true,
    "Icon": null,
    "ViewId": "00000000-0000-0000-0000-000000000000",
    "AvailableOffline": true,
    "MenuItem": null,
    "QueryApi": null,
    "Label": {
        "LocalizedLabels": [
            {
                "Label": "Contacts",
                "LanguageCode": 1033,
                "IsManaged": false,
                "MetadataId": "1fcff441-9a41-42b1-a0d9-e92daa47230f",
                "HasChanged": null
            }
        ],
        "UserLocalizedLabel": {
            "Label": "Contacts",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "1fcff441-9a41-42b1-a0d9-e92daa47230f",
            "HasChanged": null
        }
    }
},
"IsCustomizable": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "iscustomizable"
}
}

```

Salida de la consola:

```
Retrieved Many-to-Many relationship:sample_sample_BankAccounts_Contacts
```

Sección 8: Exportar una solución administrada

Use [ExportSolution Action](#) para exportar la solución como administrada. Esta acción incluye muchos modificadores que puede usar para incluir información adicional como parte de la solución, pero en este caso, todas esas opciones están desactivadas. Más información: [Trabajar con soluciones](#)

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/ExportSolution HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
    "SolutionName": "examplesolution",
    "Managed": true,
    "ExportAutoNumberingSettings": false,
    "ExportCalendarSettings": false,
    "ExportCustomizationSettings": false,
    "ExportEmailTrackingSettings": false,
    "ExportGeneralSettings": false,
    "ExportMarketingSettings": false,
    "ExportOutlookSynchronizationSettings": false,
    "ExportRelationshipRoles": false,
    "ExportIsvConfig": false,
    "ExportSales": false,
    "ExportExternalApplications": false
}
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.ExportSolutionResponse",
    "ExportSolutionFile": "[ Binary content removed for brevity]"
}
```

Salida de la consola:

```
Solution Exported to E:\GitHub\PowerApps-Samples\dataverse\webapi\C#-
```

Sección 9: Eliminar registros de muestra

Las referencias a todos los registros creados en esta muestra se han agregado a una lista. En esta sección, todos los registros creados se eliminan mediante una operación `$batch`.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/$batch HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

--batch_d6cb9873-6b53-490d-942d-0bea79b8ba9a
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-Length: 136

DELETE /api/data/v9.2/RelationshipDefinitions(991efd5f-112a-ed11-9db1-
00224804f8e2) HTTP/1.1

--batch_d6cb9873-6b53-490d-942d-0bea79b8ba9a
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-Length: 130

DELETE /api/data/v9.2/EntityDefinitions(5872b902-112a-ed11-9db1-
00224804f8e2) HTTP/1.1

--batch_d6cb9873-6b53-490d-942d-0bea79b8ba9a
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-Length: 122

DELETE /api/data/v9.2/solutions(5472b902-112a-ed11-9db1-00224804f8e2)
HTTP/1.1

--batch_d6cb9873-6b53-490d-942d-0bea79b8ba9a
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-Length: 123

DELETE /api/data/v9.2/publishers(a78ab7fc-102a-ed11-9db1-00224804f8e2)
```

HTTP/1.1

```
--batch_d6cb9873-6b53-490d-942d-0bea79b8ba9a
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-Length: 139

DELETE /api/data/v9.2/GlobalOptionSetDefinitions(7cf8c56-112a-ed11-9db1-
00224804f8e2) HTTP/1.1

--batch_d6cb9873-6b53-490d-942d-0bea79b8ba9a--
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

--batchresponse_9fa66f62-38d1-4890-91ce-4185fd556745
Content-Type: application/http
Content-Transfer-Encoding: binary

HTTP/1.1 204 No Content
OData-Version: 4.0

--batchresponse_9fa66f62-38d1-4890-91ce-4185fd556745
Content-Type: application/http
Content-Transfer-Encoding: binary

HTTP/1.1 204 No Content
OData-Version: 4.0

--batchresponse_9fa66f62-38d1-4890-91ce-4185fd556745
Content-Type: application/http
Content-Transfer-Encoding: binary

HTTP/1.1 204 No Content
OData-Version: 4.0
```

```
--batchresponse_9fa66f62-38d1-4890-91ce-4185fd556745
Content-Type: application/http
Content-Transfer-Encoding: binary

HTTP/1.1 204 No Content
OData-Version: 4.0

--batchresponse_9fa66f62-38d1-4890-91ce-4185fd556745--
```

Salida de la consola:

```
Deleting created records...
```

Sección 10: Importar y eliminar una solución administrada

- Este paso importa solución administrada exportado en [Sección 8: Exportar solución administrada](#) utilizando el [ImportSolution Action](#).

Solicitud:

```
HTTP
POST [Organization Uri]/api/data/v9.2/ImportSolution HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{{
    "OverwriteUnmanagedCustomizations": false,
    "PublishWorkflows": false,
    "CustomizationFile": "[ Binary content removed for brevity]",
    "ImportJobId": "00000000-0000-0000-0000-000000000000"
}}
```

Respuesta:

```
HTTP
HTTP/1.1 204 NoContent
```

OData-Version: 4.0

Salida de la consola:

```
Sending request to import the examplesolution solution...
Solution imported as a managed solution.
```

2. Obtenga el `solutionid` de la solución administrada por `uniquename` para que pueda eliminarlo.

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/solutions?
$select=solutionid&$filter=uniquename%20eq%20'examplesolution' HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#solutions(solutionid)",
  "value": [
    {
      "@odata.etag": "W/\\"13291034\\\"",
      "solutionid": "07439497-6992-4e30-81e0-628a91984af5"
    }
  ]
}
```

Eliminar una solución administrada

Este paso final elimina el solución administrada importado para devolver el sistema al estado original.

Solicitud:

HTTP

```
DELETE [Organization Uri]/api/data/v9.2/solutions(07439497-6992-4e30-81e0-  
628a91984af5) HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent  
OData-Version: 4.0
```

Salida de la consola:

```
Sending request to delete the examplesolution solution...  
Managed solution deleted.  
--Metadata Operations sample completed--
```

Consulte también

[Usar la API web de Dataverse](#)

[Utilizar la API web con definiciones de tabla](#)

[Consultar definiciones de tabla con la API web](#)

[Crear y actualizar definiciones de tabla mediante la API web](#)

[Crear y actualizar relaciones de tabla mediante la API web](#)

[Crear y actualizar selecciones \(conjuntos de opciones\) mediante la API web](#)

[Ejemplo de operaciones de metadatos de API web \(C#\)](#)

ⓘ Nota

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta. [↗](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad ↗](#)).

Ejemplo de operaciones de metadatos de API web (C#)

Artículo • 23/02/2023

Este ejemplo de .NET 6.0 muestra cómo realizar operaciones que crean y modifican definiciones de tablas, columnas y relaciones mediante la API web de Dataverse.

Esta muestra utiliza el código auxiliar común de la [biblioteca de clases WebAPIService \(C#\)](#).

ⓘ Nota

Este ejemplo implementa las operaciones de Dataverse y la salida de la consola detalladas en el [Ejemplo de operaciones de metadatos de API web](#) y utiliza las construcciones comunes de C# que se describen en [Ejemplos de API web \(C#\)](#).

Requisitos previos

Los siguientes elementos son necesarios para compilar y ejecutar este ejemplo:

- Microsoft Visual Studio 2022.
- Acceso a Dataverse con privilegios para realizar operaciones de datos.

Cómo ejecutar esta muestra

1. Clonar o descargar el repositorio de [ejemplos de PowerApps](#).
2. Localizar la carpeta [/dataverse/webapi/C#-NETx/MetadataOperations/](#).
3. Abrir el archivo `MetadataOperations.sln` usando Visual Studio 2022
4. Editar el archivo `appsettings.json` para establecer los siguientes valores de propiedad:

Property	Instrucciones
<code>Url</code>	La URL para su entorno. Reemplazar el valor de marcador de posición <code>https://yourorg.api.crm.dynamics.com</code> con el valor para su entorno. Consulte Ver recursos para desarrolladores para encontrar esto.

Property	Instrucciones
UserPrincipalName	Reemplazar el valor de marcador de posición <code>you@yourorg.onmicrosoft.com</code> con el valor de UPN que usa para acceder al entorno.
Password	Reemplazar el valor de marcador de posición <code>yourPassword</code> con la contraseña que utiliza.

5. Guarde el archivo `appsettings.json`

6. Presione F5 para ejecutar el ejemplo.

Código

El código para este ejemplo está aquí: [PowerApps-Samples/dataverse/webapi/C#-NETx/MetadataOperations/Program.cs](#) ↗

Demostraciones

Esta muestra tiene 11 regiones:

Sección 0: Crear un editor y una solución

Operaciones: cree un registro de solución y un registro de editor asociado.

ⓘ Nota

Todos los componentes de la solución creados en este ejemplo se asociarán a la solución para que puedan exportarse. Para operaciones sin un mensaje con nombre, esta asociación se crea usando el encabezado de solicitud `MSCRM.SolutionUniqueName` que establece el nombre único de la solución establecido como valor. Todos los nombres de los componentes de la solución tienen el prefijo de personalización del editor.

Sección 1: Crear, recuperar y actualizar una tabla

Operaciones:

1. Crear una nueva tabla propiedad del usuario `sample_BankAccount` enviando una solicitud `POST` a `/EntityDefinitions`.

2. Recuperar la tabla creada enviando una solicitud `GET` a
`/EntityDefinitions(LogicalName='sample_bankaccount')`.
3. Actualizar la tabla creada enviando una solicitud `PUT` a
`/EntityDefinitions(LogicalName='sample_bankaccount')`.

Sección 2: Crear, recuperar y actualizar columnas

Operaciones:

1. Crear una nueva columna booleana `sample_boolean` para la tabla `sample_BankAccount` enviando una solicitud `POST` a
`/EntityDefinitions(LogicalName='sample_bankaccount')/Attributes`.
2. Recuperar la columna booleana `sample_boolean` enviando una solicitud `GET` a
`/EntityDefinitions(LogicalName='sample_bankaccount')/Attributes(LogicalName='sample_boolean')`.
3. Actualizar la columna booleana `sample_boolean` enviando una solicitud `PUT` a
`/EntityDefinitions(LogicalName='sample_bankaccount')/Attributes(LogicalName='sample_boolean')`.
4. Actualizar las etiquetas de opción para la columna booleana `sample_boolean` usando [UpdateOptionValue Action](#).
5. Crear y recuperar una nueva columna datetime `sample_datetime` para la tabla `sample_BankAccount`.
6. Crear y recuperar una nueva columna decimal `sample_decimal` para la tabla `sample_BankAccount`.
7. Crear y recuperar una nueva columna de enteros `sample_integer` para la tabla `sample_BankAccount`.
8. Crear y recuperar una nueva columna de notas `sample_memo` para la tabla `sample_BankAccount`.
9. Crear y recuperar una nueva columna de dinero `sample_money` para la tabla `sample_BankAccount`.
10. Crear y recuperar una nueva columna de opción `sample_choice` para la tabla `sample_BankAccount`.
11. Agregue una opción nueva a la columna `sample_choice` usando [InsertOptionValue Action](#).
12. Cambie el orden de las opciones de la columna `sample_choice` usando [OrderOption Action](#).
13. Elimine una de las opciones de la columna `sample_choice` usando [DeleteOptionValue Action](#).

14. Cree y recupere una nueva columna de selección múltiple `sample_multiselectchoice` para la tabla `sample_BankAccount`.
15. Cree una nueva opción de Estado para la tabla `sample_BankAccount` usando [InsertStatusValue Action](#).

Sección 3: Crear y usar OptionSet global

Operaciones:

1. Cree una nueva opción global denominada `sample_colors` enviando una solicitud `POST` a `/GlobalOptionSetDefinitions`.
2. Recupere la opción global `sample_colors` enviando una solicitud `GET` a `/GlobalOptionSetDefinitions(<id value>)`.
3. Cree una nueva columna de elección `sample_colors` para la tabla `sample_BankAccount` usando la elección global `sample_colors` enviando una solicitud `POST` a `/EntityDefinitions(LogicalName='sample_bankaccount')/Attributes` y asociándola a la elección global.

Sección 4: Crear relación con el cliente

Operaciones:

1. Cree una nueva columna de cliente `sample_customerid` para la tabla `sample_BankAccount` usando [CreateCustomerRelationships Action](#).
2. Recupere la columna de cliente `sample_customerid` enviando una solicitud `GET` a `/EntityDefinitions(LogicalName='sample_bankaccount')/Attributes(LogicalName='sample_customerid')`.
3. Recupere el Relaciones creado para la columna del cliente enviando solicitudes `GET` a `/RelationshipDefinitions(<id>)/Microsoft.Dynamics.CRM.OneToManyRelationshipMetadata``.

Sección 5: Crear y recuperar una relación de uno a muchos

Operaciones:

1. Verifique que la tabla `sample_BankAccount` es elegible para ser referenciada en una relación 1:N usando [CanBeReferenced Function](#).

2. Verifique que la tabla `contact` es elegible para hacer referencia a otras tablas en una relación 1:N usando [CanBeReferencing Function](#).
3. Identifique qué otras tablas pueden hacer referencia a la tabla `sample_BankAccount` en una relación 1:N usando [GetValidReferencingEntities Function](#).
4. Cree una relación 1:N entre las tablas `sample_BankAccount` y `contact` enviando una solicitud `POST` a `/RelationshipDefinitions`.
5. Recupere la relación 1:N enviando una solicitud `GET` a
`/RelationshipDefinitions(<id>)/Microsoft.Dynamics.CRM.OneToManyRelationshipMetadata`.

Sección 6: Crear y recuperar una relación de muchos a uno

Operaciones:

1. Cree una relación N:1 entre las tablas `sample_BankAccount` y `account` enviando una solicitud `POST` a `/RelationshipDefinitions`.
2. Recupere la relación N:1 enviando una solicitud `GET` a
`/RelationshipDefinitions(<id>)/Microsoft.Dynamics.CRM.OneToManyRelationshipMetadata`.

Sección 7: Crear y recuperar una relación de muchos a muchos

Operaciones:

1. Verifique que las tablas `sample_BankAccount` y `contact` son elegibles para participar en una relación N:N usando [CanManyToMany Function](#).
2. Verifique que las tablas `sample_BankAccount` y `contact` son elegibles para participar en una relación N:N usando [GetValidManyToMany Function](#).
3. Cree una relación N:N entre las tablas `sample_BankAccount` y `contact` enviando una solicitud `POST` a `/RelationshipDefinitions`.
4. Recupere la relación N:N enviando una solicitud `GET` a
`/RelationshipDefinitions(<id>)/Microsoft.Dynamics.CRM.ManyToManyRelationshipMetadata`.

Sección 8: Exportar una solución administrada

Operaciones: Exportar la solución creada en [Sección 0: Crear publicador y solución que contiene los elementos creados en esta muestra](#) usando el [ExportSolution Action](#).

Sección 9: Eliminar registros de muestra

Operaciones: se agregó una referencia a cada registro creado en esta muestra a una lista a medida que se creaba. En este ejemplo, todos los registros se eliminan mediante una operación `$batch`.

Sección 10: Importar y eliminar una solución administrada

Operaciones:

1. Importe la solución exportada en la [Sección 8](#) utilizando [ImportSolution Action](#).
2. Consulte la tabla de soluciones para obtener el `solutionid` de la solución importada.
3. Elimine la solución importada usando el `solutionid`.

Limpiar

Por defecto esta muestra eliminará todos los registros creados en ella. Si desea ver los registros creados después de completar la muestra, cambie la variable `deleteCreatedRecords` a `false` y se le pedirá que decida si desea eliminar los registros.

Consulte también

[Usar la API web de Dataverse](#)

[Biblioteca de clases WebAPIService \(C#\)](#)

[Utilizar la API web con definiciones de tabla](#)

[Ejemplos de la API web](#)

[Ejemplo de operaciones básicas de la API web \(C#\)](#)

[Ejemplo de datos de consulta de API web \(C#\)](#)

[Ejemplo de operaciones condicionales de la API web \(C#\)](#)

[Ejemplo de funciones y acciones de la API web \(C#\)](#)

[Ejemplo de operaciones en paralelo de la API web WebAPIService \(C#\)](#)

[Ejemplo de operaciones paralelas de API web con componentes de flujo de datos TPL \(C#\)](#)

 **Nota**

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta. [↗](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales (**declaración de privacidad** [↗](#)).

Usar Postman con API web

Artículo • 19/04/2023

Hay varias herramientas de terceros que permiten autenticarse para instancias de Microsoft Dataverse y componer y enviar solicitudes API web y ver respuestas. El envío es una de las más populares.

Use Postman para realizar consulta ad hoc o para comprobar el comportamiento de operaciones sin necesidad de un programa. En esta sección se describen cómo obtener información sobre cómo configurar un entorno de envío con conexión a su instancia Dataverse y el envío para realizar operaciones con la API web.

El envío ofrece muchas otras capacidades más allá de las estipulados en este contenido. Más información: [Primeras 5 cosas a intentar si es nuevo en Postman](#)

En esta sección

[Configure un entorno Postman](#)

[Use Postman para realizar operaciones con la API web](#)

ⓘ Nota

¿Puede indicarnos sus preferencias de idioma de documentación? [Realice una breve encuesta.](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad](#)).

Configurar un entorno de Postman

Artículo • 02/11/2023

Para ahorrarle tiempo y comenzar de inmediato, en este artículo se describe cómo configurar y usar un entorno de Postman para que funcione en sus entornos de Dataverse sin que tenga que registrar su propia aplicación de Microsoft Entra ID. Para obtener información sobre el entorno y las variables de Postman, consulte [Documentación de Postman > Variables](#).

Requisitos previos

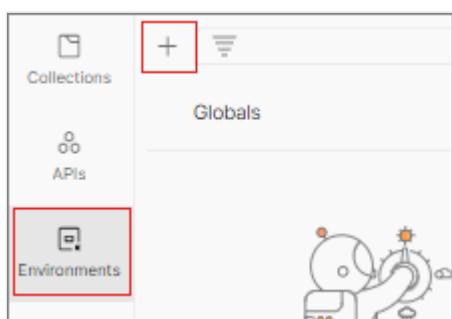
- Tenga un entorno de Dataverse de Power Apps al que pueda conectarse.
- Descargue e instale la [aplicación de escritorio Postman](#).
- Tener una cuenta de cartero [Registrarse como cartero](#).

Conectarse con su entorno de Dataverse

ⓘ Importante

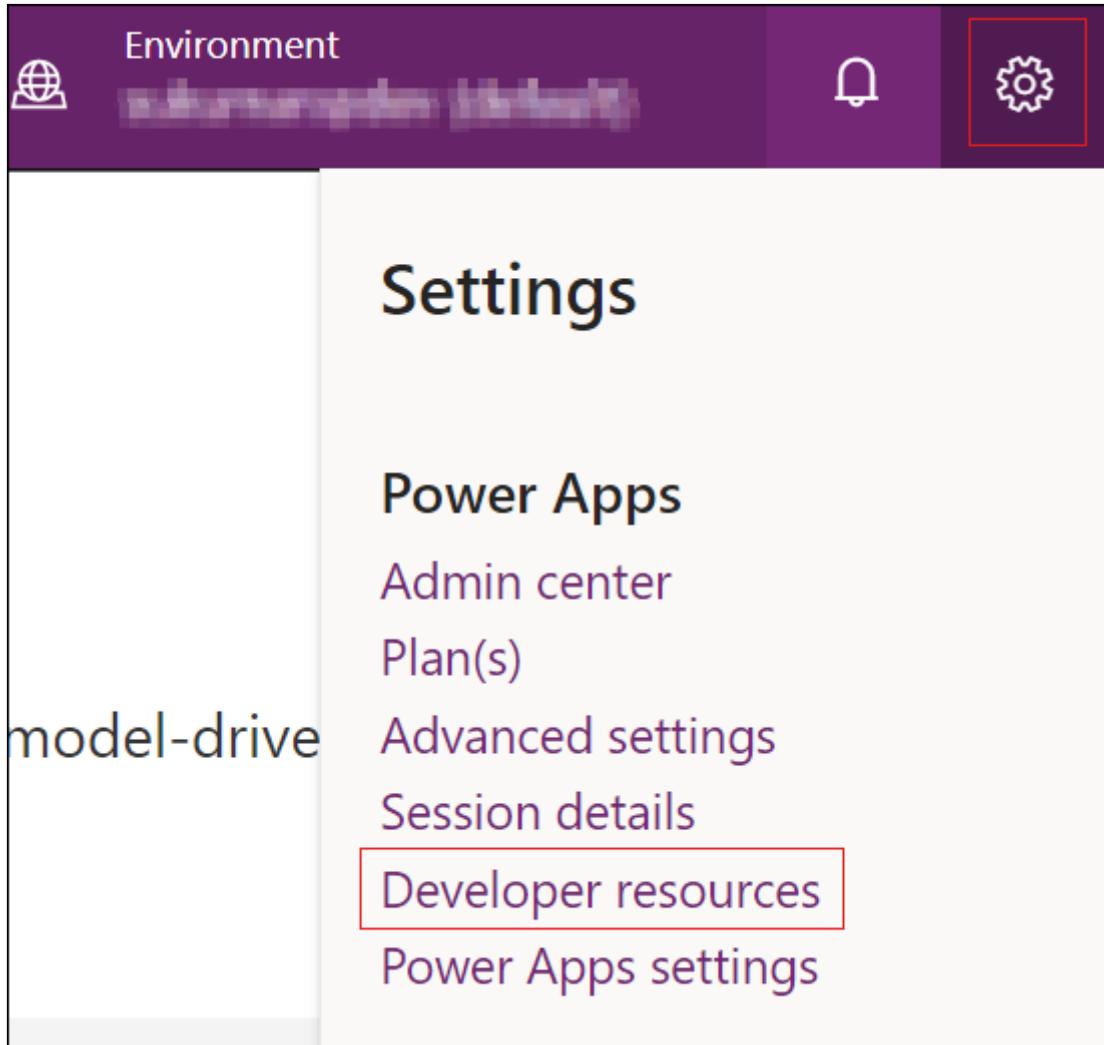
Para ahorrarle tiempo y comenzar de inmediato, proporcionamos una id. de cliente para una aplicación que está registrada para todos los entornos de Dataverse, por lo que no tiene que registrar su propia aplicación de Microsoft Entra ID para conectarse con la API de Dataverse.

1. Inicie la aplicación de escritorio Postman.
2. Para crear un nuevo entorno, seleccione **Entornos** a la izquierda y haga clic en +.



3. Introduzca un nombre para su entorno, por ejemplo, **MiNuevoEntorno**.
4. Inicie sesión en [Power Apps](#) para obtener la URL base del punto de conexión de API web.

5. Seleccione su entorno de Power Apps y haga clic en el botón **Configuración** en la esquina superior derecha. A continuación, haga clic en **Recursos de desarrollador**.



6. En el panel Recursos de desarrollador, recupere la URL base del punto de conexión de API web.

Environment
[REDACTED]
Default

Developer resources

Environment unique name
[REDACTED]

Environment ID
Default-[REDACTED]

Organization ID
[REDACTED]

Web API endpoint

`https://[REDACTED].api.crm8.dynamics.com/api/data/v9.2`

[Learn more](#)

Discovery endpoint

`https://[REDACTED].crm8.dynamics.com/api/discovery/v2.0/Instances`

[Learn more](#)

7. En Postman, agregue los siguientes pares clave-valor en el espacio de edición y use el valor inicial para el valor actual.

VARIABLE	VALOR INICIAL	ACCIÓN
url	<code>https://<your org name>.api.crm.dynamics.com</code>	Use la URL base del punto de conexión de API web
clientid	<code>51f81489-12ee-4a9e-aaaae-a2591f45987d</code>	Copie el valor
version	<code>9.2</code>	Copie el valor
webapiurl	<code>{{url}}/api/data/v{{version}}/</code>	Copie el valor
callback	<code>https://localhost</code>	Copie el valor

VARIABLE	VALOR INICIAL	ACCIÓN
authurl	https://login.microsoftonline.com/common/oauth2/authorize? resource={{url}}	Copie el valor

8. Su configuración debe ser como la siguiente:



VARIABLE		TYPE	INITIAL VALUE	CURRENT VALUE	
<input checked="" type="checkbox"/>	url	default	https://org172f75b9.api.crm8.dynamics.com	https://org172f75b9.api.crm8.dynamics.com	
<input checked="" type="checkbox"/>	clientId	default	51f81489-12ee-4a9e-aaae-a2591f45987d	51f81489-12ee-4a9e-aaae-a2591f45987d	
<input checked="" type="checkbox"/>	version	default	9.2	9.2	
<input checked="" type="checkbox"/>	webapiurl	default	{{url}}/api/data/v{{version}}/	{{url}}/api/data/v{{version}}/	
<input checked="" type="checkbox"/>	callback	default	https://localhost	https://localhost	
<input checked="" type="checkbox"/>	authurl	default	→ https://login.microsoftonline.com/common/oauth2/authorize?resource={{url}}	→ https://login.microsoftonline.com/common/oauth2/authorize?resource={{url}}	

9. Haga clic en **Guardar** para guardar su entorno recién creado llamado **MiNuevoEntorno**.

10. Tras seleccionar su entorno recién creado, configúrelo como el entorno *activo* por uno de estos medios:

- Haga clic en el menú de puntos suspensivos (cerca de la esquina superior derecha) y seleccione **Establecer como entorno activo**.
- Haga clic en el menú desplegable de entorno en la parte superior derecha y seleccione **MyNewEnvironment**.

Generar un token de acceso para usar con su entorno

Para conectar con OAuth 2.0, debe tener un token de acceso. Use los siguientes pasos para permitir el acceso a un nuevo token de acceso:

1. Asegúrese de que esté seleccionado el entorno recién creado **MiNuevoEntorno**. Haga clic en **+** junto a **MiNuevoEntorno**.



2. Aparece el siguiente panel. Seleccione la pestaña **Authorization (Autorización)**.

The screenshot shows the Postman interface with the following details:

- Project: MyNewEnvironment
- Method: GET
- URL: Enter request URL
- Authorization tab is selected (highlighted with a red box).
- Other tabs: Params, Headers (6), Body, Pre-request Script, Tests, Settings.
- Sub-sections under Authorization: Query Params, Key.

3. Establezca el **Tipo** en OAuth 2.0 y Agregar datos de autorización en Encabezados de solicitud.

The screenshot shows the Postman interface with the following details:

- Method: GET
- URL: Enter request URL
- Authorization tab is selected (highlighted with a red box).
- Type dropdown: OAuth 2.0 (highlighted with a red box).
- Configuration Options tab is selected (highlighted with a red box).
- Sub-sections under Configuration Options: Configure New Token, Token Name, Grant Type.
- Other tabs: Params, Headers (6), Body, Pre-request Script, Tests, Settings.
- Sub-sections under Headers (6): Add authorization data to, Request Headers (highlighted with a red box).

4. En el panel **Configurar nuevo token**, establezca los siguientes valores:

Nombre	Valor	Acción
Tipo de concesión	implícito	Elija implícito en la lista desplegable
URL de devolución de llamada	<code>{{callback}}</code>	Copie el valor
URL de autenticación	<code>{{authurl}}</code>	Copie el valor
Id. de cliente	<code>{{clientid}}</code>	Copie el valor

5. Su configuración debe ser como la siguiente:

Configure New Token

Configuration Options ● Advanced Options

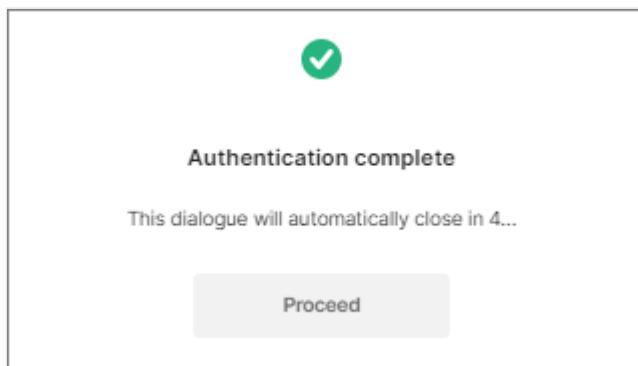
Token Name	PostmanTest01
Grant Type	Implicit
Callback URL ⓘ	<code>{{callback}}</code>
<input type="checkbox"/> Authorize using browser	
Auth URL ⓘ	<code>{{authurl}}</code>
Client ID ⓘ	<code>{{clientid}}</code>
Scope ⓘ	e.g. read:org
State ⓘ	State
Client Authentication	Send as Basic Auth header
Clear cookies ⓘ	
Get New Access Token	

ⓘ Nota

Si está configurando entornos en Postman para múltiples instancias de Dataverse utilizando diferentes credenciales de usuario, haga clic en **Borrar cookies** para eliminar las cookies que Postman almacena en caché.

6. Haga clic en **Obtener un token de acceso nuevo**.

Una vez que haga clic en **Obtener nuevo token de acceso**, aparecerá un cuadro de diálogo de inicio de sesión en Microsoft Entra ID. Especifique su nombre de usuario y contraseña y, a continuación, haga clic en **Iniciar sesión**. Una vez que se completa la autenticación, aparece el siguiente diálogo.



7. Una vez que el diálogo de autenticación se cierre automáticamente en unos segundos, aparece el panel **Administrar tokens de acceso**. Haga clic en **Usar token**.

The screenshot shows a 'Token Details' panel with the following fields:

- Token Name: Token Name (with a pencil icon)
- Access Token: eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1Nlslng1dCl6lmpTMVhvMU9XRGp

An orange 'Use Token' button is located in the top right corner.

8. El token recién generado aparecerá automáticamente en el cuadro de texto debajo del desplegable **Tokens disponibles**.

The screenshot shows a 'Current Token' panel with the following interface:

- A message: "This access token is only available to you. Sync the token to let collaborators on this request use it."
- An 'Access Token' field containing the value: eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1Nlslng1dCl6lmpTMVhvMU9XRGp
- A dropdown menu labeled 'Available Tokens' containing the same token value.
- A yellow 'Copy' icon next to the token value.

Probar la conexión

A continuación se muestra cómo probar su conexión usando [WhoAmI](#):

1. Seleccione `GET` como el método de HTTP y agregan `{{webapiurl}}WhoAmI` en el espacio de edición.

The screenshot shows a tool interface with the following configuration:

- METHOD: GET
- URL: {{webapiurl}}WhoAmI (highlighted with a red box)
- Send button (blue)

2. Haga clic en **Enviar** para enviar esta solicitud.
3. Si su solicitud se realiza correctamente, verá los datos que se devuelven del punto de conexión de `WhoAmI`, como los siguientes:

GET {{webapiurl}}WhoAmI

Params Auth Headers (8) Body Pre-req. Tests Settings Cookies

Body Cookies (3) Headers (21) Test Results

Pretty Raw Preview Visualize JSON ↗

```
1 "@odata.context": "https://[REDACTED].crm.dynamics.com/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.WhoAmIResponse",
2 "BusinessUnitId": "[REDACTED]-2-ec11-a7b6-000d3a8d3920",
3 "UserId": "[REDACTED]-ec11-a7b6-000d3a8d3920",
4 "OrganizationId": "1ed170e-[REDACTED]-425b-b8b0-c7ffe93b130f"
```

Pasos siguientes

Aprenda a usar Postman para realizar operaciones con la API web.

[Documentos de servicio](#)

Consulte también

[Usar Postman para realizar operaciones](#)

[Tutorial: registrar una aplicación de Dataverse con Microsoft Entra ID](#)

ⓘ Nota

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta. [↗](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad ↗](#)).

Use Postman para realizar operaciones con la API web

Artículo • 23/02/2023

Use el envío para crear y enviar las solicitudes web API y ver respuestas. Este tema describe cómo usar Postman para crear las solicitudes web API que realizan operaciones de creación, recuperación, actualización y eliminación (CRUD) y usar funciones y acciones.

ⓘ Importante

Deberá tener un entorno creado mediante los pasos descritos en [Configure un entorno Postman](#)

El entorno creado mediante las instrucciones en [Configure un entorno de Postman](#), crea `{{webapiurl}}` una variable de Postman que proporciona la dirección URL base para atender solicitudes. Anexar a esta variable para definir la dirección URL de las solicitudes.

Los métodos y valores HTTP que use dependen del tipo de operación que desea realizar. En las siguientes secciones se muestran ejemplos de operaciones comunes.

Recuperar múltiples registros

Use la solicitud `GET` para recuperar un conjunto de opciones globales. El siguiente ejemplo devuelve las primeras tres registros de cuenta.

! Nota

La solicitudes web API deben incluir determinados encabezados HTTP. Cada solicitud debe incluir el valor del encabezado `Accept` de `application/json`, aunque no se espere ningún cuerpo de la respuesta. La versión actual de OData es `4.0`, por lo que los encabezados incluyen `OData-Version: 4.0`. Incluya el encabezado `OData-MaxVersion` de forma que no haya ambigüedades sobre la versión cuando haya nuevos lanzamientos de OData. Más información: [Encabezados HTTP](#)

Ejemplo

```
GET {{webapiurl}}accounts?$select=name,accountnumber&$top=3
```

GET {{webapiurl}}accounts?\$select=name,accountnumber&\$top=3

Headers (5)

Key	Value	Description
If-None-Match	null	
OData-Version	4.0	
Content-Type	application/json	
Accept	application/json	
OData-MaxVersion	4.0	

El cuerpo de la respuesta se parecerá a esto:

JSON

```
{
  "@odata.context": "https://yourorg.crm.dynamics.com/api/data/v9.0/$metadata#accounts(name,accountnumber)",
  "value": [
    {
      "@odata.etag": "W/\"2291741\"",
      "name": "Contoso Ltd",
      "accountnumber": null,
      "accountid": "9c706dc8-d2f5-e711-a956-000d3a328903"
    },
    {
      "@odata.etag": "W/\"2291742\"",
      "name": "Fourth Coffee",
      "accountnumber": null,
      "accountid": "a2706dc8-d2f5-e711-a956-000d3a328903"
    },
    {
      "@odata.etag": "W/\"2291743\"",
      "name": "Contoso Ltd",
      "accountnumber": null,
      "accountid": "9c3216b8-3efb-e711-a957-000d3a328903"
    }
  ]
}
```

Para obtener más información, consulte [Consulta de datos mediante la API web](#).

Recupere un registro en particular

Use la solicitud `GET` para recuperar un conjunto de opciones globales. El siguiente ejemplo recupera dos propiedades de una cuenta específica y expande información sobre el contacto principal relacionados para incluir su nombre completo.

```
GET {{webapiurl}}accounts(<accountid>)?  
$select=name,accountnumber&$expand=primarycontactid($select=fullname)
```

El cuerpo de la respuesta se parecerá a esto:

```
{
  "@odata.context": "https://yourorg.crm.dynamics.com/api/data/v9.0/$metadata#accounts(name,accountnumber,primarycontactid(fullname))/$entity",
  "@odata.etag": "W/\"2291742\"",
  "name": "Fourth Coffee",
  "accountnumber": null,
  "accountid": "a2706dc8-d2f5-e711-a956-000d3a328903",
  "primarycontactid": {
    "@odata.etag": "W/\"1697263\"",
    "fullname": "Susie Curtis",
    "contactid": "a3706dc8-d2f5-e711-a956-000d3a328903"
  }
}
```

Más información: [Recuperar una fila de tabla mediante la API web.](#)

Creación de un registro

Use una solicitud `POST` para enviar datos para crear una entidad. Establezca la dirección URL del nombre fijado para entidad, en este caso, `accounts`, y establezca los encabezados como se muestra aquí.

`POST {{webapiurl}}accounts`

Establezca el cuerpo de la solicitud con información sobre la cuenta a crear.

The screenshot shows a Postman request configuration. The URL is {{webapiurl}}accounts. The method is POST. The Headers tab shows 'Content-Type: application/json'. The Body tab is selected and set to 'raw' with type 'JSON (application/json)'. The JSON payload is:

```
1 {  
2   "name": "New Account"  
3 }  
4
```

Cuando envía esta solicitud, el cuerpo estará vacío, pero el identificador de la cuenta creada estará en el valor encabezado `OData-EntityId`.

Más información: [Crear una fila de tabla usando la API web](#).

Actualizar un registro

Utilice el método `PATCH` para actualizar un registro de tabla, como se muestra aquí.

`PATCH {{webapiurl}}accounts(<accountid>)`

The screenshot shows a Postman request configuration. The URL is {{webapiurl}}accounts(aaa19cdd-88df-e311-b8e5-6c3be5a8b200). The method is PATCH. The Headers tab shows 'Content-Type: application/json'. The Body tab is selected and set to 'raw' with type 'JSON (application/json)'. The JSON payload is:

```
1 {  
2   "name": "Fabrikam, Inc.",  
3   "telephone1": "+01-338-333-5401"  
4 }
```

Cuando envía esta solicitud, la respuesta estará vacía, pero el identificador de la cuenta actualizada estará en el valor encabezado `OData-EntityId`.

Más información: [Actualizar y eliminar filas de tablas mediante la API web](#).

Eliminar un registro

Use el método `DELETE` para eliminar un registro existente.

`DELETE {{webapiurl}}accounts(<accountid>)`

The screenshot shows a Postman request configuration. The URL is {{webapiurl}}accounts(aaa19cdd-88df-e311-b8e5-6c3be5a8b200). The method is DELETE. The Headers tab is selected and shows the following headers:

Key	Value	Description
If-None-Match	null	
OData-Version	4.0	
Content-Type	application/json	
Accept	application/json	
OData-MaxVersion	4.0	

Cuando envíe esta solicitud, el registro de cuenta con lo dado `accountid` obtiene eliminar.

Más información: [Actualizar y eliminar filas de tablas mediante la API web](#).

Use una función

Use una solicitud `GET` con las funciones enumeradas en [Referencia de función de la API Web](#) para realizar operaciones que pueden reutilizarse con API web. El siguiente ejemplo muestra cómo enviar una solicitud web API que use [RetrieveDuplicates function](#) para detectar y para recuperar los duplicados de un registro especificado.

Método	Dirección URL
<code>HTTP</code>	
<code>GET</code>	<code>{{webapiurl}}RetrieveDuplicates(BusinessEntity=@p1,MatchingEntityName=@p2,PagingInfo=@p3)?@p1='@odata.type':'Microsoft.Dynamics.CRM.account','accountid': '<accountid>'&@p2='account'&@p3='{'PageNumber':1,'Count':50}</code>

The screenshot shows the Postman interface with a GET request. The URL is `{{webapiurl}}RetrieveDuplicates(BusinessEntity=@p1,MatchingEntityName=@p2,PagingInfo=@p3)?@p1='@odata.type':'Microsoft.Dynamics.CRM.account','accountid': '<accountid>'&@p2='account'&@p3='{'PageNumber':1,'Count':50}`. The Headers tab is selected, showing the following configuration:

Key	Value	Description
If-None-Match	null	
OData-Version	4.0	
Content-Type	application/json	
Accept	application/json	
OData-MaxVersion	4.0	

Las funciones devuelven una colección o un tipo complejo. La respuesta anterior de [RetrieveDuplicates function](#) debería verse así:

The screenshot shows a JSON response with the following structure:

```
{  
  "  
    "@odata.context":  
      "https://yourorgname.crm.dynamics.com/api/data/v9.0/$metadata#accounts",  
    "value": [  
      <Omitted for brevity: JSON data for any matching accounts including all  
      properties>  
    ]  
}
```

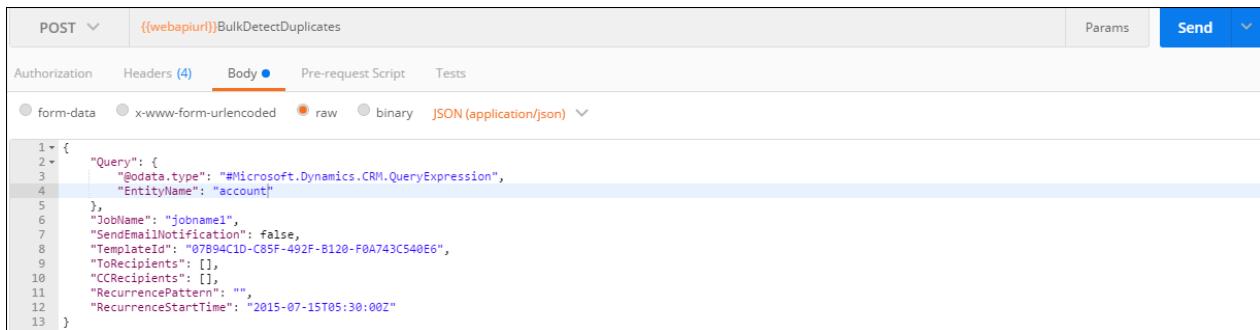
Más infomación: [Usar funciones web API](#).

Usar una acción

Use una solicitud `POST` con las acciones listadas en [Referencia de acción de la API Web](#) para realizar operaciones que tienen efectos secundarios.

Este ejemplo muestra cómo usar [BulkDetectDuplicates action](#).

`POST {{webapiurl}}BulkDetectDuplicates`



```
1 {  
2   "Query": {  
3     "@odata.type": "#Microsoft.Dynamics.CRM.QueryExpression",  
4     "EntityName": "account"  
5   },  
6   "JobName": "jobname1",  
7   "SendEmailNotification": false,  
8   "TemplateId": "07B94C1D-C85F-492F-B120-F0A743C540E6",  
9   "ToRecipients": [],  
10  "CCRecipients": [],  
11  "RecurrencePattern": "",  
12  "RecurrenceStartTime": "2015-07-15T05:30:00Z"  
13 }
```

La solicitud en el ejemplo que acaba de ver envía un trabajo de detección duplicado asíncrono que se ejecuta en segundo plano. Los duplicados se detectan mediante las reglas de duplicados publicadas para el tipo de tabla. [BulkDetectDuplicatesResponse ComplexType](#) se vuelve como una respuesta desde [BulkDetectDuplicates action](#). La respuesta incluye la propiedad `JobId`, que contiene el GUID de trabajo duplicado asíncrono de detección de duplicados que detecta y registra registros duplicados.

Más información: [Usar acciones web API](#).

Vea también

[Usar Postman con API web](#)

[Realizar operaciones mediante la API web](#)

⚠ Nota

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta. ↗ (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad](#) ↗).

JavaScript del lado del cliente mediante API Web en aplicaciones basadas en modelo

Artículo • 25/04/2023

En recursos web HTML, secuencias de comandos de formularios o comandos de cinta de opciones en aplicaciones basadas en modelo puede utilizar JavaScript para realizar operaciones en los datos de Microsoft Dataverse utilizando la API web. Utilice los métodos de la API de cliente [Xrm.WebApi](#) para usar la API web con JavaScript y con los recursos web.

Vea también

[Aplicar lógica empresarial mediante scripting de cliente](#)

[Personalizar comandos y la cinta de opciones](#)

[Recursos web](#)

Nota

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta. [↗](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad ↗](#)).

Microsoft Dataverse Versiones de la API web

Artículo • 09/04/2023

A partir de la v9.0 de Dynamics 365, la API web admite diferencias específicas de versión en el mismo entorno.

Esto es diferente del comportamiento para las versiones de v8.x. En las versiones anteriores nuevas funcionalidades estaban disponibles para cualquier versión del servicio según la actualización aplicada al entorno. Después de la actualización a v8.2, los servicios de v8.0 y v8.1 eran idénticos. Esto fue posible porque todos los cambios eran adición. Nada se ha quitado ni se han introducido cambios importantes. Como consecuencia, la versión específica que se hace referencia en la dirección URL del servicio de la versión 8.x no era realmente importante.

El futuro de las funcionalidades del servicio puede cambiar, incluidos cambios potencialmente importantes como quitar operaciones específicas. Esto le permitirá de las mejoras que se aplicarán a la forma en curso. En este tema se registra cualquier diferencia específica de la versión y las limitaciones en la API web aún no se ha igualado con el servicio de la organización.

ⓘ Nota

Si bien las versiones v9.x pueden admitir diferencias específicas, no se han agregado cambios importantes a las versiones v9.0, v9.1 o v9.2. Cada una de estas versiones tiene comportamientos de API web idénticos.

Las diferencias en el comportamiento de la API se deben más a las soluciones instaladas en el sistema que a la versión del producto. Sin embargo, si necesitamos hacer un cambio fundamental que no sea compatible con versiones anteriores, se incluirá en un nuevo número de versión.

Guia : Use el número de versión que estaba vigente cuando se escribió su código. No utilice automáticamente una versión más reciente sin buscar diferencias documentadas aquí y realizar pruebas. No asuma que una versión más reciente será totalmente compatible con versiones anteriores.

Diferencias específicas de las versiones de la API web

Las diferencias a continuación se refieren a cambios en las versiones v8.2 y v9.0 de la API web.

Codificación para caracteres especiales en respuesta de la consulta FetchXML

Para las versiones v8.x, la respuesta de las consultas FetchXML que contengan entidades de vínculo y sus atributos contiene caracteres Unicode especiales, por ejemplo '.' se convierte en 'x002e' y '@' se convierte en 'x0040'. Esta codificación de caracteres especiales no está presente en la respuesta de las consultas FetchXML para la versión v9.x .

Mismo nombre para tabla y columna

Si el nombre de una tabla (entidad) y uno de las columnas (atributos) es el mismo, se anexa "1 "al nombre de atributo en instancias v8.x. Por ejemplo, si una entidad **new_zipcode** tiene un atributo con el nombre **new_zipcode**, el nombre del atributo cambiará a **new_zipcode1**.

Para instancias de v9.x, no se anexará nada al nombre del atributo.

Nuevas operaciones agregadas

Las operaciones siguientes se han agregado a la API web para la versión v9.x.

Operations	Operaciones (continuación)	Operaciones (continuación)
GrantAccessRequest	ModifyAccessRequest	RetrieveSharedPrincipalsAndAccessRequest

Limitaciones de la API web de

La API web de Dataverse proporciona paridad completa con las capacidades de servicio de la organización. Para Dataverse, este tema describe las limitaciones que proceden de Dataverse versión v8.x. Para las versiones anteriores, consulte [Limitaciones de API Web de Dynamics CRM 2016](#).

ⓘ Nota

Si ha definido una acción personalizada que incluya un valor de devolución complejo y un valor de devolución simple, una acción correspondiente no estaba

disponible en la API web pero estaba disponible mediante el punto de conexión de servicio de la organización. Un valor de devolución complejo es una `EntityReference`, `Entity`, o `EntityCollection`. Puede tener cualquier combinación de valores de devolución simples o un solo valor de devolución complejo. Más información: [Crear sus propias acciones](#).

Consultar también

[Usar la API web de Dataverse](#)

[Autenticarse en Dataverse con la API web](#)

[Tipos y operaciones de API web](#)

[Realizar operaciones mediante la API web](#)

Nota

¿Puede indicarnos sus preferencias de idioma de documentación? [Realice una breve encuesta](#).[↗] (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad](#)[↗]).

Ejemplos de operaciones de datos de API web

Artículo • 23/02/2023

Puede usar la API web de Microsoft Dataverse con una gran variedad de lenguajes o bibliotecas de programación. Esta guía proporciona una serie de ejemplos de código que demuestran cómo utilizar la API web de diferentes formas. Este tema introduce los ejemplos disponibles para cada grupo de operaciones y cómo realizar estas operaciones con otros lenguajes o bibliotecas.

Lista de ejemplos de la API web

En la siguiente tabla se describen los ejemplos de la API web de Dataverse y sus implementaciones específicas del lenguaje.

Descripción en lenguaje neutro	Implementación en C#	Implementación JavaScript del lado del cliente
Ejemplos de API web (este tema)	Ejemplos de la API web (C#)	Ejemplos de la API web (JavaScript del lado del cliente)

Grupos de operaciones

En la siguiente tabla se clasifican los ejemplos por grupos de operación demostrados.

Agrupar	Descripción
Ejemplo de operaciones básicas de la API web	Cómo realizar operaciones CRUD (Crear, Recuperar, Actualizar y Eliminar) y asociativas básicas. Más información: <ul style="list-style-type: none">- Crear una fila de tabla usando la API web- Recuperar una fila de tabla usando la API web- Actualizar y eliminar filas de tablas usando la API web- Asociar y anular la asociación de filas de tabla mediante la API web
Ejemplo de datos de consulta de la API web	Cómo realizar solicitudes de consulta básicas. Más información: <ul style="list-style-type: none">- Consultar datos utilizando la API web- Recuperar y ejecutar consultas predefinidas

Agrupar	Descripción
Ejemplo de operaciones condicionales de la API web	<p>Cómo realizar ciertas categorías de operaciones que se basan condicionalmente en la versión de la fila de la tabla contenida en el servidor y/o mantenido actualmente por el cliente.</p> <p>Más información:</p> <ul style="list-style-type: none"> - Realizar operaciones condicionales mediante la API web
Ejemplo de funciones y acciones de la API web	<p>Cómo usar funciones y acciones enlazadas y sin enlazar, incluidas acciones personalizadas.</p> <p>Más información:</p> <ul style="list-style-type: none"> - Usar funciones de la API web - Usar acciones de la API web

Lenguaje o biblioteca

La siguiente tabla muestra los temas que hacen referencia a los problemas comunes de implementación específicos bibliotecas o de lenguajes.

Lenguaje o biblioteca	Descripción
Ejemplos de la API web (C#)	<p>Describe los elementos comunes que se usan en este grupo de ejemplos en C# que demuestren operaciones con clases .NET básicas y un mínimo de bibliotecas de código auxiliar.</p>
Ejemplos de la API web (JavaScript del lado del cliente)	<p>En construcción.</p>

Consultar también

[Usar la API web de Dataverse](#)

[Ejemplo de operaciones básicas de la API web](#)

[Ejemplo de datos de consulta de la API web](#)

[Ejemplo de operaciones condicionales de la API web](#)

[Ejemplo de funciones y acciones de la API web](#)

[Ejemplos de la API web \(C#\)](#)

ⓘ Nota

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta. ↗ (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales
[\(declaración de privacidad ↗\)](#).

Ejemplo de operaciones básicas de la API web

Artículo • 25/08/2023

Esta colección de fragmentos de código muestra cómo realizar operaciones básicas de CRUD (Crear, Recuperar, Actualizar y Eliminar) y asociativas usando la API web de Microsoft Dataverse.

- [Ejemplo de operaciones básicas de la API web \(C#\)](#)

Este tema describe un conjunto común de operaciones implementadas por cada muestra fragmento de código en este grupo. Este tema describe las solicitudes y respuestas HTTP y la salida de texto que realizará cada muestra sin los detalles específicos del idioma. Consulte las descripciones específicas del idioma y los ejemplos individuales para obtener más información acerca de cómo se realizan estas operaciones.

Demostraciones

Este ejemplo se divide en las siguientes secciones, que contienen operaciones de datos de consulta de la API web de Dataverse que se describen minuciosamente en los temas conceptuales asociados especificados.

Sección de código	Temas conceptuales asociados
Sección 1: Operaciones básicas crear y actualizar	Crear básico Actualización básica
Sección 2: Crear con asociación	Asociar filas de tabla en la creación
Sección 3: Crear filas de tablas relacionadas (inserción profunda)	Crea filas de tablas relacionadas en una sola operación
Sección 4: Asociar y disociar filas de tablas existentes	Asociar y anular la asociación de filas de tabla mediante la API web
Sección 5: Eliminar filas de la tabla (limpieza de muestra)	Eliminación básica

Nota

Para razones de brevedad, se han omitido los encabezados HTTP menos pertinentes. Las URL de los registros variarán con la dirección de la organización base y el id. de la fila asignada por su servidor de Dataverse.

Sección 1: Operaciones básicas crear y actualizar

Esta sección crea un solo contacto y después realiza una serie de actualizaciones sobre esa instancia. Tenga en cuenta que el encabezado de respuesta [OData-EntityId](#) contiene la dirección URL de esta fila recién creada (instancia de entidad), que incluye entre paréntesis el identificador único de este registro.

1. Crear un nuevo contacto, llamado Rafel Shillo.

Solicitud:

```
HTTP

POST [Organization Uri]/api/data/v9.2/contacts HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
    "firstname": "Rafel",
    "lastname": "Shillo"
}
```

Respuesta:

```
HTTP

HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization Uri]/api/data/v9.2/contacts(0928bcb4-
bb27-ed11-9db1-002248274ada)
```

Salida de la consola:

```
Contact URI: [Organization Uri]/api/data/v9.2/contacts(0928bcb4-bb27-
ed11-9db1-002248274ada)
```

Contact relative Uri: contacts(0928bcb4-bb27-ed11-9db1-002248274ada)

Las propiedades disponibles para cada tipo se definen en el documento de metadatos y también se documentan para cada tipo en la sección [Web API Entity Type Reference](#). Para obtener más información general, vea [tipos de API web y operaciones](#).

- Actualice el contacto con valores de ingresos anuales (\$80,000) y el nombre del puesto (Desarrollador junior).

Solicitud:

HTTP

```
PATCH [Organization Uri]/api/data/v9.2/contacts(0928bcb4-bb27-ed11-9db1-002248274ada) HTTP/1.1
If-Match: *
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
    "annualincome": 80000,
    "jobtitle": "Junior Developer"
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization Uri]/api/data/v9.2/contacts(0928bcb4-bb27-ed11-9db1-002248274ada)
```

Salida de la consola:

```
Contact 'Rafel Shillo' updated with jobtitle and annual income
```

- Recupere el contacto con su conjunto de propiedades inicializadas de forma explícita. El `fullname` es una propiedad de sólo lectura que se calcula desde las propiedades `firstname` y `lastname`, que se inicializaron de forma explícita cuando

se creó la instancia. En cambio, la propiedad `description` no se inicializó explícitamente, por lo que mantiene su valor predeterminado, una cadena `null`.

Tenga en cuenta que la respuesta, además de los valores solicitados y los encabezados típicos, también devuelven automáticamente los siguientes tipos de información adicional:

- El ID principal para el tipo de tabla actual, aquí `contactid`.
- Un valor *ETag*, denotado por la clave `@odata.etag`, que identifica la versión específica del recurso solicitado. Para obtener más información, consulte [realizar operaciones condicionales mediante la API web](#).
- El contexto de metadatos, denotado por la clave `@odata.context`, proporciona una forma de comparar resultados de la consulta para determinar si proceden de la misma consulta.
- `_transactioncurrencyid_value` que indica la divisa local de la transacción monetaria.

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/contacts(0928bcb4-bb27-ed11-9db1-002248274ada)?$select=fullname,annualincome,jobtitle,description
HTTP/1.1
Prefer: odata.include-annotations="*"
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
ETag: W/"72935648"
OData-Version: 4.0
Preference-Applied: odata.include-annotations="*"

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#contacts(fullname,annualincome,jobtitle,de
scription)/$entity",
  "@odata.etag": "W/\"72935648\"",
  "fullname": "Rafel Shillo",
  "annualincome@OData.Community.Display.V1.FormattedValue":
  "$80,000.00",
```

```
"annualincome": 80000.0,  
"jobtitle": "Junior Developer",  
"description": null,  
  
"_transactioncurrencyid_value@odata.Community.Display.V1.FormattedValue  
": "US Dollar",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigation  
onproperty": "transactioncurrencyid",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname"  
: "transactioncurrency",  
"_transactioncurrencyid_value": "228f42f8-e646-e111-8eb7-  
78e7d162ced1",  
"contactid": "0928bcb4-bb27-ed11-9db1-002248274ada"  
}
```

Salida de la consola:

```
Contact 'Rafel Shillo' retrieved:  
    Annual income: $80,000.00  
    Job title: Junior Developer  
    Description:
```

ⓘ Importante

Debe usar siempre la selección y filtrado en operaciones de recuperación para optimizar el rendimiento. Para obtener más información, consulte [Consulta de datos mediante la API web](#).

4. Actualice la instancia de contacto proporcionando nuevos valores a estas mismas propiedades.

Solicitud:

HTTP

```
PATCH [Organization Uri]/api/data/v9.2/contacts(0928bcb4-bb27-ed11-  
9db1-002248274ada) HTTP/1.1  
If-Match: *  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json
```

```
{
```

```
"jobtitle": "Senior Developer",
"annualincome": 95000,
"description": "Assignment to-be-determined"
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization Uri]/api/data/v9.2/contacts(0928bcb4-
bb27-ed11-9db1-002248274ada)
```

Salida de la consola:

```
Contact 'Rafel Shillo' updated:
    Job title: Senior Developer
    Annual income: 95000
    Description: Assignment to-be-determined
```

Importante

Envíe solo propiedades cambiadas en solicitudes de actualización. Para obtener más información, consulte [Actualización básica](#).

5. Establezca explícitamente una sola propiedad, el número de teléfono principal.

Tenga en cuenta que esto es una solicitud `PUT` y que la clave JSON llamada `value` se utiliza al realizar operaciones en propiedades individuales.

Solicitud:

HTTP

```
PUT [Organization Uri]/api/data/v9.2/contacts(0928bcb4-bb27-ed11-9db1-
002248274ada)/telephone1 HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
    "value": "555-0105"
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent  
OData-Version: 4.0
```

Salida de la consola:

```
Contact 'Rafel Shillo' phone number updated.
```

6. Recupere esa misma propiedad individual, el número de teléfono principal. Una vez más, note el uso de la clave llamada `value`.

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/contacts(0928bcb4-bb27-ed11-9db1-  
002248274ada)/telephone1 HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK  
OData-Version: 4.0
```

```
{  
  "@odata.context": "[Organization  
Uri]/api/data/v9.2/$metadata#contacts(0928bcb4-bb27-ed11-9db1-  
002248274ada)/telephone1",  
  "value": "555-0105"  
}
```

Salida de la consola:

```
Contact's telephone # is: 555-0105.
```

Sección 2: Crear con asociación

Esta sección crea un nuevo registro de cuenta denominado `Contoso, Ltd.` y la asocia a un contacto existente, `Rafel Shillo`, que se creó en [Sección 1](#). Esta creación y asociación se realizan en una sola operación POST.

1. Cree la cuenta Contoso, Ltd. y establezca su atributo de contacto principal al contacto existente Rafel Shillo. La anotación `@odata.bind` indica que se está creando a una asociación, enlazando aquí la propiedad de navegación de un solo valor `primarycontactid` con un contacto existente, Rafel Shillo.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/accounts HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
  "name": "Contoso Ltd",
  "telephone1": "555-5555",
  "primarycontactid@odata.bind": "contacts(0928bc4-bb27-ed11-9db1-002248274ada)"
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization Uri]/api/data/v9.2/accounts(2728bc4-bb27-ed11-9db1-002248274ada)
```

Salida de la consola:

```
Account 'Contoso Ltd' created.
Account URI: accounts(2728bc4-bb27-ed11-9db1-002248274ada)
```

2. Recupere el contacto principal de la cuenta Contoso, Ltd., de nuevo usando `$expand` con la propiedad de navegación de un solo valor `primarycontactid` para acceder al registro asociado [contact EntityType](#).

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/accounts(2728bcb4-bb27-ed11-9db1-002248274ada)?
$select=name,&$expand=primarycontactid($select=fullname,jobtitle,annualincome) HTTP/1.1
Prefer: odata.include-annotations="*"
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
ETag: W/"72935670"
OData-Version: 4.0
Preference-Applied: odata.include-annotations="*"

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#accounts(name,primarycontactid(fullname,jo
bttitle,annualincome))/$entity",
    "@odata.etag": "W/\"72935670\",
    "name": "Contoso Ltd",
    "accountid": "2728bcb4-bb27-ed11-9db1-002248274ada",
    "primarycontactid": {
        "@odata.etag": "W/\"72935663\",
        "fullname": "Rafel Shillo",
        "jobtitle": "Senior Developer",
        "annualincome@OData.Community.Display.V1.FormattedValue": "$95,000.00",
        "annualincome": 95000.0,
        "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue
": "US Dollar",
        "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigati
onproperty": "transactioncurrencyid",
        "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname"
: "transactioncurrency",
        "_transactioncurrencyid_value": "228f42f8-e646-e111-8eb7-
```

```
    78e7d162ced1",
    "contactid": "0928bcb4-bb27-ed11-9db1-002248274ada"
}
```

Salida de la consola:

```
Account 'Contoso Ltd' has primary contact 'Rafel Shillo':
Job title: Senior Developer
Annual income: $95,000.00
```

Sección 3: Crear filas de tablas relacionadas (inserción profunda)

Esta sección muestra cómo crear una fila de tabla y una fila relacionada, en una sola solicitud POST. Con este método, todas las filas se crean nuevamente; no existen filas con las que asociarse. Este método tiene dos ventajas. Es más eficiente, reemplazando múltiples operaciones más sencillas de creación y asociación con una operación combinada. Además, es atómico, pues o bien toda la operación se realiza correctamente y se crean todos los objetos relacionados, o la operación fracasa y no se crea ninguno.

Esta sección crea una cuenta, su contacto principal y un conjunto de tareas para ese contacto en una solicitud.

1. Cree la cuenta `Fourth Coffee` y su contacto principal `Susie Curtis` y sus tres tareas relacionadas en una operación. Tenga en cuenta el uso de la propiedad de navegación de un solo valor `primarycontactid` y la propiedad de navegación valorada como colección `Contact_Tasks` para definir estas relaciones, respectivamente. Las propiedades de navegación de un solo valor toman un valor de objeto, en tanto que las propiedades navegación valoradas como colección toman un valor de matriz.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/accounts HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

```
{
  "name": "Fourth Coffee",
  "primarycontactid": {
    "firstname": "Susie",
    "lastname": "Curtis",
    "jobtitle": "Coffee Master",
    "annualincome": 48000,
    "Contact_Tasks": [
      {
        "subject": "Sign invoice",
        "description": "Invoice #12321",
        "scheduledstart": "2023-04-19T03:00:00+07:00",
        "scheduledend": "2023-04-19T04:00:00+07:00",
        "scheduleddurationminutes": 60
      },
      {
        "subject": "Setup new display",
        "description": "Theme is - Spring is in the air",
        "scheduledstart": "2023-04-20T03:00:00+07:00",
        "scheduledend": "2023-04-20T04:00:00+07:00",
        "scheduleddurationminutes": 60
      },
      {
        "subject": "Conduct training",
        "description": "Train team on making our new blended coffee",
        "scheduledstart": "2023-04-21T03:00:00+07:00",
        "scheduledend": "2023-04-21T04:00:00+07:00",
        "scheduleddurationminutes": 60
      }
    ]
  }
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization Uri]/api/data/v9.2/accounts(2e28bcb4-
bb27-ed11-9db1-002248274ada)
```

Salida de la consola:

```
Account 'Fourth Coffee' created.
```

2. Recupere selectivamente la cuenta recién creada de Fourth Coffee y su contacto principal. Una extensión se realiza en la propiedad de navegación de un solo valor

```
primarycontactid.
```

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/accounts(2e28bcb4-bb27-ed11-9db1-002248274ada)?  
$select=name&$expand=primarycontactid($select=fullname,jobtitle,annualincome) HTTP/1.1  
Prefer: odata.include-annotations="*"  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK  
ETag: W/"72935710"  
OData-Version: 4.0  
Preference-Applied: odata.include-annotations="*"  
  
{  
    "@odata.context": "[Organization  
Uri]/api/data/v9.2/$metadata#accounts(name,primarycontactid(fullname,jo  
btitle,annualincome))/$entity",  
    "@odata.etag": "W/\"72935710\"",  
    "name": "Fourth Coffee",  
    "accountid": "2e28bcb4-bb27-ed11-9db1-002248274ada",  
    "primarycontactid": {  
        "@odata.etag": "W/\"72935689\"",  
        "fullname": "Susie Curtis",  
        "jobtitle": "Coffee Master",  
        "annualincome@OData.Community.Display.V1.FormattedValue":  
        "$48,000.00",  
        "annualincome": 48000.0,  
  
        "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue  
        ": "US Dollar",  
  
        "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigation  
        onproperty": "transactioncurrencyid",  
  
        "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname"  
        : "transactioncurrency",  
        "_transactioncurrencyid_value": "228f42f8-e646-e111-8eb7-  
        78e7d162ced1",  
        "contactid": "2f28bcb4-bb27-ed11-9db1-002248274ada"
```

```
}
```

Salida de la consola:

```
Account 'Fourth Coffee' has primary contact 'Susie Curtis':  
    Job title: Coffee Master  
    Annual income: $48,000.00
```

3. Recupere selectivamente las tareas asociadas con el contacto principal recuperado en la operación anterior. Una extensión se realiza en la propiedad de navegación valorada como colección `Contact_Tasks`.

Solicitud:

```
HTTP  
  
GET [Organization Uri]/api/data/v9.2/contacts(2f28bcb4-bb27-ed11-9db1-002248274ada)?  
$select=fullname&$expand=Contact_Tasks($select=subject,description,sche  
duledstart,scheduledend) HTTP/1.1  
Prefer: odata.include-annotations="*"  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json
```

Respuesta:

```
HTTP  
  
HTTP/1.1 200 OK  
ETag: W/"72935689"  
OData-Version: 4.0  
Preference-Applied: odata.include-annotations="*"  
  
{  
    "@odata.context": "[Organization  
Uri]/api/data/v9.2/$metadata#contacts(fullname,Contact_Tasks(subject,de  
scription,scheduledstart,scheduledend))/$entity",  
    "@odata.etag": "W/\\"72935689\\\"",  
    "fullname": "Susie Curtis",  
    "contactid": "2f28bcb4-bb27-ed11-9db1-002248274ada",  
    "Contact_Tasks": [  
        {  
            "@odata.etag": "W/\\"72935719\\\"",  
            "subject": "Sign invoice",  
        }  
    ]  
}
```

```

    "description": "Invoice #12321",
    "scheduledstart@OData.Community.Display.V1.FormattedValue":
    "4/18/2023 1:00 PM",
        "scheduledstart": "2023-04-18T20:00:00Z",
        "scheduledend@OData.Community.Display.V1.FormattedValue":
    "4/18/2023 2:00 PM",
        "scheduledend": "2023-04-18T21:00:00Z",
        "activityid": "3028bcb4-bb27-ed11-9db1-002248274ada"
    },
    {
        "@odata.etag": "W/\"72935723\"",
        "subject": "Setup new display",
        "description": "Theme is - Spring is in the air",
        "scheduledstart@OData.Community.Display.V1.FormattedValue":
    "4/19/2023 1:00 PM",
        "scheduledstart": "2023-04-19T20:00:00Z",
        "scheduledend@OData.Community.Display.V1.FormattedValue":
    "4/19/2023 2:00 PM",
        "scheduledend": "2023-04-19T21:00:00Z",
        "activityid": "3128bcb4-bb27-ed11-9db1-002248274ada"
    },
    {
        "@odata.etag": "W/\"72935727\"",
        "subject": "Conduct training",
        "description": "Train team on making our new blended coffee",
        "scheduledstart@OData.Community.Display.V1.FormattedValue":
    "4/20/2023 1:00 PM",
        "scheduledstart": "2023-04-20T20:00:00Z",
        "scheduledend@OData.Community.Display.V1.FormattedValue":
    "4/20/2023 2:00 PM",
        "scheduledend": "2023-04-20T21:00:00Z",
        "activityid": "3228bcb4-bb27-ed11-9db1-002248274ada"
    }
]
}

```

Salida de la consola:

```

Contact 'Susie Curtis' has the following assigned tasks:
Subject: Sign invoice,
    Description: Invoice #12321
    Start: 4/18/2023 1:00 PM
    End: 4/18/2023 2:00 PM

Subject: Setup new display,
    Description: Theme is - Spring is in the air
    Start: 4/19/2023 1:00 PM
    End: 4/19/2023 2:00 PM

Subject: Conduct training,
    Description: Train team on making our new blended coffee

```

Start: 4/20/2023 1:00 PM
End: 4/20/2023 2:00 PM

Sección 4: Asocie y desasocie entidades existentes

Esta sección demuestra cómo asociar y disociar filas de tablas existentes. La formación de una asociación requiere el uso de un URI de referencia y un objeto de relación, que a continuación se envían en una solicitud POST. Para desasociar se requiere el envío de una solicitud de ELIMINACIÓN al URI de referencia para esa asociación. Primero se forma una asociación de uno a varios entre un contacto y una cuenta. A continuación se forma una asociación de varios a varios entre un competidor y una o varias oportunidades.

1. Agregue a Rafel Shillo como contacto a la cuenta de Fourth Coffee utilizando la propiedad de navegación valorada como colección `contact_customer_accounts`. Tenga en cuenta el uso de la clave especial `@odata.id` para especificar el registro asociado.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/accounts(2e28bcb4-bb27-ed11-9db1-002248274ada)/contact_customer_accounts/$ref HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
    "@odata.id": "[Organization Uri]/api/data/v9.2/contacts(0928bcb4-bb27-ed11-9db1-002248274ada)"
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
```

2. Confirme la operación anterior recuperando la colección de contactos para la cuenta Fourth Coffee. La respuesta contiene la matriz con un solo elemento, el

contacto recientemente asignado Rafel Shillo.

Solicitud:

```
HTTP

GET [Organization Uri]/api/data/v9.2/accounts(2e28bcb4-bb27-ed11-9db1-002248274ada)/contact_customer_accounts?$select=fullname,jobtitle
HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

```
HTTP

HTTP/1.1 200 OK
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#contacts(fullname,jobtitle)",
  "value": [
    {
      "@odata.etag": "W/\\"72935741\\\"",
      "fullname": "Rafel Shillo",
      "jobtitle": "Senior Developer",
      "contactid": "0928bcb4-bb27-ed11-9db1-002248274ada"
    }
  ]
}
```

Salida de la consola:

```
Contact list for account 'Fourth Coffee':
Name: Rafel Shillo, Job title: Senior Developer
```

3. Quite la asociación que se creó recientemente entre la cuenta Fourth Coffee y el contacto Rafel Shillo.

Solicitud:

```
HTTP
```

```
DELETE [Organization Uri]/api/data/v9.2/accounts(2e28bc4-bb27-ed11-9db1-002248274ada)/contact_customer_accounts(0928bc4-bb27-ed11-9db1-002248274ada)/$ref HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
```

4. Cree un rol de seguridad llamado Example Security Role.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/roles HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
    "businessunitid@odata.bind": "businessunits(38e0dbe4-131b-e111-ba7e-78e7d1620f5e)",
    "name": "Example Security Role"
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization Uri]/api/data/v9.2/roles(e359feba-bb27-ed11-9db1-002248274ada)
```

5. Asocie el nuevo rol de seguridad a su registro de usuario del sistema.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/systemusers(4026be43-6b69-e111-  
8f65-78e7d1620f5e)/systemuserroles_association/$ref HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json  
  
{  
    "@odata.id": "[Organization Uri]/api/data/v9.2/roles(e359feba-bb27-  
ed11-9db1-002248274ada)"  
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent  
OData-Version: 4.0
```

Salida de la consola:

```
Security Role 'Example Security Role' associated with to your user  
account.
```

6. Recupere el Ejemplo rol de seguridad usando la relación de muchos a muchos de `systemuserroles_association`:

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/systemusers(4026be43-6b69-e111-  
8f65-78e7d1620f5e)/systemuserroles_association?  
$select=name&$filter=roleid%20eq%20e359feba-bb27-ed11-9db1-002248274ada  
HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#roles(name)",
  "value": [
    {
      "@odata.etag": "W/\"72935763\"",
      "name": "Example Security Role",
      "roleid": "e359feba-bb27-ed11-9db1-002248274ada"
    }
  ]
}
```

Salida de la consola:

```
Retrieved role: Example Security Role
```

7. Disocie el rol de seguridad de su registro de usuario. Observe de nuevo, que tiene la misma sintaxis general usada para quitar una asociación de uno a varios.

Solicitud:

```
HTTP

DELETE [Organization Uri]/api/data/v9.2/systemusers(4026be43-6b69-e111-
8f65-78e7d1620f5e)/systemuserroles_association(e359feba-bb27-ed11-9db1-
002248274ada)/$ref HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

```
HTTP

HTTP/1.1 204 NoContent
OData-Version: 4.0
```

Sección 5: Eliminar filas de la tabla

1. Se elimina cada elemento de la colección de URL de fila. El primero es un registro de contacto para Rafel Shillo.

Solicitud:

```
HTTP  
  
DELETE [Organization Uri]/api/data/v9.2/contacts(0928bcb4-bb27-ed11-  
9db1-002248274ada) HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json
```

Respuesta:

```
HTTP  
  
HTTP/1.1 204 NoContent  
OData-Version: 4.0
```

2. Las iteraciones siguientes a través de la colección eliminan los registros restantes.

Solicitud:

```
HTTP  
  
DELETE [Organization Uri]/api/data/v9.2/accounts(2728bcb4-bb27-ed11-  
9db1-002248274ada) HTTP/1.1  
. . .  
  
DELETE [Organization Uri]/api/data/v9.2/accounts(2e28bcb4-bb27-ed11-  
9db1-002248274ada) HTTP/1.1  
. . .  
  
DELETE [Organization Uri]/api/data/v9.2/contacts(2f28bcb4-bb27-ed11-  
9db1-002248274ada) HTTP/1.1  
. . .  
  
DELETE [Organization Uri]/api/data/v9.2/roles(e359feba-bb27-ed11-9db1-  
002248274ada) HTTP/1.1
```

Consulte también

[Usar la API web de Dataverse](#)

[Crear una fila de tabla usando la API web](#)

[Recuperar una fila de tabla usando la API web](#)

[Actualizar y eliminar filas de tablas usando la API web](#)

[Asociar y anular la asociación de filas de tabla mediante la API web](#)

[Ejemplo de operaciones básicas de la API web \(C#\)](#)

 **Nota**

¿Puede indicarnos sus preferencias de idioma de documentación? [Realice una breve encuesta.](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad](#)).

Ejemplo de datos de consulta de la API web

Artículo • 17/08/2023

Use este grupo de código de ejemplo para aprender a consultar datos utilizando la API web de Microsoft Dataverse. Este ejemplo se implementa como proyecto independiente para los siguientes idiomas:

- [Ejemplo de datos de consulta \(C#\)](#)
- [Ejemplo de datos de consulta de la API web \(JavaScript del lado del cliente\)](#)

En este tema se describe un conjunto común de operaciones implementadas por cada ejemplo en este grupo. En este tema se describen las solicitudes y las respuestas HTTP y el texto generado que cada ejemplo en este grupo realizará sin detalles específicos de idioma. Consulte las descripciones específicas del idioma y los ejemplos individuales para obtener más información acerca de cómo se realizan estas operaciones.

Demostraciones

Este ejemplo se divide en las siguientes secciones principales, que contienen operaciones de datos de consulta de la API web que se describen minuciosamente en los temas conceptuales asociados.

Sección del tema	Temas asociados
Sección 1: Seleccionar propiedades específicas	Recuperar propiedades específicas Valores con formato
Sección 2: Usar funciones de consulta	Filtrar filas Usar funciones de consulta de OData Crear una consulta con funciones Web API Query Function Reference
Sección 3: Pedidos y alias	Ordenar filas Filtrar filas
Sección 4: Limitar y contar los resultados	Usar la opción de consulta \$top Recuento del número de filas
Sección 5: Paginación	Resultados de página
Sección 6: Expandir resultados	Unir tablas
Sección 7: Agregación de resultados	Aregar datos

Sección del tema	Temas asociados
Sección 8: Consultas FetchXML	Esquema FetchXML Usar FetchXML con API web
Sección 9: Usar consultas predefinidas	Recuperar y ejecutar consultas predefinidas userquery EntityType savedquery EntityType
Sección 10: Eliminar registros de muestra	Eliminación básica Ejecute las operaciones por lotes mediante API web

Las siguientes secciones contienen una breve explicación de las operaciones de la API web de Dataverse realizadas, junto con mensajes HTTP correspondientes y la salida asociada de la consola.

Sección 0: Crear registros para consultar

El código de muestra crea un conjunto de filas de muestra para consultar. El código elimina los datos al final a menos que elija conservarlos. Puede obtener resultados diferentes en función de los datos existentes en el entorno.

Los datos se agregan mediante *inserción profunda* en una sola solicitud de `POST` y coinciden con la estructura siguiente:

JSON

```
{
  "name": "Contoso, Ltd. (sample)",
  "Account_Tasks": [
    {
      "subject": "Task 1 for Contoso, Ltd.",
      "description": "Task 1 for Contoso, Ltd. description",
      "actualdurationminutes": 10
    },
    {
      "subject": "Task 2 for Contoso, Ltd.",
      "description": "Task 2 for Contoso, Ltd. description",
      "actualdurationminutes": 10
    },
    {
      "subject": "Task 3 for Contoso, Ltd.",
      "description": "Task 3 for Contoso, Ltd. description",
      "actualdurationminutes": 10
    }
  ],
  "primarycontactid": {
    "firstname": "Yvonne",
    "lastname": "Harrington"
  }
}
```

```
"lastname": "McKay (sample)",  
"jobtitle": "Coffee Master",  
"annualincome": 45000,  
"Contact_Tasks": [  
  {  
    "subject": "Task 1 for Yvonne McKay",  
    "description": "Task 1 for Yvonne McKay description",  
    "actualdurationminutes": 5  
  },  
  {  
    "subject": "Task 2 for Yvonne McKay",  
    "description": "Task 2 for Yvonne McKay description",  
    "actualdurationminutes": 5  
  },  
  {  
    "subject": "Task 3 for Yvonne McKay",  
    "description": "Task 3 for Yvonne McKay description",  
    "actualdurationminutes": 5  
  }  
]  
,  
"contact_customer_accounts": [  
  {  
    "firstname": "Susanna",  
    "lastname": "Stubberod (sample)",  
    "jobtitle": "Senior Purchaser",  
    "annualincome": 52000,  
    "Contact_Tasks": [  
      {  
        "subject": "Task 1 for Susanna Stubberod",  
        "description": "Task 1 for Susanna Stubberod description",  
        "actualdurationminutes": 3  
      },  
      {  
        "subject": "Task 2 for Susanna Stubberod",  
        "description": "Task 2 for Susanna Stubberod description",  
        "actualdurationminutes": 3  
      },  
      {  
        "subject": "Task 3 for Susanna Stubberod",  
        "description": "Task 3 for Susanna Stubberod description",  
        "actualdurationminutes": 3  
      }  
    ]  
,  
  {  
    "firstname": "Nancy",  
    "lastname": "Anderson (sample)",  
    "jobtitle": "Activities Manager",  
    "annualincome": 55500,  
    "Contact_Tasks": [  
      {  
        "subject": "Task 1 for Nancy Anderson",  
        "description": "Task 1 for Nancy Anderson description",  
        "actualdurationminutes": 4  
      }  
    ]  
  }  
]
```

```
        },
        {
            "subject": "Task 2 for Nancy Anderson",
            "description": "Task 2 for Nancy Anderson description",
            "actualdurationminutes": 4
        },
        {
            "subject": "Task 3 for Nancy Anderson",
            "description": "Task 3 for Nancy Anderson description",
            "actualdurationminutes": 4
        }
    ]
},
{
    "firstname": "Maria",
    "lastname": "Cambell (sample)",
    "jobtitle": "Accounts Manager",
    "annualincome": 31000,
    "Contact_Tasks": [
        {
            "subject": "Task 1 for Maria Cambell",
            "description": "Task 1 for Maria Cambell description",
            "actualdurationminutes": 5
        },
        {
            "subject": "Task 2 for Maria Cambell",
            "description": "Task 2 for Maria Cambell description",
            "actualdurationminutes": 5
        },
        {
            "subject": "Task 3 for Maria Cambell",
            "description": "Task 3 for Maria Cambell description",
            "actualdurationminutes": 5
        }
    ]
},
{
    "firstname": "Scott",
    "lastname": "Konersmann (sample)",
    "jobtitle": "Accounts Manager",
    "annualincome": 38000,
    "Contact_Tasks": [
        {
            "subject": "Task 1 for Scott Konersmann",
            "description": "Task 1 for Scott Konersmann description",
            "actualdurationminutes": 6
        },
        {
            "subject": "Task 2 for Scott Konersmann",
            "description": "Task 2 for Scott Konersmann description",
            "actualdurationminutes": 6
        },
        {
            "subject": "Task 3 for Scott Konersmann",
            "description": "Task 3 for Scott Konersmann description",
            "actualdurationminutes": 6
        }
    ]
}
```

```
        "actualdurationminutes": 6
    }
]
},
{
    "firstname": "Robert",
    "lastname": "Lyon (sample)",
    "jobtitle": "Senior Technician",
    "annualincome": 78000,
    "Contact_Tasks": [
        {
            "subject": "Task 1 for Robert Lyon",
            "description": "Task 1 for Robert Lyon description",
            "actualdurationminutes": 7
        },
        {
            "subject": "Task 2 for Robert Lyon",
            "description": "Task 2 for Robert Lyon description",
            "actualdurationminutes": 7
        },
        {
            "subject": "Task 3 for Robert Lyon",
            "description": "Task 3 for Robert Lyon description",
            "actualdurationminutes": 7
        }
    ]
},
{
    "firstname": "Paul",
    "lastname": "Cannon (sample)",
    "jobtitle": "Ski Instructor",
    "annualincome": 68500,
    "Contact_Tasks": [
        {
            "subject": "Task 1 for Paul Cannon",
            "description": "Task 1 for Paul Cannon description",
            "actualdurationminutes": 8
        },
        {
            "subject": "Task 2 for Paul Cannon",
            "description": "Task 2 for Paul Cannon description",
            "actualdurationminutes": 8
        },
        {
            "subject": "Task 3 for Paul Cannon",
            "description": "Task 3 for Paul Cannon description",
            "actualdurationminutes": 8
        }
    ]
},
{
    "firstname": "Rene",
    "lastname": "Valdes (sample)",
    "jobtitle": "Data Analyst III",
    "annualincome": 86000,
```

```

"Contact_Tasks": [
  {
    "subject": "Task 1 for Rene Valdes",
    "description": "Task 1 for Rene Valdes description",
    "actualdurationminutes": 9
  },
  {
    "subject": "Task 2 for Rene Valdes",
    "description": "Task 2 for Rene Valdes description",
    "actualdurationminutes": 9
  },
  {
    "subject": "Task 3 for Rene Valdes",
    "description": "Task 3 for Rene Valdes description",
    "actualdurationminutes": 9
  }
]
},
{
  "firstname": "Jim",
  "lastname": "Glynn (sample)",
  "jobtitle": "Senior International Sales Manager",
  "annualincome": 81400,
  "Contact_Tasks": [
    {
      "subject": "Task 1 for Jim Glynn",
      "description": "Task 1 for Jim Glynn description",
      "actualdurationminutes": 10
    },
    {
      "subject": "Task 2 for Jim Glynn",
      "description": "Task 2 for Jim Glynn description",
      "actualdurationminutes": 10
    },
    {
      "subject": "Task 3 for Jim Glynn",
      "description": "Task 3 for Jim Glynn description",
      "actualdurationminutes": 10
    }
  ]
}
]
}

```

Sección 1: Seleccionar propiedades específicas

Incluya siempre la opción de consulta `$select` en sus consultas. Si no lo hace, el servidor devuelve todas las propiedades de cada fila de la tabla, lo que reduce el rendimiento. [Aprender a seleccionar columnas](#).

Este ejemplo demuestra cómo generar una consulta básica seleccionando tres propiedades de un [contact EntityType](#). Las propiedades son `fullname`, `jobtitle`, `annualincome`. La sección también muestra las diferencias entre los valores con `formato` y sin formato que se ve en los resultados de la propiedad `annualincome` de un contacto.

En este ejemplo, solicitamos un contacto específico. En este caso, es el contacto principal de la cuenta, `Yvonne McKay (sample)`.

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/contacts(81716234-9628-ed11-9db1-000d3a320482)?$select=fullname,jobtitle,annualincome HTTP/1.1
Prefer: odata.include-annotations="*"
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
ETag: W/"1146626"
OData-Version: 4.0
Preference-Applied: odata.include-annotations="*"

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#contacts(fullname,jobtitle,annualincome)/$entity",
    "@odata.etag": "W/\"1146626\"",
    "fullname": "Yvonne McKay (sample)",
    "jobtitle": "Coffee Master",
    "annualincome@OData.Community.Display.V1.FormattedValue": "$45,000.00",
    "annualincome": 45000.0,
    "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
    "US Dollar",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",
    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
    "transactioncurrency",
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
    "contactid": "81716234-9628-ed11-9db1-000d3a320482"
}
```

Salida de la consola:

```
text

Contact basic info:
  Fullname: Yvonne McKay (sample)
  Jobtitle: Coffee Master
  Annualincome (unformatted): 45000
  Annualincome (formatted): $45,000.00
```

Sección 2: Usar funciones de consulta

Use opciones de filtro para establecer criterios para los resultados deseados. Puede crear filtros sencillos y complejos con una combinación de funciones de consulta, operadores de comparación, y operadores lógicos. Más información: [Filtrar filas](#)

Las funciones de consulta son funciones que puede usar como criterios de filtro en una consulta. Estas funciones aceptan parámetros y devuelven un valor `Boolean`. Existen funciones de consulta estándar y funciones de Dataverse de consulta específicas. En este ejemplo se muestra cómo crear una consulta para cada tipo.

Funciones estándar de consulta

Dataverse admite un subconjunto de funciones de consulta integradas de OData, específicamente: `contains`, `endswith` y `startswith`. Por ejemplo, la función de consulta estándar `contains` le permite filtrar las propiedades que coincidan con una cadena. Más información: [Usar las funciones de consulta de OData](#)

En esta operación, consultamos todos los contactos con `fullname` que contienen la cadena `(sample)`.

Solicitud:

```
HTTP

GET [Organization Uri]/api/data/v9.2/contacts?
$select=fullname,jobtitle,annualincome&$filter=contains(fullname,'(sample)')
and _parentcustomerid_value eq 7d716234-9628-ed11-9db1-000d3a320482 HTTP/1.1
Prefer: odata.include-annotations="*"
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0
Preference-Applied: odata.include-annotations="*"

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#contacts(fullname,jobtitle,annualincome)",
    "@Microsoft.Dynamics.CRM.totalrecordcount": -1,
    "@Microsoft.Dynamics.CRM.totalrecordcountlimitexceeded": false,
    "@Microsoft.Dynamics.CRM.globalmetadataversion": "1144249",
    "value": [
        {
            "@odata.etag": "W/\"1146759\",
            "fullname": "Susanna Stubberod (sample)",
            "jobtitle": "Senior Purchaser",
            "annualincome@OData.Community.Display.V1.FormattedValue": "$52,000.00",
            "annualincome": 52000.0,
            "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue": "US Dollar",
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
            "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
            "contactid": "85716234-9628-ed11-9db1-000d3a320482"
        },
        {
            "@odata.etag": "W/\"1146761\",
            "fullname": "Nancy Anderson (sample)",
            "jobtitle": "Activities Manager",
            "annualincome@OData.Community.Display.V1.FormattedValue": "$55,500.00",
            "annualincome": 55500.0,
            "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue": "US Dollar",
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
            "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
            "contactid": "89716234-9628-ed11-9db1-000d3a320482"
        }
    ]
}
```

```
},
{
    "@odata.etag": "W/\"1146763\",
    "fullname": "Maria Cambell (sample)",
    "jobtitle": "Accounts Manager",
    "annualincome@OData.Community.Display.V1.FormattedValue":
"$31,000.00",
    "annualincome": 31000.0,

    "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
"US Dollar",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro
perty": "transactioncurrencyid",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
"transactioncurrency",
        "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-
00224802b2ac",
        "contactid": "8d716234-9628-ed11-9db1-000d3a320482"
},
{
    "@odata.etag": "W/\"1146765\",
    "fullname": "Scott Konersmann (sample)",
    "jobtitle": "Accounts Manager",
    "annualincome@OData.Community.Display.V1.FormattedValue":
"$38,000.00",
    "annualincome": 38000.0,

    "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
"US Dollar",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro
perty": "transactioncurrencyid",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
"transactioncurrency",
        "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-
00224802b2ac",
        "contactid": "91716234-9628-ed11-9db1-000d3a320482"
},
{
    "@odata.etag": "W/\"1146767\",
    "fullname": "Robert Lyon (sample)",
    "jobtitle": "Senior Technician",
    "annualincome@OData.Community.Display.V1.FormattedValue":
"$78,000.00",
    "annualincome": 78000.0,

    "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
"US Dollar",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro
perty": "transactioncurrencyid",
```

```
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
"transactioncurrency",  
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
        "contactid": "95716234-9628-ed11-9db1-000d3a320482"  
,  
{  
    "@odata.etag": "W/\"1146769\"",  
    "fullname": "Paul Cannon (sample)",  
    "jobtitle": "Ski Instructor",  
    "annualincome@OData.Community.Display.V1.FormattedValue":  
"$68,500.00",  
    "annualincome": 68500.0,  
  
"_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
"US Dollar",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
"transactioncurrency",  
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
        "contactid": "99716234-9628-ed11-9db1-000d3a320482"  
,  
{  
    "@odata.etag": "W/\"1146771\"",  
    "fullname": "Rene Valdes (sample)",  
    "jobtitle": "Data Analyst III",  
    "annualincome@OData.Community.Display.V1.FormattedValue":  
"$86,000.00",  
    "annualincome": 86000.0,  
  
"_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
"US Dollar",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
"transactioncurrency",  
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
        "contactid": "9d716234-9628-ed11-9db1-000d3a320482"  
,  
{  
    "@odata.etag": "W/\"1146773\"",  
    "fullname": "Jim Glynn (sample)",  
    "jobtitle": "Senior International Sales Manager",  
    "annualincome@OData.Community.Display.V1.FormattedValue":  
"$81,400.00",  
    "annualincome": 81400.0,  
  
"_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
```

```

    "US Dollar",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",
        "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
            "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
                "contactid": "a1716234-9628-ed11-9db1-000d3a320482"
            }
        ]
    }
}

```

Salida de la consola:

text

```

Contacts filtered by fullname containing '(sample)':
|Full Name           |Job Title
|Annual Income
|-----|-----|-----
-----|-----|-----
|Susanna Stubberod (sample) |Senior Purchaser
|$52,000.00
|Nancy Anderson (sample)   |Activities Manager
|$55,500.00
|Maria Cambell (sample)    |Accounts Manager
|$31,000.00
|Scott Konersmann (sample)|Accounts Manager
|$38,000.00
|Robert Lyon (sample)     |Senior Technician
|$78,000.00
|Paul Cannon (sample)     |Ski Instructor
|$68,500.00
|Rene Valdes (sample)     |Data Analyst III
|$86,000.00
|Jim Glynn (sample)       |Senior International Sales Manager
|$81,400.00

```

Funciones de consulta de Dataverse

Las funciones de consulta de Dataverse proporcionan varias opciones para generar consultas que son relevantes para Dataverse. Para obtener una lista completa de estas funciones, consulte [Web API Query Function Reference](#). Más información: [Crear una consulta con funciones](#)

Las funciones de consulta de YDataverse son similares a las funciones de consulta estándar. La diferencia principal es que cuando usa funciones de consulta de Dataverse,

debe proporcionar el nombre completo de la función incluidos los nombres de parámetros. Por ejemplo, para obtener una lista de contactos creados en la última hora, puede generar una consulta mediante la función [LastXHours](#).

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/contacts?
$select=fullname,jobtitle,annualincome&$filter=Microsoft.Dynamics.CRM.LastXH
ours(PropertyName=@p1,PropertyValue=@p2) and _parentcustomerid_value eq
7d716234-9628-ed11-9db1-000d3a320482&@p1='createdon'&@p2='1' HTTP/1.1
Prefer: odata.include-annotations="*"
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0
Preference-Applied: odata.include-annotations="*"

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#contacts(fullname,jobtitle,annualincome)",
    "@Microsoft.Dynamics.CRM.totalrecordcount": -1,
    "@Microsoft.Dynamics.CRM.totalrecordcountlimitexceeded": false,
    "@Microsoft.Dynamics.CRM.globalmetadataversion": "1144249",
    "value": [
        {
            "@odata.etag": "W/\"1146759\"",
            "fullname": "Susanna Stubberod (sample)",
            "jobtitle": "Senior Purchaser",
            "annualincome@OData.Community.Display.V1.FormattedValue":
"$52,000.00",
            "annualincome": 52000.0,

            "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
"US Dollar",

            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro
perty": "transactioncurrencyid",

            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
"transactioncurrency",
            "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-
00224802b2ac",
            "contactid": "85716234-9628-ed11-9db1-000d3a320482"
```

```
},
{
    "@odata.etag": "W/\"1146761\",
    "fullname": "Nancy Anderson (sample)",
    "jobtitle": "Activities Manager",
    "annualincome@OData.Community.Display.V1.FormattedValue":
"$55,500.00",
    "annualincome": 55500.0,

    "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
"US Dollar",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro
perty": "transactioncurrencyid",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
"transactioncurrency",
        "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-
00224802b2ac",
        "contactid": "89716234-9628-ed11-9db1-000d3a320482"
},
{
    "@odata.etag": "W/\"1146763\",
    "fullname": "Maria Cambell (sample)",
    "jobtitle": "Accounts Manager",
    "annualincome@OData.Community.Display.V1.FormattedValue":
"$31,000.00",
    "annualincome": 31000.0,

    "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
"US Dollar",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro
perty": "transactioncurrencyid",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
"transactioncurrency",
        "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-
00224802b2ac",
        "contactid": "8d716234-9628-ed11-9db1-000d3a320482"
},
{
    "@odata.etag": "W/\"1146765\",
    "fullname": "Scott Konersmann (sample)",
    "jobtitle": "Accounts Manager",
    "annualincome@OData.Community.Display.V1.FormattedValue":
"$38,000.00",
    "annualincome": 38000.0,

    "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
"US Dollar",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro
perty": "transactioncurrencyid",
```

```
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
"transactioncurrency",  
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
        "contactid": "91716234-9628-ed11-9db1-000d3a320482"  
,  
{  
    "@odata.etag": "W/\"1146767\"",  
    "fullname": "Robert Lyon (sample)",  
    "jobtitle": "Senior Technician",  
    "annualincome@OData.Community.Display.V1.FormattedValue":  
"$78,000.00",  
    "annualincome": 78000.0,  
  
"_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
"US Dollar",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
"transactioncurrency",  
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
        "contactid": "95716234-9628-ed11-9db1-000d3a320482"  
,  
{  
    "@odata.etag": "W/\"1146769\"",  
    "fullname": "Paul Cannon (sample)",  
    "jobtitle": "Ski Instructor",  
    "annualincome@OData.Community.Display.V1.FormattedValue":  
"$68,500.00",  
    "annualincome": 68500.0,  
  
"_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
"US Dollar",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
"transactioncurrency",  
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
        "contactid": "99716234-9628-ed11-9db1-000d3a320482"  
,  
{  
    "@odata.etag": "W/\"1146771\"",  
    "fullname": "Rene Valdes (sample)",  
    "jobtitle": "Data Analyst III",  
    "annualincome@OData.Community.Display.V1.FormattedValue":  
"$86,000.00",  
    "annualincome": 86000.0,  
  
"_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
```

```

    "US Dollar",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
        "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
            "contactid": "9d716234-9628-ed11-9db1-000d3a320482"
        },
    {
        "@odata.etag": "W/\"1146773\"",
        "fullname": "Jim Glynn (sample)",
        "jobtitle": "Senior International Sales Manager",
        "annualincome@OData.Community.Display.V1.FormattedValue": "$81,400.00",
        "annualincome": 81400.0,

    "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue": "US Dollar",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
        "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
            "contactid": "a1716234-9628-ed11-9db1-000d3a320482"
        }
    ]
}

```

Salida de la consola:

text

Contacts that were created within the last 1hr:	
Full Name	Job Title
Annual Income	
-----	----- ----- ---
Susanna Stubberod (sample)	Senior Purchaser
\$52,000.00	
Nancy Anderson (sample)	Activities Manager
\$55,500.00	
Maria Cambell (sample)	Accounts Manager
\$31,000.00	
Scott Konersmann (sample)	Accounts Manager
\$38,000.00	
Robert Lyon (sample)	Senior Technician
\$78,000.00	

Paul Cannon (sample)	Ski Instructor
\$68,500.00	
Rene Valdes (sample)	Data Analyst III
\$86,000.00	
Jim Glynn (sample)	Senior International Sales Manager
\$81,400.00	

Usar operadores

Use [operadores de comparación](#) y [operadores lógicos](#) (`eq`, `ne`, `gt`, `ge`, `lt`, `le`, `and`, `or`, `not`) para refinar aún más sus resultados. En este ejemplo, solicitamos una lista de todos los contactos donde `fullname` contiene `(sample)` y los ingresos anuales son mayores que `55000`.

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/contacts?
$select=fullname,jobtitle,annualincome&$filter=contains(fullname,'(sample)')
and annualincome gt 55000 and _parentcustomerid_value eq 7d716234-9628-ed11-
9db1-000d3a320482 HTTP/1.1
Prefer: odata.include-annotations="*"
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0
Preference-Applied: odata.include-annotations="*"

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#contacts(fullname,jobtitle,annualincome)",
  "@Microsoft.Dynamics.CRM.totalrecordcount": -1,
  "@Microsoft.Dynamics.CRM.totalrecordcountlimitexceeded": false,
  "@Microsoft.Dynamics.CRM.globalmetadataversion": "1144249",
  "value": [
    {
      "@odata.etag": "W/\"1146761\"",
      "fullname": "Nancy Anderson (sample)",
      "jobtitle": "Activities Manager",
      "annualincome@OData.Community.Display.V1.FormattedValue": "
      "$55,500.00",
      "annualincome": 55500
    }
  ]
}
```

```
        "annualincome": 55500.0,  
  
        "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
        "US Dollar",  
  
        "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
        "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
        "transactioncurrency",  
            "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
            "contactid": "89716234-9628-ed11-9db1-000d3a320482"  
        },  
        {  
            "@odata.etag": "W/\"1146767\"",  
            "fullname": "Robert Lyon (sample)",  
            "jobtitle": "Senior Technician",  
            "annualincome@OData.Community.Display.V1.FormattedValue":  
            "$78,000.00",  
            "annualincome": 78000.0,  
  
            "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
            "US Dollar",  
  
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
            "transactioncurrency",  
                "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
                "contactid": "95716234-9628-ed11-9db1-000d3a320482"  
            },  
            {  
                "@odata.etag": "W/\"1146769\"",  
                "fullname": "Paul Cannon (sample)",  
                "jobtitle": "Ski Instructor",  
                "annualincome@OData.Community.Display.V1.FormattedValue":  
                "$68,500.00",  
                "annualincome": 68500.0,  
  
                "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
                "US Dollar",  
  
                "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
                "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
                "transactioncurrency",  
                    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
                    "contactid": "99716234-9628-ed11-9db1-000d3a320482"  
                },  
                {
```

```

"@odata.etag": "W/\"1146771\"",
"fullname": "Rene Valdes (sample)",
"jobtitle": "Data Analyst III",
"annualincome@OData.Community.Display.V1.FormattedValue":
"$86,000.00",
"annualincome": 86000.0,

"_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
"US Dollar",

"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",

"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
    "contactid": "9d716234-9628-ed11-9db1-000d3a320482"
},
{
    "@odata.etag": "W/\"1146773\"",
    "fullname": "Jim Glynn (sample)",
    "jobtitle": "Senior International Sales Manager",
    "annualincome@OData.Community.Display.V1.FormattedValue": "$81,400.00",
    "annualincome": 81400.0,

"_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue": "US Dollar",

"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",

"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
    "contactid": "a1716234-9628-ed11-9db1-000d3a320482"
}
]
}

```

Salida de la consola:

text

Full Name	Job Title
Nancy Anderson (sample)	Activities Manager
\$55,500.00	

Robert Lyon (sample)	Senior Technician
\$78,000.00	
Paul Cannon (sample)	Ski Instructor
\$68,500.00	
Rene Valdes (sample)	Data Analyst III
\$86,000.00	
Jim Glynn (sample)	Senior International Sales Manager
\$81,400.00	

Establecer prioridad

Dado que todos los operadores tienen la misma precedencia, utilice paréntesis para establecer el orden en que se evalúan las condiciones de consulta. Los filtros se aplican de izquierda a derecha. El orden en el que estas instrucciones aparecen en el filtro puede afectar a los resultados.

En este ejemplo, solicitamos una lista de todos los contactos donde `fullname` contiene `(sample)`, `jobtitle` contiene `senior` o `manager` y `annualincome` es mayor que `55000`. Para obtener los resultados deseados, debemos usar paréntesis para agrupar los filtros `jobtitle`. Si no lo hacemos, el operador `or` tiene la misma precedencia que los operadores `and`. La consulta en este ejemplo es similar a esta:

```
$filter=contains(fullname,'(sample)') and (contains(jobtitle,'senior') or
contains(jobtitle,'manager')) and annualincome gt 55000.
```

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/contacts?
$select=fullname,jobtitle,annualincome&$filter=contains(fullname,'(sample)')
and (contains(jobtitle, 'senior') or contains(jobtitle,'manager')) and
annualincome gt 55000 and _parentcustomerid_value eq 7d716234-9628-ed11-
9db1-000d3a320482 HTTP/1.1
Prefer: odata.include-annotations="*"
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0
Preference-Applied: odata.include-annotations="*"
```

```
{  
    "@odata.context": "[Organization  
Uri]/api/data/v9.2/$metadata#contacts(fullname,jobtitle,annualincome)",  
    "@Microsoft.Dynamics.CRM.totalrecordcount": -1,  
    "@Microsoft.Dynamics.CRM.totalrecordcountlimitexceeded": false,  
    "@Microsoft.Dynamics.CRM.globalmetadataversion": "1144249",  
    "value": [  
        {  
            "@odata.etag": "W/\"1146761\"",  
            "fullname": "Nancy Anderson (sample)",  
            "jobtitle": "Activities Manager",  
            "annualincome@OData.Community.Display.V1.FormattedValue":  
                "$55,500.00",  
            "annualincome": 55500.0,  
  
            "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
                "US Dollar",  
  
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
                "transactioncurrency",  
                "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
                "contactid": "89716234-9628-ed11-9db1-000d3a320482"  
        },  
        {  
            "@odata.etag": "W/\"1146767\"",  
            "fullname": "Robert Lyon (sample)",  
            "jobtitle": "Senior Technician",  
            "annualincome@OData.Community.Display.V1.FormattedValue":  
                "$78,000.00",  
            "annualincome": 78000.0,  
  
            "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
                "US Dollar",  
  
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
                "transactioncurrency",  
                "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
                "contactid": "95716234-9628-ed11-9db1-000d3a320482"  
        },  
        {  
            "@odata.etag": "W/\"1146773\"",  
            "fullname": "Jim Glynn (sample)",  
            "jobtitle": "Senior International Sales Manager",  
            "annualincome@OData.Community.Display.V1.FormattedValue":  
                "$81,400.00",  
            "annualincome": 81400.0,  
        }  
    ]  
}
```

```

"_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue": "US Dollar",

"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",

"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
        "contactid": "a1716234-9628-ed11-9db1-000d3a320482"
    }
]
}

```

Salida de la consola:

text

Full Name	Job Title
Annual Income	
----- ----- -----	----- ----- -----
Nancy Anderson (sample)	Activities Manager
\$55,500.00	
Robert Lyon (sample)	Senior Technician
\$78,000.00	
Jim Glynn (sample)	Senior International Sales Manager
\$81,400.00	

Sección 3: Pedidos y alias

Ordenar resultados

Para ordenar los resultados en sentido ascendente o descendente, use la opción de filtro `$orderby`. En este ejemplo, consultaremos para todos los contactos con `fullname` que contiene `(sample)` y solicitaremos los datos en orden ascendente en función del valor de propiedad `jobtitle` y luego en orden descendente basado en el valor de propiedad `annualincome` usando esta sintaxis: `$orderby=jobtitle asc, annualincome desc`. Más información: [Ordenar filas](#)

Solicitud:

HTTP

```

GET [Organization Uri]/api/data/v9.2/contacts?
$select=fullname,jobtitle,annualincome&$filter=contains(fullname,'(sample)')
and _parentcustomerid_value eq 7d716234-9628-ed11-9db1-
000d3a320482&$orderby=jobtitle asc, annualincome desc HTTP/1.1
Prefer: odata.include-annotations="*"
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

```

Respuesta:

HTTP

```

HTTP/1.1 200 OK
OData-Version: 4.0
Preference-Applied: odata.include-annotations="*"

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#contacts(fullname,jobtitle,annualincome)",
    "@Microsoft.Dynamics.CRM.totalrecordcount": -1,
    "@Microsoft.Dynamics.CRM.totalrecordcountlimitexceeded": false,
    "@Microsoft.Dynamics.CRM.globalmetadataversion": "1144249",
    "value": [
        {
            "@odata.etag": "W/\"1146765\",
            "fullname": "Scott Konersmann (sample)",
            "jobtitle": "Accounts Manager",
            "annualincome@OData.Community.Display.V1.FormattedValue": "$38,000.00",
            "annualincome": 38000.0,
            "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue": "US Dollar",
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
            "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-
00224802b2ac",
            "contactid": "91716234-9628-ed11-9db1-000d3a320482"
        },
        {
            "@odata.etag": "W/\"1146763\",
            "fullname": "Maria Cambell (sample)",
            "jobtitle": "Accounts Manager",
            "annualincome@OData.Community.Display.V1.FormattedValue": "$31,000.00",
            "annualincome": 31000.0,
        }
    ]
}

```

```
"_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
"US Dollar",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
"transactioncurrency",  
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
        "contactid": "8d716234-9628-ed11-9db1-000d3a320482"  
,  
{  
    "@odata.etag": "W/\"1146761\"",  
    "fullname": "Nancy Anderson (sample)",  
    "jobtitle": "Activities Manager",  
    "annualincome@OData.Community.Display.V1.FormattedValue":  
"$55,500.00",  
    "annualincome": 55500.0,  
  
"_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
"US Dollar",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
"transactioncurrency",  
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
        "contactid": "89716234-9628-ed11-9db1-000d3a320482"  
,  
{  
    "@odata.etag": "W/\"1146771\"",  
    "fullname": "Rene Valdes (sample)",  
    "jobtitle": "Data Analyst III",  
    "annualincome@OData.Community.Display.V1.FormattedValue":  
"$86,000.00",  
    "annualincome": 86000.0,  
  
"_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
"US Dollar",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
"transactioncurrency",  
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
        "contactid": "9d716234-9628-ed11-9db1-000d3a320482"  
,  
{  
    "@odata.etag": "W/\"1146773\"",
```

```
        "fullname": "Jim Glynn (sample)",  
        "jobtitle": "Senior International Sales Manager",  
        "annualincome@OData.Community.Display.V1.FormattedValue":  
        "$81,400.00",  
        "annualincome": 81400.0,  
  
        "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
        "US Dollar",  
  
        "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
        "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
        "transactioncurrency",  
            "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
            "contactid": "a1716234-9628-ed11-9db1-000d3a320482"  
,  
        {  
            "@odata.etag": "W/\"1146759\"",  
            "fullname": "Susanna Stubberod (sample)",  
            "jobtitle": "Senior Purchaser",  
            "annualincome@OData.Community.Display.V1.FormattedValue":  
            "$52,000.00",  
            "annualincome": 52000.0,  
  
            "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
            "US Dollar",  
  
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
            "transactioncurrency",  
                "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
                "contactid": "85716234-9628-ed11-9db1-000d3a320482"  
,  
            {  
                "@odata.etag": "W/\"1146767\"",  
                "fullname": "Robert Lyon (sample)",  
                "jobtitle": "Senior Technician",  
                "annualincome@OData.Community.Display.V1.FormattedValue":  
                "$78,000.00",  
                "annualincome": 78000.0,  
  
                "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
                "US Dollar",  
  
                "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
                "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
                "transactioncurrency",  
                    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-
```

```

00224802b2ac",
    "contactid": "95716234-9628-ed11-9db1-000d3a320482"
},
{
    "@odata.etag": "W/\"1146769\"",
    "fullname": "Paul Cannon (sample)",
    "jobtitle": "Ski Instructor",
    "annualincome@odata.Community.Display.V1.FormattedValue":
"$68,500.00",
    "annualincome": 68500.0,
    "_transactioncurrencyid_value@odata.Community.Display.V1.FormattedValue":
"US Dollar",
    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",
    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
    "contactid": "99716234-9628-ed11-9db1-000d3a320482"
}
]
}

```

Salida de la consola:

text

Annual Income	Full Name	Job Title
\$38,000.00	Scott Konersmann (sample)	Accounts Manager
\$31,000.00	Maria Cambell (sample)	Accounts Manager
\$55,500.00	Nancy Anderson (sample)	Activities Manager
\$86,000.00	Rene Valdes (sample)	Data Analyst III
\$81,400.00	Jim Glynn (sample)	Senior International Sales Manager
\$52,000.00	Susanna Stubberod (sample)	Senior Purchaser
\$78,000.00	Robert Lyon (sample)	Senior Technician
\$68,500.00	Paul Cannon (sample)	Ski Instructor

Alias de parámetro

Use [alias de parámetro](#) para reutilizar más fácilmente los parámetros en los filtros.

Puede usar alias en las opciones `$filter` y `$orderby`. Si al alias no se asigna un valor se da por hecho que es nulo.

También puede usar alias de parámetro al [llamar funciones](#). Por ejemplo, podemos reescribir la consulta de la operación de resultados con el orden anterior para usar parámetros y obtener la misma salida.

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/contacts?
$select=fullname,jobtitle,annualincome&$filter=contains(@p1, '(sample)') and
@p2 eq @p3&$orderby=@p4 asc, @p5
desc&@p1=fullname&@p2=_parentcustomerid_value&@p3=7d716234-9628-ed11-9db1-
000d3a320482&@p4=jobtitle&@p5=annualincome HTTP/1.1
Prefer: odata.include-annotations="*"
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0
Preference-Applied: odata.include-annotations="*"

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#contacts(fullname,jobtitle,annualincome)",
    "@Microsoft.Dynamics.CRM.totalrecordcount": -1,
    "@Microsoft.Dynamics.CRM.totalrecordcountlimitexceeded": false,
    "@Microsoft.Dynamics.CRM.globalmetadataversion": "1144249",
    "value": [
        {
            "@odata.etag": "W/\"1146765\"",
            "fullname": "Scott Konersmann (sample)",
            "jobtitle": "Accounts Manager",
            "annualincome@OData.Community.Display.V1.FormattedValue":
"$38,000.00",
            "annualincome": 38000.0,
            "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
"US Dollar",
            "transactioncurrencyid": "7d716234-9628-ed11-9db1-000d3a320482"
        }
    ]
}
```

```
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",

"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
        "contactid": "91716234-9628-ed11-9db1-000d3a320482"
    },
{
    "@odata.etag": "W/\"1146763\",
    "fullname": "Maria Cambell (sample)",
    "jobtitle": "Accounts Manager",
    "annualincome@OData.Community.Display.V1.FormattedValue": "$31,000.00",
    "annualincome": 31000.0,

"_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue": "US Dollar",

"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",

"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
        "contactid": "8d716234-9628-ed11-9db1-000d3a320482"
    },
{
    "@odata.etag": "W/\"1146761\",
    "fullname": "Nancy Anderson (sample)",
    "jobtitle": "Activities Manager",
    "annualincome@OData.Community.Display.V1.FormattedValue": "$55,500.00",
    "annualincome": 55500.0,

"_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue": "US Dollar",

"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",

"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
        "contactid": "89716234-9628-ed11-9db1-000d3a320482"
    },
{
    "@odata.etag": "W/\"1146771\",
    "fullname": "Rene Valdes (sample)",
    "jobtitle": "Data Analyst III",
    "annualincome@OData.Community.Display.V1.FormattedValue":
```

```
"$86,000.00",
    "annualincome": 86000.0,

    "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
    "US Dollar",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro
perty": "transactioncurrencyid",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
    "transactioncurrency",
        "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-
00224802b2ac",
            "contactid": "9d716234-9628-ed11-9db1-000d3a320482"
        },
        {
            "@odata.etag": "W/\"1146773\",
            "fullname": "Jim Glynn (sample)",
            "jobtitle": "Senior International Sales Manager",
            "annualincome@OData.Community.Display.V1.FormattedValue":
"$81,400.00",
            "annualincome": 81400.0,

    "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
    "US Dollar",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro
perty": "transactioncurrencyid",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
    "transactioncurrency",
        "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-
00224802b2ac",
            "contactid": "a1716234-9628-ed11-9db1-000d3a320482"
        },
        {
            "@odata.etag": "W/\"1146759\",
            "fullname": "Susanna Stubberod (sample)",
            "jobtitle": "Senior Purchaser",
            "annualincome@OData.Community.Display.V1.FormattedValue":
"$52,000.00",
            "annualincome": 52000.0,

    "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
    "US Dollar",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro
perty": "transactioncurrencyid",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
    "transactioncurrency",
        "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-
00224802b2ac",
            "contactid": "85716234-9628-ed11-9db1-000d3a320482"
        },
```

```
{
    "@odata.etag": "W/\"1146767\"",
    "fullname": "Robert Lyon (sample)",
    "jobtitle": "Senior Technician",
    "annualincome@OData.Community.Display.V1.FormattedValue":
    "$78,000.00",
    "annualincome": 78000.0,
    "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
    "US Dollar",
    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",
    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
    "contactid": "95716234-9628-ed11-9db1-000d3a320482"
},
{
    "@odata.etag": "W/\"1146769\"",
    "fullname": "Paul Cannon (sample)",
    "jobtitle": "Ski Instructor",
    "annualincome@OData.Community.Display.V1.FormattedValue":
    "$68,500.00",
    "annualincome": 68500.0,
    "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
    "US Dollar",
    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",
    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
    "contactid": "99716234-9628-ed11-9db1-000d3a320482"
}
]
```

Salida de la consola:

text
Contacts ordered by jobtitle (Ascending) and annualincome (descending) Full Name Job Title Annual Income ----- ----- ----- ----- Scott Konersmann (sample) Accounts Manager

```

|$38,000.00
|  |Maria Cambell (sample)      |Accounts Manager
|$31,000.00
|  |Nancy Anderson (sample)    |Activities Manager
|$55,500.00
|  |Rene Valdes (sample)       |Data Analyst III
|$86,000.00
|  |Jim Glynn (sample)        |Senior International Sales Manager
|$81,400.00
|  |Susanna Stubberod (sample)|Senior Purchaser
|$52,000.00
|  |Robert Lyon (sample)      |Senior Technician
|$78,000.00
|  |Paul Cannon (sample)       |Ski Instructor
|$68,500.00

```

Sección 4: Limitar y contar los resultados

Como práctica recomendada, no devuelva más datos de los que necesita. Para proteger el rendimiento, el servidor devuelve un máximo de 5000 filas de tabla por solicitud.

Para restringir el número de resultados devueltos, [use la opción de consulta \\$top o agregue odata.maxpagesize](#) en el encabezado de solicitud. La opción de consulta `$top` devuelve el número superior de filas del conjunto de resultados e ignora el resto. El encabezado de solicitud `odata.maxpagesize` especifica el número de filas para devolver por página con una propiedad `@odata.nextLink` para obtener los resultados de la página siguiente. [Obtenga más información sobre los resultados de página.](#)

Resultados relevantes

Podemos aplicar la opción de consulta `$top` para limitar la operación de consulta básica a los cinco primeros contactos con `fullname` que contienen `(sample)`. En este caso, la solicitud genera realmente al menos 10 resultados, pero solo las primeras 5 entradas se devuelven en la respuesta.

Solicitud:

HTTP

```

GET [Organization Uri]/api/data/v9.2/contacts?
$select=fullname,jobtitle,annualincome&$filter=contains(fullname,'(sample)')
and _parentcustomerid_value eq 7d716234-9628-ed11-9db1-000d3a320482&$top=5
HTTP/1.1
Prefer: odata.include-annotations="*"
OData-MaxVersion: 4.0
OData-Version: 4.0

```

```
If-None-Match: null  
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK  
OData-Version: 4.0  
Preference-Applied: odata.include-annotations="*"  
  
{  
    "@odata.context": "[Organization  
Uri]/api/data/v9.2/$metadata#contacts(fullname,jobtitle,annualincome)",  
    "@Microsoft.Dynamics.CRM.totalrecordcount": -1,  
    "@Microsoft.Dynamics.CRM.totalrecordcountlimitexceeded": false,  
    "@Microsoft.Dynamics.CRM.globalmetadataversion": "1144249",  
    "value": [  
        {  
            "@odata.etag": "W/\"1146759\"",  
            "fullname": "Susanna Stubberod (sample)",  
            "jobtitle": "Senior Purchaser",  
            "annualincome@OData.Community.Display.V1.FormattedValue":  
"$52,000.00",  
            "annualincome": 52000.0,  
  
            "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
"US Dollar",  
  
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
"transactioncurrency",  
            "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
            "contactid": "85716234-9628-ed11-9db1-000d3a320482"  
        },  
        {  
            "@odata.etag": "W/\"1146761\"",  
            "fullname": "Nancy Anderson (sample)",  
            "jobtitle": "Activities Manager",  
            "annualincome@OData.Community.Display.V1.FormattedValue":  
"$55,500.00",  
            "annualincome": 55500.0,  
  
            "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
"US Dollar",  
  
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
```

```
"transactioncurrency",
  "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-
00224802b2ac",
    "contactid": "89716234-9628-ed11-9db1-000d3a320482"
  },
  {
    "@odata.etag": "W/\"1146763\",
    "fullname": "Maria Cambell (sample)",
    "jobtitle": "Accounts Manager",
    "annualincome@OData.Community.Display.V1.FormattedValue":
"$31,000.00",
      "annualincome": 31000.0,

    "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
"US Dollar",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro-
perty": "transactioncurrencyid",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
"transactioncurrency",
      "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-
00224802b2ac",
        "contactid": "8d716234-9628-ed11-9db1-000d3a320482"
      },
      {
        "@odata.etag": "W/\"1146765\",
        "fullname": "Scott Konersmann (sample)",
        "jobtitle": "Accounts Manager",
        "annualincome@OData.Community.Display.V1.FormattedValue":
"$38,000.00",
          "annualincome": 38000.0,

        "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
"US Dollar",

        "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro-
perty": "transactioncurrencyid",

        "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
"transactioncurrency",
          "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-
00224802b2ac",
            "contactid": "91716234-9628-ed11-9db1-000d3a320482"
          },
          {
            "@odata.etag": "W/\"1146767\",
            "fullname": "Robert Lyon (sample)",
            "jobtitle": "Senior Technician",
            "annualincome@OData.Community.Display.V1.FormattedValue":
"$78,000.00",
              "annualincome": 78000.0,

            "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
"US Dollar",
```

```
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",
    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
        "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
            "contactid": "95716234-9628-ed11-9db1-000d3a320482"
        }
    ]
}
```

Salida de la consola:

```
text
```

```
Contacts top 5 results:
|Full Name |Job Title
|Annual Income
-----+-----+-----
-----+-----+
|Susanna Stubberod (sample) |Senior Purchaser
|$52,000.00
|Nancy Anderson (sample) |Activities Manager
|$55,500.00
|Maria Cambell (sample) |Accounts Manager
|$31,000.00
|Scott Konersmann (sample) |Accounts Manager
|$38,000.00
|Robert Lyon (sample) |Senior Technician
|$78,000.00
```

Recuento de colección

Si solo desea la cantidad de registros en una colección, agregue `/$count` a la URL de la colección. El valor máximo es 5,000.

Solicitud:

```
HTTP
```

```
GET [Organization Uri]/api/data/v9.2/contacts/$count HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0
```

9

Salida de la consola:

text

```
The contacts collection has 9 contacts.
```

Número de resultados

Puede obtener [el recuento de filas](#) de una propiedad con valor de colección o un recuento de filas de tabla coincidentes en un filtro. El recuento les dice el número de filas posibles en su resultado. Sin embargo, el servidor de Dataverse devuelve 5000 como recuento máximo, aunque el resultado puede tener más.

En este ejemplo, construimos un filtro donde `jobtitle` contiene `Senior` o `Manager` y también solicitamos un `$count` del resultado. La respuesta contiene el recuento en la propiedad `@odata.count` junto con los resultados de la consulta.

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/contacts?
$select=fullname,jobtitle,annualincome&$filter=(contains(jobtitle,'senior')
or contains(jobtitle, 'manager')) and _parentcustomerid_value eq 7d716234-
9628-ed11-9db1-000d3a320482&$count=true HTTP/1.1
Prefer: odata.include-annotations="*"
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0
Preference-Applied: odata.include-annotations="*"
```

```
{  
    "@odata.context": "[Organization  
Uri]/api/data/v9.2/$metadata#contacts(fullname,jobtitle,annualincome)",  
    "@odata.count": 6,  
    "@Microsoft.Dynamics.CRM.totalrecordcount": 6,  
    "@Microsoft.Dynamics.CRM.totalrecordcountlimitexceeded": false,  
    "@Microsoft.Dynamics.CRM.globalmetadataversion": "1144249",  
    "value": [  
        {  
            "@odata.etag": "W/\"1146759\"",  
            "fullname": "Susanna Stubberod (sample)",  
            "jobtitle": "Senior Purchaser",  
            "annualincome@OData.Community.Display.V1.FormattedValue":  
                "$52,000.00",  
            "annualincome": 52000.0,  
  
            "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
                "US Dollar",  
  
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
                "transactioncurrency",  
            "transactioncurrencyid": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
            "contactid": "85716234-9628-ed11-9db1-000d3a320482"  
        },  
        {  
            "@odata.etag": "W/\"1146761\"",  
            "fullname": "Nancy Anderson (sample)",  
            "jobtitle": "Activities Manager",  
            "annualincome@OData.Community.Display.V1.FormattedValue":  
                "$55,500.00",  
            "annualincome": 55500.0,  
  
            "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
                "US Dollar",  
  
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
                "transactioncurrency",  
            "transactioncurrencyid": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
            "contactid": "89716234-9628-ed11-9db1-000d3a320482"  
        },  
        {  
            "@odata.etag": "W/\"1146763\"",  
            "fullname": "Maria Cambell (sample)",  
            "jobtitle": "Accounts Manager",  
            "annualincome@OData.Community.Display.V1.FormattedValue":  
                "$31,000.00",  
            "annualincome": 31000.0,  
        }  
    ]  
}
```

```
"_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
"US Dollar",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
"transactioncurrency",  
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
        "contactid": "8d716234-9628-ed11-9db1-000d3a320482"  
    },  
    {  
        "@odata.etag": "W/\"1146765\"",  
        "fullname": "Scott Konersmann (sample)",  
        "jobtitle": "Accounts Manager",  
        "annualincome@OData.Community.Display.V1.FormattedValue":  
"$38,000.00",  
        "annualincome": 38000.0,  
  
"_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
"US Dollar",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
"transactioncurrency",  
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
        "contactid": "91716234-9628-ed11-9db1-000d3a320482"  
    },  
    {  
        "@odata.etag": "W/\"1146767\"",  
        "fullname": "Robert Lyon (sample)",  
        "jobtitle": "Senior Technician",  
        "annualincome@OData.Community.Display.V1.FormattedValue":  
"$78,000.00",  
        "annualincome": 78000.0,  
  
"_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
"US Dollar",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
"transactioncurrency",  
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
        "contactid": "95716234-9628-ed11-9db1-000d3a320482"  
    },  
    {  
        "@odata.etag": "W/\"1146773\"",
```

```

    "fullname": "Jim Glynn (sample)",
    "jobtitle": "Senior International Sales Manager",
    "annualincome@odata.Community.Display.V1.FormattedValue":
    "$81,400.00",
    "annualincome": 81400.0,

    "_transactioncurrencyid_value@odata.Community.Display.V1.FormattedValue":
    "US Dollar",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro
    perty": "transactioncurrencyid",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
    "transactioncurrency",
        "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-
    00224802b2ac",
        "contactid": "a1716234-9628-ed11-9db1-000d3a320482"
    }
]
}

```

Salida de la consola:

text

	Full Name	Job Title
Annual Income	-----	-----
-----	-----	-----
\$52,000.00	Susanna Stubberod (sample)	Senior Purchaser
\$55,500.00	Nancy Anderson (sample)	Activities Manager
\$31,000.00	Maria Cambell (sample)	Accounts Manager
\$38,000.00	Scott Konersmann (sample)	Accounts Manager
\$78,000.00	Robert Lyon (sample)	Senior Technician
\$81,400.00	Jim Glynn (sample)	Senior International Sales Manager

Sección 5: Paginación

Algunas consultas devuelven una gran cantidad de filas y es más fácil verlas una página a la vez. Para [recuperar resultados paginados](#), utilice la opción `odata.maxpagesize` en lugar de `$top`.

En este ejemplo, pedimos `$count` y establecemos `odata.maxpagesize` como `4`. Este filtro coincide con 10 contactos, pero solo estamos recuperando 4 a la vez. También usamos el recuento y el tamaño máximo de página para averiguar el número de páginas que hay. El resultado de la primera página se devuelve en esta solicitud.

Solicitud:

```
HTTP

GET [Organization Uri]/api/data/v9.2/contacts?
$select=fullname,jobtitle,annualincome&$filter=contains(fullname,'(sample)')
&$count=true HTTP/1.1
Prefer: odata.maxpagesize=4; odata.include-annotations="*"
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

```
HTTP

HTTP/1.1 200 OK
OData-Version: 4.0
Preference-Applied: odata.include-annotations="*"; odata.maxpagesize=4

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#contacts(fullname,jobtitle,annualincome)",
    "@odata.count": 9,
    "@Microsoft.Dynamics.CRM.totalrecordcount": 9,
    "@Microsoft.Dynamics.CRM.totalrecordcountlimitexceeded": false,
    "@Microsoft.Dynamics.CRM.globalmetadataversion": "1144249",
    "value": [
        {
            "@odata.etag": "W/\"1146626\"",
            "fullname": "Yvonne McKay (sample)",
            "jobtitle": "Coffee Master",
            "annualincome@OData.Community.Display.V1.FormattedValue":
            "$45,000.00",
            "annualincome": 45000.0,
            "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
            "US Dollar",
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro
            perty": "transactioncurrencyid",
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
            "transactioncurrency",
            "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-
```

```
00224802b2ac",
    "contactid": "81716234-9628-ed11-9db1-000d3a320482"
},
{
    "@odata.etag": "W/\"1146759\",
    "fullname": "Susanna Stubberod (sample)",
    "jobtitle": "Senior Purchaser",
    "annualincome@OData.Community.Display.V1.FormattedValue":
"$52,000.00",
    "annualincome": 52000.0,

    "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
"US Dollar",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro
perty": "transactioncurrencyid",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
"transactioncurrency",
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-
00224802b2ac",
    "contactid": "85716234-9628-ed11-9db1-000d3a320482"
},
{
    "@odata.etag": "W/\"1146761\",
    "fullname": "Nancy Anderson (sample)",
    "jobtitle": "Activities Manager",
    "annualincome@OData.Community.Display.V1.FormattedValue":
"$55,500.00",
    "annualincome": 55500.0,

    "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
"US Dollar",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro
perty": "transactioncurrencyid",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
"transactioncurrency",
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-
00224802b2ac",
    "contactid": "89716234-9628-ed11-9db1-000d3a320482"
},
{
    "@odata.etag": "W/\"1146763\",
    "fullname": "Maria Cambell (sample)",
    "jobtitle": "Accounts Manager",
    "annualincome@OData.Community.Display.V1.FormattedValue":
"$31,000.00",
    "annualincome": 31000.0,

    "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
"US Dollar",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro
```

```

    "property": "transactioncurrencyid",
    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
    "transactioncurrency",
        "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-
00224802b2ac",
            "contactid": "8d716234-9628-ed11-9db1-000d3a320482"
        }
    ],
    "@odata.nextLink": "[Organization Uri]/api/data/v9.2/contacts?
$select=fullname,jobtitle,annualincome&$filter=contains(fullname,'(sample)')
&$count=true&$skiptoken=%3Ccookie%20pagenumber=%222%22%20pagingcookie=%22%25
3ccookie%2520page%253d%25221%2522%253e%253ccontactid%2520last%253d%2522%257b
8D716234-9628-ED11-9DB1-
000D3A320482%257d%2522%2520first%253d%2522%257b81716234-9628-ED11-9DB1-
000D3A320482%257d%2522%2520%252f%253e%253c%252fcookie%253e%22%20istracking=%
22False%22%20/%3E"
}

```

Salida de la consola:

text

	Full Name	Job Title
Annual Income	----- ----- -----	

\$45,000.00	Yvonne McKay (sample)	Coffee Master
\$52,000.00	Susanna Stubberod (sample)	Senior Purchaser
\$55,500.00	Nancy Anderson (sample)	Activities Manager
\$31,000.00	Maria Cambell (sample)	Accounts Manager

Para recuperar la página 2, use una solicitud `GET` con el valor de la propiedad `@odata.nextLink`.

Solicitud:

HTTP

```

GET [Organization Uri]/api/data/v9.2/contacts?
$select=fullname,jobtitle,annualincome&$filter=contains(fullname,'(sample)')
&$count=true&$skiptoken=<cookie pagename="2"
pagingcookie="%3ccookie%20page%3d%221%22%3e%3ccontactid%20last%3d%22%7b8D716
234-9628-ED11-9DB1-000D3A320482%7d%22%20first%3d%22%7b81716234-9628-ED11-
9DB1-000D3A320482%7d%22%20%2f%3e%3c%2fcookie%3e" istracking="False" />

```

```
HTTP/1.1
Prefer: odata.maxpagesize=4; odata.include-annotations="*"
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0
Preference-Applied: odata.include-annotations="*"; odata.maxpagesize=4

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#contacts(fullname,jobtitle,annualincome)",
    "@odata.count": 9,
    "@Microsoft.Dynamics.CRM.totalrecordcount": 9,
    "@Microsoft.Dynamics.CRM.totalrecordcountlimitexceeded": false,
    "@Microsoft.Dynamics.CRM.globalmetadataversion": "1144249",
    "value": [
        {
            "@odata.etag": "W/\"1146765\",
            "fullname": "Scott Konersmann (sample)",
            "jobtitle": "Accounts Manager",
            "annualincome@OData.Community.Display.V1.FormattedValue": "$38,000.00",
            "annualincome": 38000.0,
            "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue": "US Dollar",
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
            "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
            "contactid": "91716234-9628-ed11-9db1-000d3a320482"
        },
        {
            "@odata.etag": "W/\"1146767\",
            "fullname": "Robert Lyon (sample)",
            "jobtitle": "Senior Technician",
            "annualincome@OData.Community.Display.V1.FormattedValue": "$78,000.00",
            "annualincome": 78000.0,
            "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue": "US Dollar",
        }
    ]
}
```

```
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",

"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
        "contactid": "95716234-9628-ed11-9db1-000d3a320482"
    },
{
    "@odata.etag": "W/\"1146769\",
    "fullname": "Paul Cannon (sample)",
    "jobtitle": "Ski Instructor",
    "annualincome@OData.Community.Display.V1.FormattedValue": "$68,500.00",
    "annualincome": 68500.0,

"_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue": "US Dollar",

"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",

"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
        "contactid": "99716234-9628-ed11-9db1-000d3a320482"
    },
{
    "@odata.etag": "W/\"1146771\",
    "fullname": "Rene Valdes (sample)",
    "jobtitle": "Data Analyst III",
    "annualincome@OData.Community.Display.V1.FormattedValue": "$86,000.00",
    "annualincome": 86000.0,

"_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue": "US Dollar",

"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",

"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
        "contactid": "9d716234-9628-ed11-9db1-000d3a320482"
    }
],
"@odata.nextLink": "[Organization Uri]/api/data/v9.2/contacts?$select=fullname,jobtitle,annualincome&$filter=contains(fullname,'(sample)')&$count=true&$skiptoken=%3Ccookie%20pagenumber=%223%22%20pagingcookie=%22%253ccookie%2520page%253d%25222%2522%253e%253ccontactid%2520last%253d%2522%257b
```

```
9D716234-9628-ED11-9DB1-
000D3A320482%257d%2522%2520first%253d%2522%257b91716234-9628-ED11-9DB1-
000D3A320482%257d%2522%2520%252f%253e%253c%252fcookie%253e%22%20istracking=%
22False%22%20/%3E"
}
```

Salida de la consola:

text		
Page 2 of 3:		
Full Name Job Title		
Annual Income ----- ----- -----		
----- Scott Konersmann (sample) Accounts Manager		
\$38,000.00 ----- ----- -----		
Robert Lyon (sample) Senior Technician		
\$78,000.00 ----- ----- -----		
Paul Cannon (sample) Ski Instructor		
\$68,500.00 ----- ----- -----		
Rene Valdes (sample) Data Analyst III		
\$86,000.00		

Sección 6: Expandir resultados

Para recuperar información sobre las filas de la tabla asociadas, utilice la opción de consulta `$expand` sobre las propiedades de navegación. Más información: [Unir tablas](#)

Expandir la propiedad de navegación de un solo valor

Una propiedad de navegación de un solo valor representa una relación de varios a uno. En nuestros datos de muestra, la cuenta tiene una relación con un contacto a través de la columna `primarycontactid` o atributo. En esta relación, la cuenta solo puede tener solo un contacto principal. Con el [EntityType de cuenta](#), podemos crear una consulta para obtener información sobre la cuenta y información ampliada sobre su contacto principal.

Solicitud:

HTTP
<pre>GET [Organization Uri]/api/data/v9.2/accounts(7d716234-9628-ed11-9db1- 000d3a320482)? \$select=name&\$expand=primarycontactid(\$select=fullname,jobtitle,annualincome) HTTP/1.1</pre>

```
Prefer: odata.include-annotations="*"
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
ETag: W/"1146751"
OData-Version: 4.0
Preference-Applied: odata.include-annotations="*"

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#accounts(name,primarycontactid(fullname,jobtitle,
annualincome))/$entity",
    "@odata.etag": "W/\"1146751\",
    "name": "Contoso, Ltd. (sample)",
    "accountid": "7d716234-9628-ed11-9db1-000d3a320482",
    "primarycontactid": {
        "@odata.etag": "W/\"1146626\",
        "fullname": "Yvonne McKay (sample)",
        "jobtitle": "Coffee Master",
        "annualincome@OData.Community.Display.V1.FormattedValue": "$45,000.00",
        "annualincome": 45000.0,
        "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue": "US Dollar",

        "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",
        "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
        "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
        "contactid": "81716234-9628-ed11-9db1-000d3a320482"
    }
}
```

Salida de la consola:

text

```
Account Contoso, Ltd. (sample) has the following primary contact person:
    Fullname: Yvonne McKay (sample)
    Jobtitle: Coffee Master
    Annualincome: 45000
```

Expandir la propiedad de asociados

Cada propiedad de navegación tiene una propiedad "partner" correspondiente. Después de que se crea una asociación, podemos recuperar la información a través de esta. La columna que usemos depende de la tabla base que consultemos.

Por ejemplo, en la operación anterior, creamos una consulta contra el [tipo de entidad de cuenta](#). Queríamos obtener más información sobre su contacto principal, por lo que usamos la columna o atributo `primarycontactid`. Si buscamos el [tipo de entidad de cuenta](#), en la sección de [propiedades de navegación de valor único](#) encontramos que el socio propiedad de `primarycontactid` es la propiedad de navegación con valor de colección `account_primary_contact` que se encuentra en el [tipo de entidad de contacto](#).

Si genera una consulta contra un contacto, puede ampliar la columna `account_primary_contact` para obtener información sobre las cuentas en las que el contacto es el contacto principal. En los datos de ejemplo, `Yvonne McKay (sample)` es la persona de contacto principal únicamente para una cuenta. Sin embargo, Yvonne puede asignarse potencialmente a otras cuentas como el contacto principal. Porque la propiedad `account_primary_contact` tiene una relación de muchos a uno, el resultado se devuelve como una matriz de filas de cuenta.

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/contacts(81716234-9628-ed11-9db1-000d3a320482)?  
$select=fullname,jobtitle,annualincome&$expand=account_primary_contact($sele  
ct=name) HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK  
ETag: W/"1146626"  
OData-Version: 4.0  
  
{  
    "@odata.context": "[Organization  
Uri]/api/data/v9.2/$metadata#contacts(fullname,jobtitle,annualincome,account  
_primary_contact(name))/$entity",  
    "id": "81716234-9628-ed11-9db1-000d3a320482",  
    "fullname": "Yvonne McKay",  
    "jobtitle": "Sales Representative",  
    "annualincome": 100000,  
    "account_primary_contact": {  
        "id": "81716234-9628-ed11-9db1-000d3a320482",  
        "name": "Acme Corp."  
    }  
}
```

```
"@odata.etag": "W/\"1146626\"",  
"fullname": "Yvonne McKay (sample)",  
"jobtitle": "Coffee Master",  
"annualincome": 45000.0,  
"_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",  
"contactid": "81716234-9628-ed11-9db1-000d3a320482",  
"account_primary_contact": [  
    {  
        "@odata.etag": "W/\"1146751\"",  
        "name": "Contoso, Ltd. (sample)",  
        "accountid": "7d716234-9628-ed11-9db1-000d3a320482"  
    }  
]  
}
```

Salida de la consola:

```
text
```

```
Contact 'Yvonne McKay (sample)' is the primary contact for the following  
accounts:  
    Contoso, Ltd. (sample)
```

Expandir la propiedad de navegación valorada como colección

Las propiedades de navegación valoradas como colección admiten relaciones de uno a varios o de varios a varios. Por ejemplo, en nuestros datos de muestra, la cuenta tiene una relación con muchos contactos a través de la columna `contact_customer_accounts` o atributo.

Podemos usar el [tipo de entidad de cuenta](#) en una consulta para obtener información sobre la cuenta y expanda información sobre sus contactos. En este caso, el `Contoso, Ltd. (sample)` se asocia a nueve contactos mediante la propiedad de navegación valorada como colección `contact_customer_accounts`.

Solicitud:

```
HTTP
```

```
GET [Organization Uri]/api/data/v9.2/accounts(7d716234-9628-ed11-9db1-  
000d3a320482)?  
$select=name&$expand=contact_customer_accounts($select=fullname,jobtitle,ann  
ualincome) HTTP/1.1  
Prefer: odata.include-annotations="*"  
OData-MaxVersion: 4.0  
OData-Version: 4.0
```

```
If-None-Match: null  
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK  
ETag: W/"1146751"  
OData-Version: 4.0  
Preference-Applied: odata.include-annotations="*"  
  
{  
    "@odata.context": "[Organization  
Uri]/api/data/v9.2/$metadata#accounts(name,contact_customer_accounts(fullnam  
e,jobtitle,annualincome))/$entity",  
    "@odata.etag": "W/\"1146751\"",  
    "name": "Contoso, Ltd. (sample)",  
    "accountid": "7d716234-9628-ed11-9db1-000d3a320482",  
    "contact_customer_accounts": [  
        {  
            "@odata.etag": "W/\"1146759\"",  
            "fullname": "Susanna Stubberod (sample)",  
            "jobtitle": "Senior Purchaser",  
            "annualincome@OData.Community.Display.V1.FormattedValue":  
            "$52,000.00",  
            "annualincome": 52000.0,  
  
            "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
            "US Dollar",  
  
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
            "transactioncurrency",  
            "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
            "contactid": "85716234-9628-ed11-9db1-000d3a320482"  
        },  
        {  
            "@odata.etag": "W/\"1146761\"",  
            "fullname": "Nancy Anderson (sample)",  
            "jobtitle": "Activities Manager",  
            "annualincome@OData.Community.Display.V1.FormattedValue":  
            "$55,500.00",  
            "annualincome": 55500.0,  
  
            "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
            "US Dollar",  
  
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",
```

```
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
"transactioncurrency",  
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
        "contactid": "89716234-9628-ed11-9db1-000d3a320482"  
    },  
    {  
        "@odata.etag": "W/\"1146763\"",  
        "fullname": "Maria Cambell (sample)",  
        "jobtitle": "Accounts Manager",  
        "annualincome@OData.Community.Display.V1.FormattedValue":  
"$31,000.00",  
        "annualincome": 31000.0,  
  
"_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
"US Dollar",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
"transactioncurrency",  
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
        "contactid": "8d716234-9628-ed11-9db1-000d3a320482"  
    },  
    {  
        "@odata.etag": "W/\"1146765\"",  
        "fullname": "Scott Konersmann (sample)",  
        "jobtitle": "Accounts Manager",  
        "annualincome@OData.Community.Display.V1.FormattedValue":  
"$38,000.00",  
        "annualincome": 38000.0,  
  
"_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
"US Dollar",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
"transactioncurrency",  
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
        "contactid": "91716234-9628-ed11-9db1-000d3a320482"  
    },  
    {  
        "@odata.etag": "W/\"1146767\"",  
        "fullname": "Robert Lyon (sample)",  
        "jobtitle": "Senior Technician",  
        "annualincome@OData.Community.Display.V1.FormattedValue":  
"$78,000.00",  
        "annualincome": 78000.0,
```

```
"_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
"US Dollar",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
"transactioncurrency",  
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
        "contactid": "95716234-9628-ed11-9db1-000d3a320482"  
    },  
    {  
        "@odata.etag": "W/\"1146769\"",  
        "fullname": "Paul Cannon (sample)",  
        "jobtitle": "Ski Instructor",  
        "annualincome@OData.Community.Display.V1.FormattedValue":  
"$68,500.00",  
        "annualincome": 68500.0,  
  
"_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
"US Dollar",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
"transactioncurrency",  
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
        "contactid": "99716234-9628-ed11-9db1-000d3a320482"  
    },  
    {  
        "@odata.etag": "W/\"1146771\"",  
        "fullname": "Rene Valdes (sample)",  
        "jobtitle": "Data Analyst III",  
        "annualincome@OData.Community.Display.V1.FormattedValue":  
"$86,000.00",  
        "annualincome": 86000.0,  
  
"_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
"US Dollar",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
"transactioncurrency",  
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
        "contactid": "9d716234-9628-ed11-9db1-000d3a320482"  
    },  
    {  
        "@odata.etag": "W/\"1146773\"",  
        "fullname": "Jim Glynn (sample)",
```

```
"jobtitle": "Senior International Sales Manager",
"annualincome@odata.Community.Display.V1.FormattedValue":
"$81,400.00",
"annualincome": 81400.0,

"_transactioncurrencyid_value@odata.Community.Display.V1.FormattedValue":
"US Dollar",

"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",

"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
"transactioncurrency",
"_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-
00224802b2ac",
"contactid": "a1716234-9628-ed11-9db1-000d3a320482"
}

]
}
```

Salida de la consola:

text

Account 'Contoso, Ltd. (sample)' has the following contact customers:

	Full Name	Job Title
Annual Income	-----	----- -----

	Susanna Stubberod (sample)	Senior Purchaser
\$52,000.00		
	Nancy Anderson (sample)	Activities Manager
\$55,500.00		
	Maria Cambell (sample)	Accounts Manager
\$31,000.00		
	Scott Konersmann (sample)	Accounts Manager
\$38,000.00		
	Robert Lyon (sample)	Senior Technician
\$78,000.00		
	Paul Cannon (sample)	Ski Instructor
\$68,500.00		
	Rene Valdes (sample)	Data Analyst III
\$86,000.00		
	Jim Glynn (sample)	Senior International Sales Manager
\$81,400.00		

Expandir varias propiedades de navegación

Puede expandir tantas propiedades de navegación como la consulta requiera. Sin embargo, la opción `$expand` solo puede ir a un nivel de profundidad.

Este ejemplo expande las propiedades de navegación `primarycontactid`, `contact_customer_accounts` y `Account_Tasks` del tipo de entidad de cuenta. La consulta devuelve una respuesta que contiene información sobre la cuenta y dos colecciones: una colección de contactos y una colección de tareas. El código de ejemplo procesa las colecciones de forma independiente.

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/accounts(7d716234-9628-ed11-9db1-000d3a320482)?  
$select=name&$expand=primarycontactid($select=fullname,jobtitle,annualincome),contact_customer_accounts($select=fullname,jobtitle,annualincome),Account_Tasks($select=subject,description) HTTP/1.1  
Prefer: odata.include-annotations="*"  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK  
ETag: W/"1146751"  
OData-Version: 4.0  
Preference-Applied: odata.include-annotations="*"  
  
{  
    "@odata.context": "[Organization  
Uri]/api/data/v9.2/$metadata#accounts(name,primarycontactid(fullname,jobtitl  
e,annualincome),contact_customer_accounts(fullname,jobtitle,annualincome),Ac  
count_Tasks(subject,description))/$entity",  
    "@odata.etag": "W/\"1146751\"",  
    "name": "Contoso, Ltd. (sample)",  
    "accountid": "7d716234-9628-ed11-9db1-000d3a320482",  
    "primarycontactid": {  
        "@odata.etag": "W/\"1146626\"",  
        "fullname": "Yvonne McKay (sample)",  
        "jobtitle": "Coffee Master",  
        "annualincome@OData.Community.Display.V1.FormattedValue": "$45,000.00",  
        "annualincome": 45000.0,  
  
        "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
        "US Dollar",  
  
        "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
        "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
```

```
"transactioncurrency",
  "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
  "contactid": "81716234-9628-ed11-9db1-000d3a320482"
},
"contact_customer_accounts": [
{
  "@odata.etag": "W/\"1146759\"",
  "fullname": "Susanna Stubberod (sample)",
  "jobtitle": "Senior Purchaser",
  "annualincome@OData.Community.Display.V1.FormattedValue":
"$52,000.00",
  "annualincome": 52000.0,
  "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
"US Dollar",

  "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro
perty": "transactioncurrencyid",
  "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
"transactioncurrency",
  "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-
00224802b2ac",
  "contactid": "85716234-9628-ed11-9db1-000d3a320482"
},
{
  "@odata.etag": "W/\"1146761\"",
  "fullname": "Nancy Anderson (sample)",
  "jobtitle": "Activities Manager",
  "annualincome@OData.Community.Display.V1.FormattedValue":
"$55,500.00",
  "annualincome": 55500.0,
  "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
"US Dollar",

  "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro
perty": "transactioncurrencyid",
  "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
"transactioncurrency",
  "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-
00224802b2ac",
  "contactid": "89716234-9628-ed11-9db1-000d3a320482"
},
{
  "@odata.etag": "W/\"1146763\"",
  "fullname": "Maria Cambell (sample)",
  "jobtitle": "Accounts Manager",
  "annualincome@OData.Community.Display.V1.FormattedValue":
"$31,000.00",
  "annualincome": 31000.0,
  "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
"US Dollar",
```

```
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",

"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
        "contactid": "8d716234-9628-ed11-9db1-000d3a320482"
    },
{
    "@odata.etag": "W/\"1146765\",
    "fullname": "Scott Konersmann (sample)",
    "jobtitle": "Accounts Manager",
    "annualincome@OData.Community.Display.V1.FormattedValue": "$38,000.00",
    "annualincome": 38000.0,

    "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue": "US Dollar",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
        "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
            "contactid": "91716234-9628-ed11-9db1-000d3a320482"
    },
{
    "@odata.etag": "W/\"1146767\",
    "fullname": "Robert Lyon (sample)",
    "jobtitle": "Senior Technician",
    "annualincome@OData.Community.Display.V1.FormattedValue": "$78,000.00",
    "annualincome": 78000.0,

    "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue": "US Dollar",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
        "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
            "contactid": "95716234-9628-ed11-9db1-000d3a320482"
    },
{
    "@odata.etag": "W/\"1146769\",
    "fullname": "Paul Cannon (sample)",
    "jobtitle": "Ski Instructor",
    "annualincome@OData.Community.Display.V1.FormattedValue":
```

```
"$68,500.00",
    "annualincome": 68500.0,

    "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
    "US Dollar",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
    "transactioncurrency",
        "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
            "contactid": "99716234-9628-ed11-9db1-000d3a320482"
        },
        {
            "@odata.etag": "W/\"1146771\"",
            "fullname": "Rene Valdes (sample)",
            "jobtitle": "Data Analyst III",
            "annualincome@OData.Community.Display.V1.FormattedValue":
            "$86,000.00",
                "annualincome": 86000.0,

    "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
    "US Dollar",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
    "transactioncurrency",
        "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
            "contactid": "9d716234-9628-ed11-9db1-000d3a320482"
        },
        {
            "@odata.etag": "W/\"1146773\"",
            "fullname": "Jim Glynn (sample)",
            "jobtitle": "Senior International Sales Manager",
            "annualincome@OData.Community.Display.V1.FormattedValue":
            "$81,400.00",
                "annualincome": 81400.0,

    "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
    "US Dollar",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",

    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
    "transactioncurrency",
        "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
            "contactid": "a1716234-9628-ed11-9db1-000d3a320482"
        }
```

```

],
"Account_Tasks": [
{
  "@odata.etag": "W/\"1146746\"",
  "subject": "Task 1 for Contoso, Ltd.",
  "description": "Task 1 for Contoso, Ltd. description",
  "activityid": "7e716234-9628-ed11-9db1-000d3a320482"
},
{
  "@odata.etag": "W/\"1146748\"",
  "subject": "Task 2 for Contoso, Ltd.",
  "description": "Task 2 for Contoso, Ltd. description",
  "activityid": "7f716234-9628-ed11-9db1-000d3a320482"
},
{
  "@odata.etag": "W/\"1146750\"",
  "subject": "Task 3 for Contoso, Ltd.",
  "description": "Task 3 for Contoso, Ltd. description",
  "activityid": "80716234-9628-ed11-9db1-000d3a320482"
}
]
}

```

Salida de la consola:

text

```

-- Expanding multiple property types in one request --

Account Contoso, Ltd. (sample) has the following primary contact person:
  Fullname: Yvonne McKay (sample)
  Jobtitle: Coffee Master
  Annualincome: 45000

Account 'Contoso, Ltd. (sample)' has the following contact customers:
  |Full Name          |Job Title
  |Annual Income
  |-----|-----|-----
  |-----|-----|-----|-----|
  |Susanna Stubberod (sample) |Senior Purchaser
  |$52,000.00
  |Nancy Anderson (sample)   |Activities Manager
  |$55,500.00
  |Maria Cambell (sample)    |Accounts Manager
  |$31,000.00
  |Scott Konersmann (sample)|Accounts Manager
  |$38,000.00
  |Robert Lyon (sample)     |Senior Technician
  |$78,000.00
  |Paul Cannon (sample)     |Ski Instructor
  |$68,500.00
  |Rene Valdes (sample)     |Data Analyst III
  |$86,000.00

```

Jim Glynn (sample)	Senior International Sales Manager
\$81,400.00	

Account 'Contoso, Ltd. (sample)' has the following tasks:

- Task 1 for Contoso, Ltd.
- Task 2 for Contoso, Ltd.
- Task 3 for Contoso, Ltd.

Expansión multivel

Con las propiedades de navegación de un solo valor, puede usar `$expand` para continuar subiendo varios niveles usando solo propiedades de navegación de un solo valor. Esta consulta comienza con registros `task` y amplía los datos de las tablas `contact`, `account` y `systemuser` usando solo propiedades de navegación de un solo valor.

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/tasks?
$select=subject&$filter=regardingobjectid_contact_task/_accountid_value eq
7d716234-9628-ed11-9db1-
000d3a320482&$expand=regardingobjectid_contact_task($select=fullname;$expand
=parentcustomerid_account($select=name;$expand=createdby($select=fullname)))
HTTP/1.1
Prefer: odata.include-annotations="*"
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0
Preference-Applied: odata.include-annotations="*"

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#tasks(subject,regardingobjectid_contact_task(fu
llname,parentcustomerid_account(name,createdby(fullname))))",
    "@Microsoft.Dynamics.CRM.totalrecordcount": -1,
    "@Microsoft.Dynamics.CRM.totalrecordcountlimitexceeded": false,
    "@Microsoft.Dynamics.CRM.globalmetadataversion": "1144249",
    "value": [
```

```
{  
    "@odata.etag": "W/\"1146775\"",  
    "subject": "Task 1 for Susanna Stubberod",  
    "activityid": "86716234-9628-ed11-9db1-000d3a320482",  
    "regardingobjectid_contact_task": {  
        "fullname": "Susanna Stubberod (sample)",  
        "contactid": "85716234-9628-ed11-9db1-000d3a320482",  
        "parentcustomerid_account": {  
            "name": "Contoso, Ltd. (sample)",  
            "accountid": "7d716234-9628-ed11-9db1-000d3a320482",  
            "createdby": {  
                "fullname": "FirstName Lastname",  
                "systemuserid": "ce939f72-a724-ed11-b83e-00224804438a",  
                "ownerid": "ce939f72-a724-ed11-b83e-00224804438a"  
            }  
        }  
    }  
},  
{  
    "@odata.etag": "W/\"1146777\"",  
    "subject": "Task 2 for Susanna Stubberod",  
    "activityid": "87716234-9628-ed11-9db1-000d3a320482",  
    "regardingobjectid_contact_task": {  
        "fullname": "Susanna Stubberod (sample)",  
        "contactid": "85716234-9628-ed11-9db1-000d3a320482",  
        "parentcustomerid_account": {  
            "name": "Contoso, Ltd. (sample)",  
            "accountid": "7d716234-9628-ed11-9db1-000d3a320482",  
            "createdby": {  
                "fullname": "FirstName Lastname",  
                "systemuserid": "ce939f72-a724-ed11-b83e-00224804438a",  
                "ownerid": "ce939f72-a724-ed11-b83e-00224804438a"  
            }  
        }  
    }  
},  
{  
    "@odata.etag": "W/\"1146779\"",  
    "subject": "Task 3 for Susanna Stubberod",  
    "activityid": "88716234-9628-ed11-9db1-000d3a320482",  
    "regardingobjectid_contact_task": {  
        "fullname": "Susanna Stubberod (sample)",  
        "contactid": "85716234-9628-ed11-9db1-000d3a320482",  
        "parentcustomerid_account": {  
            "name": "Contoso, Ltd. (sample)",  
            "accountid": "7d716234-9628-ed11-9db1-000d3a320482",  
            "createdby": {  
                "fullname": "FirstName Lastname",  
                "systemuserid": "ce939f72-a724-ed11-b83e-00224804438a",  
                "ownerid": "ce939f72-a724-ed11-b83e-00224804438a"  
            }  
        }  
    }  
},  
{
```

```
"@odata.etag": "W/\"1146781\"",
"subject": "Task 1 for Nancy Anderson",
"activityid": "8a716234-9628-ed11-9db1-000d3a320482",
"regardingobjectid_contact_task": {
    "fullname": "Nancy Anderson (sample)",
    "contactid": "89716234-9628-ed11-9db1-000d3a320482",
    "parentcustomerid_account": {
        "name": "Contoso, Ltd. (sample)",
        "accountid": "7d716234-9628-ed11-9db1-000d3a320482",
        "createdby": {
            "fullname": "FirstName Lastname",
            "systemuserid": "ce939f72-a724-ed11-b83e-00224804438a",
            "ownerid": "ce939f72-a724-ed11-b83e-00224804438a"
        }
    }
},
{
    "@odata.etag": "W/\"1146783\"",
    "subject": "Task 2 for Nancy Anderson",
    "activityid": "8b716234-9628-ed11-9db1-000d3a320482",
    "regardingobjectid_contact_task": {
        "fullname": "Nancy Anderson (sample)",
        "contactid": "89716234-9628-ed11-9db1-000d3a320482",
        "parentcustomerid_account": {
            "name": "Contoso, Ltd. (sample)",
            "accountid": "7d716234-9628-ed11-9db1-000d3a320482",
            "createdby": {
                "fullname": "FirstName Lastname",
                "systemuserid": "ce939f72-a724-ed11-b83e-00224804438a",
                "ownerid": "ce939f72-a724-ed11-b83e-00224804438a"
            }
        }
    }
},
{
    "@odata.etag": "W/\"1146785\"",
    "subject": "Task 3 for Nancy Anderson",
    "activityid": "8c716234-9628-ed11-9db1-000d3a320482",
    "regardingobjectid_contact_task": {
        "fullname": "Nancy Anderson (sample)",
        "contactid": "89716234-9628-ed11-9db1-000d3a320482",
        "parentcustomerid_account": {
            "name": "Contoso, Ltd. (sample)",
            "accountid": "7d716234-9628-ed11-9db1-000d3a320482",
            "createdby": {
                "fullname": "FirstName Lastname",
                "systemuserid": "ce939f72-a724-ed11-b83e-00224804438a",
                "ownerid": "ce939f72-a724-ed11-b83e-00224804438a"
            }
        }
    }
},
{
    "@odata.etag": "W/\"1146787\"",
```

```
"subject": "Task 1 for Maria Cambell",
"activityid": "8e716234-9628-ed11-9db1-000d3a320482",
"regardingobjectid_contact_task": {
    "fullname": "Maria Cambell (sample)",
    "contactid": "8d716234-9628-ed11-9db1-000d3a320482",
    "parentcustomerid_account": {
        "name": "Contoso, Ltd. (sample)",
        "accountid": "7d716234-9628-ed11-9db1-000d3a320482",
        "createdby": {
            "fullname": "FirstName Lastname",
            "systemuserid": "ce939f72-a724-ed11-b83e-00224804438a",
            "ownerid": "ce939f72-a724-ed11-b83e-00224804438a"
        }
    }
},
{
    "@odata.etag": "W/\"1146789\"",
    "subject": "Task 2 for Maria Cambell",
    "activityid": "8f716234-9628-ed11-9db1-000d3a320482",
    "regardingobjectid_contact_task": {
        "fullname": "Maria Cambell (sample)",
        "contactid": "8d716234-9628-ed11-9db1-000d3a320482",
        "parentcustomerid_account": {
            "name": "Contoso, Ltd. (sample)",
            "accountid": "7d716234-9628-ed11-9db1-000d3a320482",
            "createdby": {
                "fullname": "FirstName Lastname",
                "systemuserid": "ce939f72-a724-ed11-b83e-00224804438a",
                "ownerid": "ce939f72-a724-ed11-b83e-00224804438a"
            }
        }
    }
},
{
    "@odata.etag": "W/\"1146791\"",
    "subject": "Task 3 for Maria Cambell",
    "activityid": "90716234-9628-ed11-9db1-000d3a320482",
    "regardingobjectid_contact_task": {
        "fullname": "Maria Cambell (sample)",
        "contactid": "8d716234-9628-ed11-9db1-000d3a320482",
        "parentcustomerid_account": {
            "name": "Contoso, Ltd. (sample)",
            "accountid": "7d716234-9628-ed11-9db1-000d3a320482",
            "createdby": {
                "fullname": "FirstName Lastname",
                "systemuserid": "ce939f72-a724-ed11-b83e-00224804438a",
                "ownerid": "ce939f72-a724-ed11-b83e-00224804438a"
            }
        }
    }
},
{
    "@odata.etag": "W/\"1146793\"",
    "subject": "Task 1 for Scott Konersmann",
```

```
"activityid": "92716234-9628-ed11-9db1-000d3a320482",
"regardingobjectid_contact_task": {
    "fullname": "Scott Konersmann (sample)",
    "contactid": "91716234-9628-ed11-9db1-000d3a320482",
    "parentcustomerid_account": {
        "name": "Contoso, Ltd. (sample)",
        "accountid": "7d716234-9628-ed11-9db1-000d3a320482",
        "createdby": {
            "fullname": "FirstName Lastname",
            "systemuserid": "ce939f72-a724-ed11-b83e-00224804438a",
            "ownerid": "ce939f72-a724-ed11-b83e-00224804438a"
        }
    }
},
{
    "@odata.etag": "W/\"1146795\",
    "subject": "Task 2 for Scott Konersmann",
    "activityid": "93716234-9628-ed11-9db1-000d3a320482",
    "regardingobjectid_contact_task": {
        "fullname": "Scott Konersmann (sample)",
        "contactid": "91716234-9628-ed11-9db1-000d3a320482",
        "parentcustomerid_account": {
            "name": "Contoso, Ltd. (sample)",
            "accountid": "7d716234-9628-ed11-9db1-000d3a320482",
            "createdby": {
                "fullname": "FirstName Lastname",
                "systemuserid": "ce939f72-a724-ed11-b83e-00224804438a",
                "ownerid": "ce939f72-a724-ed11-b83e-00224804438a"
            }
        }
    }
},
{
    "@odata.etag": "W/\"1146797\",
    "subject": "Task 3 for Scott Konersmann",
    "activityid": "94716234-9628-ed11-9db1-000d3a320482",
    "regardingobjectid_contact_task": {
        "fullname": "Scott Konersmann (sample)",
        "contactid": "91716234-9628-ed11-9db1-000d3a320482",
        "parentcustomerid_account": {
            "name": "Contoso, Ltd. (sample)",
            "accountid": "7d716234-9628-ed11-9db1-000d3a320482",
            "createdby": {
                "fullname": "FirstName Lastname",
                "systemuserid": "ce939f72-a724-ed11-b83e-00224804438a",
                "ownerid": "ce939f72-a724-ed11-b83e-00224804438a"
            }
        }
    }
},
{
    "@odata.etag": "W/\"1146799\",
    "subject": "Task 1 for Robert Lyon",
    "activityid": "96716234-9628-ed11-9db1-000d3a320482",
```

```
"regardingobjectid_contact_task": {
    "fullname": "Robert Lyon (sample)",
    "contactid": "95716234-9628-ed11-9db1-000d3a320482",
    "parentcustomerid_account": {
        "name": "Contoso, Ltd. (sample)",
        "accountid": "7d716234-9628-ed11-9db1-000d3a320482",
        "createdby": {
            "fullname": "FirstName Lastname",
            "systemuserid": "ce939f72-a724-ed11-b83e-00224804438a",
            "ownerid": "ce939f72-a724-ed11-b83e-00224804438a"
        }
    }
},
{
    "@odata.etag": "W/\"1146801\"",
    "subject": "Task 2 for Robert Lyon",
    "activityid": "97716234-9628-ed11-9db1-000d3a320482",
    "regardingobjectid_contact_task": {
        "fullname": "Robert Lyon (sample)",
        "contactid": "95716234-9628-ed11-9db1-000d3a320482",
        "parentcustomerid_account": {
            "name": "Contoso, Ltd. (sample)",
            "accountid": "7d716234-9628-ed11-9db1-000d3a320482",
            "createdby": {
                "fullname": "FirstName Lastname",
                "systemuserid": "ce939f72-a724-ed11-b83e-00224804438a",
                "ownerid": "ce939f72-a724-ed11-b83e-00224804438a"
            }
        }
    }
},
{
    "@odata.etag": "W/\"1146803\"",
    "subject": "Task 3 for Robert Lyon",
    "activityid": "98716234-9628-ed11-9db1-000d3a320482",
    "regardingobjectid_contact_task": {
        "fullname": "Robert Lyon (sample)",
        "contactid": "95716234-9628-ed11-9db1-000d3a320482",
        "parentcustomerid_account": {
            "name": "Contoso, Ltd. (sample)",
            "accountid": "7d716234-9628-ed11-9db1-000d3a320482",
            "createdby": {
                "fullname": "FirstName Lastname",
                "systemuserid": "ce939f72-a724-ed11-b83e-00224804438a",
                "ownerid": "ce939f72-a724-ed11-b83e-00224804438a"
            }
        }
    }
},
{
    "@odata.etag": "W/\"1146806\"",
    "subject": "Task 1 for Paul Cannon",
    "activityid": "9a716234-9628-ed11-9db1-000d3a320482",
    "regardingobjectid_contact_task": {
```

```
        "fullname": "Paul Cannon (sample)",
        "contactid": "99716234-9628-ed11-9db1-000d3a320482",
        "parentcustomerid_account": {
            "name": "Contoso, Ltd. (sample)",
            "accountid": "7d716234-9628-ed11-9db1-000d3a320482",
            "createdby": {
                "fullname": "FirstName Lastname",
                "systemuserid": "ce939f72-a724-ed11-b83e-00224804438a",
                "ownerid": "ce939f72-a724-ed11-b83e-00224804438a"
            }
        }
    },
    {
        "@odata.etag": "W/\"1146808\"",
        "subject": "Task 2 for Paul Cannon",
        "activityid": "9b716234-9628-ed11-9db1-000d3a320482",
        "regardingobjectid_contact_task": {
            "fullname": "Paul Cannon (sample)",
            "contactid": "99716234-9628-ed11-9db1-000d3a320482",
            "parentcustomerid_account": {
                "name": "Contoso, Ltd. (sample)",
                "accountid": "7d716234-9628-ed11-9db1-000d3a320482",
                "createdby": {
                    "fullname": "FirstName Lastname",
                    "systemuserid": "ce939f72-a724-ed11-b83e-00224804438a",
                    "ownerid": "ce939f72-a724-ed11-b83e-00224804438a"
                }
            }
        }
    },
    {
        "@odata.etag": "W/\"1146810\"",
        "subject": "Task 3 for Paul Cannon",
        "activityid": "9c716234-9628-ed11-9db1-000d3a320482",
        "regardingobjectid_contact_task": {
            "fullname": "Paul Cannon (sample)",
            "contactid": "99716234-9628-ed11-9db1-000d3a320482",
            "parentcustomerid_account": {
                "name": "Contoso, Ltd. (sample)",
                "accountid": "7d716234-9628-ed11-9db1-000d3a320482",
                "createdby": {
                    "fullname": "FirstName Lastname",
                    "systemuserid": "ce939f72-a724-ed11-b83e-00224804438a",
                    "ownerid": "ce939f72-a724-ed11-b83e-00224804438a"
                }
            }
        }
    },
    {
        "@odata.etag": "W/\"1146812\"",
        "subject": "Task 1 for Rene Valdes",
        "activityid": "9e716234-9628-ed11-9db1-000d3a320482",
        "regardingobjectid_contact_task": {
            "fullname": "Rene Valdes (sample)",
```

```
    "contactid": "9d716234-9628-ed11-9db1-000d3a320482",
    "parentcustomerid_account": {
        "name": "Contoso, Ltd. (sample)",
        "accountid": "7d716234-9628-ed11-9db1-000d3a320482",
        "createdby": {
            "fullname": "FirstName Lastname",
            "systemuserid": "ce939f72-a724-ed11-b83e-00224804438a",
            "ownerid": "ce939f72-a724-ed11-b83e-00224804438a"
        }
    }
},
{
    "@odata.etag": "W/\"1146814\"",
    "subject": "Task 2 for Rene Valdes",
    "activityid": "9f716234-9628-ed11-9db1-000d3a320482",
    "regardingobjectid_contact_task": {
        "fullname": "Rene Valdes (sample)",
        "contactid": "9d716234-9628-ed11-9db1-000d3a320482",
        "parentcustomerid_account": {
            "name": "Contoso, Ltd. (sample)",
            "accountid": "7d716234-9628-ed11-9db1-000d3a320482",
            "createdby": {
                "fullname": "FirstName Lastname",
                "systemuserid": "ce939f72-a724-ed11-b83e-00224804438a",
                "ownerid": "ce939f72-a724-ed11-b83e-00224804438a"
            }
        }
    }
},
{
    "@odata.etag": "W/\"1146816\"",
    "subject": "Task 3 for Rene Valdes",
    "activityid": "a0716234-9628-ed11-9db1-000d3a320482",
    "regardingobjectid_contact_task": {
        "fullname": "Rene Valdes (sample)",
        "contactid": "9d716234-9628-ed11-9db1-000d3a320482",
        "parentcustomerid_account": {
            "name": "Contoso, Ltd. (sample)",
            "accountid": "7d716234-9628-ed11-9db1-000d3a320482",
            "createdby": {
                "fullname": "FirstName Lastname",
                "systemuserid": "ce939f72-a724-ed11-b83e-00224804438a",
                "ownerid": "ce939f72-a724-ed11-b83e-00224804438a"
            }
        }
    }
},
{
    "@odata.etag": "W/\"1146820\"",
    "subject": "Task 1 for Jim Glynn",
    "activityid": "a2716234-9628-ed11-9db1-000d3a320482",
    "regardingobjectid_contact_task": {
        "fullname": "Jim Glynn (sample)",
        "contactid": "a1716234-9628-ed11-9db1-000d3a320482",
        "parentcustomerid_account": {
            "name": "Contoso, Ltd. (sample)",
            "accountid": "7d716234-9628-ed11-9db1-000d3a320482",
            "createdby": {
                "fullname": "FirstName Lastname",
                "systemuserid": "ce939f72-a724-ed11-b83e-00224804438a",
                "ownerid": "ce939f72-a724-ed11-b83e-00224804438a"
            }
        }
    }
}
```

```

    "parentcustomerid_account": {
        "name": "Contoso, Ltd. (sample)",
        "accountid": "7d716234-9628-ed11-9db1-000d3a320482",
        "createdby": {
            "fullname": "FirstName Lastname",
            "systemuserid": "ce939f72-a724-ed11-b83e-00224804438a",
            "ownerid": "ce939f72-a724-ed11-b83e-00224804438a"
        }
    }
},
{
    "@odata.etag": "W/\"1146822\",
    "subject": "Task 2 for Jim Glynn",
    "activityid": "a3716234-9628-ed11-9db1-000d3a320482",
    "regardingobjectid_contact_task": {
        "fullname": "Jim Glynn (sample)",
        "contactid": "a1716234-9628-ed11-9db1-000d3a320482",
        "parentcustomerid_account": {
            "name": "Contoso, Ltd. (sample)",
            "accountid": "7d716234-9628-ed11-9db1-000d3a320482",
            "createdby": {
                "fullname": "FirstName Lastname",
                "systemuserid": "ce939f72-a724-ed11-b83e-00224804438a",
                "ownerid": "ce939f72-a724-ed11-b83e-00224804438a"
            }
        }
    }
},
{
    "@odata.etag": "W/\"1146827\",
    "subject": "Task 3 for Jim Glynn",
    "activityid": "a4716234-9628-ed11-9db1-000d3a320482",
    "regardingobjectid_contact_task": {
        "fullname": "Jim Glynn (sample)",
        "contactid": "a1716234-9628-ed11-9db1-000d3a320482",
        "parentcustomerid_account": {
            "name": "Contoso, Ltd. (sample)",
            "accountid": "7d716234-9628-ed11-9db1-000d3a320482",
            "createdby": {
                "fullname": "FirstName Lastname",
                "systemuserid": "ce939f72-a724-ed11-b83e-00224804438a",
                "ownerid": "ce939f72-a724-ed11-b83e-00224804438a"
            }
        }
    }
},
]
}

```

Salida de la consola:

Expanded values from Task:

Subject	Contact
Account	Account CreatedBy
-----	----- -----
Task 1 for Susanna Stubberod	Susanna Stubberod (sample)
Contoso, Ltd. (sample)	FirstName Lastname
Task 2 for Susanna Stubberod	Susanna Stubberod (sample)
Contoso, Ltd. (sample)	FirstName Lastname
Task 3 for Susanna Stubberod	Susanna Stubberod (sample)
Contoso, Ltd. (sample)	FirstName Lastname
Task 1 for Nancy Anderson	Nancy Anderson (sample)
Contoso, Ltd. (sample)	FirstName Lastname
Task 2 for Nancy Anderson	Nancy Anderson (sample)
Contoso, Ltd. (sample)	FirstName Lastname
Task 3 for Nancy Anderson	Nancy Anderson (sample)
Contoso, Ltd. (sample)	FirstName Lastname
Task 1 for Maria Cambell	Maria Cambell (sample)
Contoso, Ltd. (sample)	FirstName Lastname
Task 2 for Maria Cambell	Maria Cambell (sample)
Contoso, Ltd. (sample)	FirstName Lastname
Task 3 for Maria Cambell	Maria Cambell (sample)
Contoso, Ltd. (sample)	FirstName Lastname
Task 1 for Scott Konersmann	Scott Konersmann (sample)
Contoso, Ltd. (sample)	FirstName Lastname
Task 2 for Scott Konersmann	Scott Konersmann (sample)
Contoso, Ltd. (sample)	FirstName Lastname
Task 3 for Scott Konersmann	Scott Konersmann (sample)
Contoso, Ltd. (sample)	FirstName Lastname
Task 1 for Robert Lyon	Robert Lyon (sample)
Contoso, Ltd. (sample)	FirstName Lastname
Task 2 for Robert Lyon	Robert Lyon (sample)
Contoso, Ltd. (sample)	FirstName Lastname
Task 3 for Robert Lyon	Robert Lyon (sample)
Contoso, Ltd. (sample)	FirstName Lastname
Task 1 for Paul Cannon	Paul Cannon (sample)
Contoso, Ltd. (sample)	FirstName Lastname
Task 2 for Paul Cannon	Paul Cannon (sample)
Contoso, Ltd. (sample)	FirstName Lastname
Task 3 for Paul Cannon	Paul Cannon (sample)
Contoso, Ltd. (sample)	FirstName Lastname
Task 1 for Rene Valdes	Rene Valdes (sample)
Contoso, Ltd. (sample)	FirstName Lastname
Task 2 for Rene Valdes	Rene Valdes (sample)
Contoso, Ltd. (sample)	FirstName Lastname
Task 3 for Rene Valdes	Rene Valdes (sample)
Contoso, Ltd. (sample)	FirstName Lastname
Task 1 for Jim Glynn	Jim Glynn (sample)
Contoso, Ltd. (sample)	FirstName Lastname
Task 2 for Jim Glynn	Jim Glynn (sample)
Contoso, Ltd. (sample)	FirstName Lastname
Task 3 for Jim Glynn	Jim Glynn (sample)
Contoso, Ltd. (sample)	FirstName Lastname

Sección 7: Agregación de resultados

Para devolver valores agregados, utilice la opción `$apply`. Este ejemplo aplica `average`, `sum`, `min` y `max` a los valores de ingresos anuales.

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/accounts(7d716234-9628-ed11-9db1-000d3a320482)/contact_customer_accounts?$apply=aggregate(annualincome with average as average, annualincome with sum as total, annualincome with min as minimum, annualincome with max as maximum) HTTP/1.1
Prefer: odata.include-annotations="*"
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0
Preference-Applied: odata.include-annotations="*"

{
    "@odata.context": "[Organization Uri]/api/data/v9.2/$metadata#contacts",
    "@Microsoft.Dynamics.CRM.totalrecordcount": -1,
    "@Microsoft.Dynamics.CRM.totalrecordcountlimitexceeded": false,
    "value": [
        {
            "telephone3": null,
            "address1_shippingmethodcode": null,
            "familystatuscode": null,
            "haschildrencode": null,
            "nickname": null,
            "address1_freighttermscode": null,
            "address3_upszone": null,
            "_ownerid_value": null,
            "annualincome_base": null,
            "anniversary": null,
            "address1_upszone": null,
            "fullname": null,
            "merged": null,
            "websiteurl": null,
            "address2_city": null,
            "_slainvokedid_value": null,
            "address1_postofficebox": null,
            "importsequencenumber": null,
```

```
"address3_longitude": null,
"preferredappointmentdaycode": null,
"customertypecode": null,
"utcconversionontimezonecode": null,
"overriddencreatedon": null,
"aging90": null,
"stageid": null,
"address3_primarycontactname": null,
"address1_utcoffset": null,
"address1_latitude": null,
"home2": null,
"yomifirstname": null,
"isbackofficecustomer": null,
"_masterid_value": null,
"address3_shippingmethodcode": null,
"lastonholdtime": null,
"address2_fax": null,
"address3_stateorprovince": null,
"address3_telephone3": null,
"address3_telephone2": null,
"address3_telephone1": null,
"_transactioncurrencyid_value": null,
"governmentid": null,
"yomifullname": null,
"participatesinworkflow": null,
"address2_line1": null,
"followemail": null,
"address1_telephone3": null,
"educationcode": null,
"address1_telephone2": null,
"address1_telephone1": null,
"address2_postofficebox": null,
"_owninguser_value": null,
"emailaddress1": null,
"ftpsiteurl": null,
"emailaddress2": null,
"address2_latitude": null,
"processid": null,
"emailaddress3": null,
"address2_shippingmethodcode": null,
"address2_composite": null,
"creditonhold": null,
"traversedpath": null,
"address1_city": null,
"spousesname": null,
"address3_addressid": null,
"address3_name": null,
"address3_postofficebox": null,
"address2_line2": null,
"aging30_base": null,
"address1_addressid": null,
"address1_addressstypecode": null,
"donotphone": null,
"managerphone": null,
"contactid": null,
```

```
"address2_stateorprovince": null,
"_createdby_value": null,
"donotemail": null,
"address2_postalcode": null,
"donotsendmm": null,
"entityimage_url": null,
"firstname": null,
"address1_composite": null,
"leadsourcecode": null,
"aging60": null,
"managername": null,
"_modifiedby_value": null,
"address3_postalcode": null,
"marketingonly": null,
"jobtitle": null,
"timezoneruleversionnumber": null,
"address3_utcoffset": null,
"address2_telephone3": null,
"address2_telephone2": null,
"address2_telephone1": null,
"numberofchildren": null,
"address1_postalcode": null,
"address2_upszone": null,
"_owningteam_value": null,
"address2_line3": null,
"timespentbymeonemailandmeetings": null,
"territorycode": null,
"department": null,
"address1_country": null,
"address2_longitude": null,
"suffix": null,
"_modifiedonbehalfby_value": null,
"creditlimit": null,
"address1_line2": null,
"paymenttermscode": null,
"address1_county": null,
"donotpostalmail": null,
"_preferredsystemuserid_value": null,
"accountrolecode": null,
"preferredappointmenttimecode": null,
"assistantname": null,
"address1_fax": null,
"_owningbusinessunit_value": null,
"_parentcustomerid_value": null,
"_createdonbehalfby_value": null,
"annualincome": null,
"_accountid_value": null,
"modifiedon": null,
"address2_name": null,
"creditlimit_base": null,
"_modifiedbyexternalparty_value": null,
"address2_utcoffset": null,
"business2": null,
"statuscode": null,
"address3_composite": null,
```

```
"_slaid_value": null,
"fax": null,
"address1_line1": null,
"shippingmethodcode": null,
"donotbulkemail": null,
"childrensnames": null,
"address2_county": null,
"lastname": null,
"versionnumber": null,
"address3_city": null,
"address2_freighttermscode": null,
"aging30": null,
"externaluseridentifier": null,
"address1_line3": null,
"_parentcontactid_value": null,
"assistantphone": null,
"statecode": null,
"address1_stateorprovince": null,
"birthdate": null,
"customersizecode": null,
"address3_addressstypecode": null,
"onholdtime": null,
"_createdbyexternalparty_value": null,
"entityimage_timestamp": null,
"mobilephone": null,
"address3_county": null,
"employeeid": null,
"exchangerate": null,
"subscriptionid": null,
"entityimageid": null,
"company": null,
"donotbulkpostalmail": null,
"gendercode": null,
"callback": null,
"lastusedincampaign": null,
"address3_line3": null,
"donotfax": null,
"telephone2": null,
"address3_freighttermscode": null,
"yomilastname": null,
"address3_fax": null,
"description": null,
"address3_line1": null,
"address3_line2": null,
"address2_addressstypecode": null,
"createdon": null,
"yomimiddlename": null,
"aging90_base": null,
"address1_name": null,
"telephone1": null,
"address1_primarycontactname": null,
"address1_longitude": null,
"middlename": null,
"address2_primarycontactname": null,
"entityimage": null,
```

```

    "address2_addressid": null,
    "preferredcontactmethodcode": null,
    "address3_latitude": null,
    "salutation": null,
    "aging60_base": null,
    "pager": null,
    "address2_country": null,
    "address3_country": null,
    "average@odata.Community.Display.V1.AttributeName": "annualincome",
    "average@odata.Community.Display.V1.FormattedValue": "$61,300.00",
    "average": 61300.0,
    "total@odata.Community.Display.V1.AttributeName": "annualincome",
    "total@odata.Community.Display.V1.FormattedValue": "$490,400.00",
    "total": 490400.0,
    "minimum@odata.Community.Display.V1.AttributeName": "annualincome",
    "minimum@odata.Community.Display.V1.FormattedValue": "$31,000.00",
    "minimum": 31000.0,
    "maximum@odata.Community.Display.V1.AttributeName": "annualincome",
    "maximum@odata.Community.Display.V1.FormattedValue": "$86,000.00",
    "maximum": 86000.0
}
]
}

```

Salida de la consola:

```

Aggregated Annual Income information for Contoso contacts:
Average income: $61,300.00
Total income: $490,400.00
Minium income: $31,000.00
Maximum income: $86,000.00

```

Sección 8: Consultas FetchXML

Además de las operaciones de filtrado de consultas, la API web también admite [consultas FetchXML](#). FetchXML proporciona una forma alternativa de definir consultas y más opciones para la agregación. Todas las opciones de consulta que definiríamos normalmente, como `$select`, `$filter` y `$orderby`, ahora se definen en el XML.

Para usar FetchXML, redacte una cadena que represente la consulta. Asegúrese de que la cadena de consulta se ajuste al [esquema FetchXML](#). Antes de incluir la cadena en la URL, debe codificarla como URL.

En este ejemplo, consultamos todos los contactos donde `fullname` coincide con `(sample)` y devuelve los resultados en orden descendente por `fullname`.

Este es el XML para la consulta:

XML

```
<fetch mapping="logical" output-format="xml-platform" version="1.0"
distinct="false">
  <entity name="contact">
    <attribute name="fullname" />
    <attribute name="jobtitle" />
    <attribute name="annualincome" />
    <order descending="true"
      attribute="fullname" />
    <filter type="and">
      <condition value="%(sample)%" attribute="fullname" operator="like" />
      <condition value="7d716234-9628-ed11-9db1-000d3a320482" attribute="parentcustomerid" operator="eq" />
    </filter>
  </entity>
</fetch>
```

En este ejemplo, la consulta se envía mediante una solicitud `$batch`. Las consultas de FetchXML pueden ser largas, por lo que pueden alcanzar los límites de URL que se pueden enviar mediante `GET`. Con una operación `$batch`, la URL se envía en el cuerpo de la solicitud, donde los límites son más altos.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/$batch HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

--batch_7118cb08-27c1-44c0-be91-c2442fe9d454
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-Length: 881

GET /api/data/v9.2/contacts?
fetchXml=%3Cfetch+mapping%3D%22logical%22+output-format%3D%22xml-
platform%22+version%3D%221.0%22+distinct%3D%22false%22%3E%0D%0A++%3Centity+n
ame%3D%22contact%22%3E%0D%0A++++%3Cattribute+name%3D%22fullname%22+%2F%3E%0D
%0A++++%3Cattribute+name%3D%22jobtitle%22+%2F%3E%0D%0A++++%3Cattribute+name%3D
%22annualincome%22+%2F%3E%0D%0A++++%3Corder+descending%3D%22true%22+attrib
ute%3D%22fullname%22+%2F%3E%0D%0A++++%3Cfilter+type%3D%22and%22%3E%0D%0A++++
++%3Ccondition+value%3D%22%25(sample)%25%22+attribute%3D%22fullname%22+opera
```

```
tor%3D%22like%22+%2F%3E%0D%0A+++++%3Ccondition+value%3D%227d716234-9628-
ed11-9db1-
000d3a320482%22+attribute%3D%22parentcustomerid%22+operator%3D%22eq%22+%2F%3
E%0D%0A++++%3C%2Ffilter%3E%0D%0A++%3C%2Fentity%3E%0D%0A%3C%2Ffetch%3E&$count
=true HTTP/1.1
Prefer: odata.include-annotations="*"
```

```
--batch_7118cb08-27c1-44c0-be91-c2442fe9d454--
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

--batchresponse_ed1a482b-c942-4da4-8257-adaa53acc8e0
Content-Type: application/http
Content-Transfer-Encoding: binary

HTTP/1.1 200 OK
Content-Type: application/json; odata.metadata=minimal; odata.streaming=true
OData-Version: 4.0
Preference-Applied: odata.include-annotations="*"

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#contacts(fullname,jobtitle,annualincome,_transa
ctioncurrencyid_value,transactioncurrencyid,contactid,transactioncurrencyid(
))",
    "@odata.count": 8,
    "@Microsoft.Dynamics.CRM.totalrecordcount": 8,
    "@Microsoft.Dynamics.CRM.totalrecordcountlimitexceeded": false,
    "@Microsoft.Dynamics.CRM.globalmetadataversion": "1144249",
    "value": [
        {
            "@odata.etag": "W/\"1146759\"",
            "fullname": "Susanna Stubberod (sample)",
            "jobtitle": "Senior Purchaser",
            "annualincome@OData.Community.Display.V1.FormattedValue":
"$52,000.00",
            "annualincome": 52000.0,
            "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
"US Dollar",
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro
perty": "transactioncurrencyid",
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
"transactioncurrency",
```

```
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
        "contactid": "85716234-9628-ed11-9db1-000d3a320482"
    },
    {
        "@odata.etag": "W/\"1146765\",
        "fullname": "Scott Konersmann (sample)",
        "jobtitle": "Accounts Manager",
        "annualincome@OData.Community.Display.V1.FormattedValue": "$38,000.00",
        "annualincome": 38000.0,

        "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue": "US Dollar",

        "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",

        "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
            "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
                "contactid": "91716234-9628-ed11-9db1-000d3a320482"
            },
            {
                "@odata.etag": "W/\"1146767\",
                "fullname": "Robert Lyon (sample)",
                "jobtitle": "Senior Technician",
                "annualincome@OData.Community.Display.V1.FormattedValue": "$78,000.00",
                "annualincome": 78000.0,

                "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue": "US Dollar",

                "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",

                "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
                    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
                        "contactid": "95716234-9628-ed11-9db1-000d3a320482"
                    },
                    {
                        "@odata.etag": "W/\"1146771\",
                        "fullname": "Rene Valdes (sample)",
                        "jobtitle": "Data Analyst III",
                        "annualincome@OData.Community.Display.V1.FormattedValue": "$86,000.00",
                        "annualincome": 86000.0,

                        "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue": "US Dollar",
```

```
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",

"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
        "contactid": "9d716234-9628-ed11-9db1-000d3a320482"
    },
{
    "@odata.etag": "W/\"1146769\"",
    "fullname": "Paul Cannon (sample)",
    "jobtitle": "Ski Instructor",
    "annualincome@OData.Community.Display.V1.FormattedValue": "$68,500.00",
    "annualincome": 68500.0,

"_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue": "US Dollar",

"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",

"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
        "contactid": "99716234-9628-ed11-9db1-000d3a320482"
    },
{
    "@odata.etag": "W/\"1146761\"",
    "fullname": "Nancy Anderson (sample)",
    "jobtitle": "Activities Manager",
    "annualincome@OData.Community.Display.V1.FormattedValue": "$55,500.00",
    "annualincome": 55500.0,

"_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue": "US Dollar",

"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",

"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac",
        "contactid": "89716234-9628-ed11-9db1-000d3a320482"
    },
{
    "@odata.etag": "W/\"1146763\"",
    "fullname": "Maria Cambell (sample)",
    "jobtitle": "Accounts Manager",
    "annualincome@OData.Community.Display.V1.FormattedValue": "$31,000.00",
```

```

    "annualincome": 31000.0,
    "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
    "US Dollar",
    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro
    perty": "transactioncurrencyid",
    "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
    "transactioncurrency",
        "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-
    00224802b2ac",
        "contactid": "8d716234-9628-ed11-9db1-000d3a320482"
    },
    {
        "@odata.etag": "W/\"1146773\"",
        "fullname": "Jim Glynn (sample)",
        "jobtitle": "Senior International Sales Manager",
        "annualincome@OData.Community.Display.V1.FormattedValue":
        "$81,400.00",
        "annualincome": 81400.0,
        "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
        "US Dollar",
        "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro
        perty": "transactioncurrencyid",
        "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
        "transactioncurrency",
            "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-
        00224802b2ac",
            "contactid": "a1716234-9628-ed11-9db1-000d3a320482"
        }
    ]
}
--batchresponse_ed1a482b-c942-4da4-8257-adaa53acc8e0--

```

Salida de la consola:

Contacts Fetched by fullname containing '(sample)':	
Full Name	Job Title
Annual Income	----- ----- -----
----- ----- ----- -----	
Susanna Stubberod (sample) Senior Purchaser	
\$52,000.00	
Scott Konersmann (sample) Accounts Manager	
\$38,000.00	
Robert Lyon (sample) Senior Technician	

\$78,000.00		
Rene Valdes (sample)		Data Analyst III
\$86,000.00		
Paul Cannon (sample)		Ski Instructor
\$68,500.00		
Nancy Anderson (sample)		Activities Manager
\$55,500.00		
Maria Cambell (sample)		Accounts Manager
\$31,000.00		
Jim Glynn (sample)		Senior International Sales Manager
\$81,400.00		

Paginación FetchXML

La forma en que FetchXML administra la paginación es diferente de cómo lo hace el filtro de consulta. En FetchXML, puede especificar una columna `count` que indica cuántos resultados devolver por página. En la misma solicitud, usa la columna `page` para especificar el número de página para devolver.

La siguiente operación solicita la página 2 del ejemplo FetchXML anterior. En función de nuestros datos de ejemplo, deberíamos tener ocho contactos en nuestro resultado. Al dividir cada página en solo cuatro contactos por página, deberíamos tener dos páginas. La página 2 debe contener solo cuatro contactos. Si a continuación pedimos la página 3, el sistema devuelve cero resultados.

XML

```

<fetch mapping="logical"
      output-format="xml-platform"
      version="1.0"
      distinct="false"
      page="2"
      count="4">
  <entity name="contact">
    <attribute name="fullname" />
    <attribute name="jobtitle" />
    <attribute name="annualincome" />
    <order descending="true"
          attribute="fullname" />
    <filter type="and">
      <condition value="%(sample)%"
                 attribute="fullname"
                 operator="like" />
      <condition value="7d716234-9628-ed11-9db1-000d3a320482"
                 attribute="parentcustomerid"
                 operator="eq" />
    </filter>
  </entity>
</fetch>
```

```
</entity>
</fetch>
```

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/contacts?
fetchXml=%3Cfetch+mapping%3D%22logical%22+output-format%3D%22xml-
platform%22+version%3D%221.0%22+distinct%3D%22false%22+count%3D%224%22+page%
3D%222%22%3E%0D%0A++%3Centity+name%3D%22contact%22%3E%0D%0A++++%3Cattribute+
name%3D%22fullname%22+%2F%3E%0D%0A++++%3Cattribute+name%3D%22jobtitle%22+%2F
%3E%0D%0A++++%3Cattribute+name%3D%22annualincome%22+%2F%3E%0D%0A++++%3Corder
+descending%3D%22true%22+attribute%3D%22fullname%22+%2F%3E%0D%0A++++%3Cfilter
+type%3D%22and%22%3E%0D%0A++++++%3Ccondition+value%3D%22%25(sample)%25%22+a
ttribute%3D%22fullname%22+operator%3D%22like%22+%2F%3E%0D%0A++++++%3Cconditi
on+value%3D%227d716234-9628-ed11-9db1-
000d3a320482%22+attribute%3D%22parentcustomerid%22+operator%3D%22eq%22+%2F%3
E%0D%0A++++%3C%2Ffilter%3E%0D%0A++%3C%2Fentity%3E%0D%0A%3C%2Ffetch%3E&$count
=true HTTP/1.1
Prefer: odata.include-annotations="*"
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0
Preference-Applied: odata.include-annotations="*"

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#contacts(fullname,jobtitle,annualincome,_transa
ctioncurrencyid_value,transactioncurrencyid,contactid,transactioncurrencyid(
))",
    "@odata.count": 8,
    "@Microsoft.Dynamics.CRM.totalrecordcount": 8,
    "@Microsoft.Dynamics.CRM.totalrecordcountlimitexceeded": false,
    "@Microsoft.Dynamics.CRM.globalmetadataversion": "1144249",
    "value": [
        {
            "@odata.etag": "W/\"1147963\"",
            "fullname": "Paul Cannon (sample)",
            "jobtitle": "Ski Instructor",
            "annualincome@OData.Community.Display.V1.FormattedValue":
"$68,500.00",
            "annualincome": 68500.0,
```

```
"_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
"US Dollar",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
"transactioncurrency",  
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
        "contactid": "99716234-9628-ed11-9db1-000d3a320482"  
    },  
    {  
        "@odata.etag": "W/\"1147945\"",  
        "fullname": "Nancy Anderson (sample)",  
        "jobtitle": "Activities Manager",  
        "annualincome@OData.Community.Display.V1.FormattedValue":  
"$55,500.00",  
        "annualincome": 55500.0,  
  
"_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
"US Dollar",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
"transactioncurrency",  
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
        "contactid": "89716234-9628-ed11-9db1-000d3a320482"  
    },  
    {  
        "@odata.etag": "W/\"1147948\"",  
        "fullname": "Maria Cambell (sample)",  
        "jobtitle": "Accounts Manager",  
        "annualincome@OData.Community.Display.V1.FormattedValue":  
"$31,000.00",  
        "annualincome": 31000.0,  
  
"_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":  
"US Dollar",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro  
perty": "transactioncurrencyid",  
  
"_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":  
"transactioncurrency",  
    "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-  
00224802b2ac",  
        "contactid": "8d716234-9628-ed11-9db1-000d3a320482"  
    },  
    {  
        "@odata.etag": "W/\"1147979\"",  
        "fullname": "Jim Glynn (sample)",
```

```

        "jobtitle": "Senior International Sales Manager",
        "annualincome@OData.Community.Display.V1.FormattedValue":
      "$81,400.00",
        "annualincome": 81400.0,
      "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue":
      "US Dollar",
      "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationpro
      perty": "transactioncurrencyid",
      "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
      "transactioncurrency",
        "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-
      00224802b2ac",
        "contactid": "a1716234-9628-ed11-9db1-000d3a320482"
      }
    ]
}

```

Salida de la consola:

text

Contacts Fetched by fullname containing '(sample)' - Page 2:	
Full Name	Job Title
Annual Income	
----- ----- -----	
Paul Cannon (sample)	Ski Instructor
\$68,500.00	
Nancy Anderson (sample)	Activities Manager
\$55,500.00	
Maria Cambell (sample)	Accounts Manager
\$31,000.00	
Jim Glynn (sample)	Senior International Sales Manager
\$81,400.00	

Sección 9: Usar consultas predefinidas

Puede usar la API web para [recuperar y ejecutar consultas predefinidas](#).

Consulta guardada

En esta operación, solicitamos el GUID `savedqueryid` de la consulta guardada con el nombre **Cuentas activas**. A continuación usamos el GUID y el parámetro `savedQuery` para consultar una lista de cuentas activas.

Primero, obtenemos el GUID de la consulta guardada.

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/savedqueries?  
$select=name,savedqueryid&$filter=name eq 'Active Accounts' HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK  
OData-Version: 4.0  
  
{  
    "@odata.context": "[Organization  
Uri]/api/data/v9.2/$metadata#savedqueries(name,savedqueryid)",  
    "value": [  
        {  
            "@odata.etag": "W/\\"966435\\\"",  
            "name": "Active Accounts",  
            "savedqueryid": "00000000-0000-0000-00aa-000010001002"  
        }  
    ]  
}
```

Luego usamos el parámetro `savedQuery` para obtener contenido de la consulta guardada.

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/accounts?savedQuery=00000000-0000-0000-  
00aa-000010001002 HTTP/1.1  
Prefer: odata.maxpagesize=3; odata.include-annotations="*"  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0
Preference-Applied: odata.include-annotations="*"; odata.maxpagesize=3

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#accounts(name,_primarycontactid_value,primaryco
ntactid,accountid,address1_city,telephone1,primarycontactid())",
    "@Microsoft.Dynamics.CRM.totalrecordcount": -1,
    "@Microsoft.Dynamics.CRM.totalrecordcountlimitexceeded": false,
    "@Microsoft.Dynamics.CRM.globalmetadataversion": "1144249",

"@accountprimarycontactidcontactcontactid.OData.Community.Display.V1.Current
EntityField": "primarycontactid",
    "value": [
        {
            "@odata.etag": "W/\"1147935\"",
            "name": "Contoso, Ltd. (sample)",
            "_primarycontactid_value@OData.Community.Display.V1.FormattedValue":
"Yvonne McKay (sample)",

            "_primarycontactid_value@Microsoft.Dynamics.CRM.associatednavigationproperty
": "primarycontactid",
                "_primarycontactid_value@Microsoft.Dynamics.CRM.lookuplogicalname":
"contact",
                "_primarycontactid_value": "d36e86e2-a228-ed11-9db1-000d3a320482",
                "accountid": "cf6e86e2-a228-ed11-9db1-000d3a320482"
        }
    ]
}
```

Salida de la consola:

```
text
```

```
-- Saved Query --

Active Accounts
    1) Contoso, Ltd. (sample), Yvonne McKay (sample), NULL
```

Consulta de usuario

Este ejemplo crea una consulta de usuario, la ejecuta y luego la elimina del sistema.

Solicitud:

HTTP

```

POST [Organization Uri]/api/data/v9.2/userqueries HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
    "name": "My User Query",
    "description": "User query to display contact info.",
    "querytype": 0,
    "returnedtypecode": "contact",
    "fetchxml": "<fetch mapping='logical' output-format='xml-platform'
version='1.0' distinct='false'>
    <entity name ='contact'>
        <attribute name ='fullname' />
        <attribute name ='contactid' />
        <attribute name ='jobtitle' />
        <attribute name ='annualincome' />
        <order descending ='false' attribute='fullname' />
        <filter type ='and'>
            <condition value ='%(sample)%' attribute='fullname' operator='like' />
            <condition value ='%Manager%' attribute='jobtitle' operator='like' />
            <condition value ='55000' attribute='annualincome' operator='gt' />
        </filter>
    </entity>
</fetch>"
}

```

Respuesta:

HTTP

```

HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization Uri]/api/data/v9.2/userqueries(f76e86e2-a228-
ed11-9db1-000d3a320482)

```

Esta consulta de usuario pide contactos donde `fullname` contiene `(sample)`, `jobtitle` contiene `manager`, y `annualincome` es mayor que `55000`. Nuestros datos de ejemplo tiene dos contactos que coinciden con esta consulta.

En el código de ejemplo, el valor `userqueryid` se devuelve con la solicitud que lo crea. Normalmente, necesitaría consultar el sistema para recuperarlo mediante una consulta como la siguiente:

Solicitud:

HTTP

```
GET https://[Organization URI]/api/data/v9.0/userqueries?  
$select=name,userqueryid,&$filter=name%20eq%20'My%20User%20Query' HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
Content-Type: application/json; charset=utf-8
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK  
Content-Type: application/json; odata.metadata=minimal  
OData-Version: 4.0  
Content-Length: 246  
  
{  
    "@odata.context": "https://[Organization  
URI]/api/data/v9.0/$metadata#userqueries(name,userqueryid)",  
    "value": [  
        {  
            "@odata.etag": "W/\"621698\"",  
            "name": "My User Query",  
            "userqueryid": "f76e86e2-a228-ed11-9db1-000d3a320482"  
        }  
    ]  
}
```

Este ejemplo pasa el valor GUID con el parámetro `userQuery` para obtener el contenido de la consulta del usuario.

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/contacts?userQuery=f76e86e2-a228-ed11-  
9db1-000d3a320482 HTTP/1.1  
Prefer: odata.maxpagesize=3; odata.include-annotations="*"  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json
```

Respuesta:

HTTP

HTTP/1.1 200 OK
OData-Version: 4.0
Preference-Applied: odata.include-annotations="*"; odata.maxpagesize=3

```
{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#contacts(fullname,contactid,jobtitle,annualincome,_transactioncurrencyid_value,transactioncurrencyid,transactioncurrencyid(
))",
    "@Microsoft.Dynamics.CRM.totalrecordcount": -1,
    "@Microsoft.Dynamics.CRM.totalrecordcountlimitexceeded": false,
    "@Microsoft.Dynamics.CRM.globalmetadataversion": "1144249",
    "value": [
        {
            "@odata.etag": "W/\"1147979\"",
            "fullname": "Jim Glynn (sample)",
            "contactid": "f36e86e2-a228-ed11-9db1-000d3a320482",
            "jobtitle": "Senior International Sales Manager",
            "annualincome@OData.Community.Display.V1.FormattedValue": "$81,400.00",
            "annualincome": 81400.0,
            "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue": "US Dollar",
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
            "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac"
        },
        {
            "@odata.etag": "W/\"1147945\"",
            "fullname": "Nancy Anderson (sample)",
            "contactid": "db6e86e2-a228-ed11-9db1-000d3a320482",
            "jobtitle": "Activities Manager",
            "annualincome@OData.Community.Display.V1.FormattedValue": "$55,500.00",
            "annualincome": 55500.0,
            "_transactioncurrencyid_value@OData.Community.Display.V1.FormattedValue": "US Dollar",
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "transactioncurrencyid",
            "_transactioncurrencyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "transactioncurrency",
            "_transactioncurrencyid_value": "daf76074-6820-ed11-b83b-00224802b2ac"
        }
    ]
}
```

Salida de la consola:

```
text

-- User Query --

Contacts Fetched by My User Query:
|Full Name           |Job Title
|Annual Income
|-----|-----|-----
| Jim Glynn (sample) |Senior International Sales Manager
|$81,400.00
| Nancy Anderson (sample) |Activities Manager
|$55,500.00
```

Sección 10: Eliminar registros de muestra

Use una operación `$batch` para eliminar todos los registros creados en la [Sección 0: Crear registros para consultar](#).

Solicitud:

```
HTTP

POST [Organization Uri]/api/data/v9.2/$batch HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

--batch_23ea682f-a60a-412a-b37d-7df10a976508
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-Length: 121

DELETE /api/data/v9.2/contacts(f36e86e2-a228-ed11-9db1-000d3a320482)
HTTP/1.1

--batch_23ea682f-a60a-412a-b37d-7df10a976508
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-Length: 121

DELETE /api/data/v9.2/contacts(ef6e86e2-a228-ed11-9db1-000d3a320482)
HTTP/1.1

--batch_23ea682f-a60a-412a-b37d-7df10a976508
```

Content-Type: application/http
Content-Transfer-Encoding: binary
Content-Length: 121

DELETE /api/data/v9.2/contacts(eb6e86e2-a228-ed11-9db1-000d3a320482)
HTTP/1.1

--batch_23ea682f-a60a-412a-b37d-7df10a976508
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-Length: 121

DELETE /api/data/v9.2/contacts(e76e86e2-a228-ed11-9db1-000d3a320482)
HTTP/1.1

--batch_23ea682f-a60a-412a-b37d-7df10a976508
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-Length: 121

DELETE /api/data/v9.2/contacts(e36e86e2-a228-ed11-9db1-000d3a320482)
HTTP/1.1

--batch_23ea682f-a60a-412a-b37d-7df10a976508
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-Length: 121

DELETE /api/data/v9.2/contacts(df6e86e2-a228-ed11-9db1-000d3a320482)
HTTP/1.1

--batch_23ea682f-a60a-412a-b37d-7df10a976508
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-Length: 121

DELETE /api/data/v9.2/contacts(db6e86e2-a228-ed11-9db1-000d3a320482)
HTTP/1.1

--batch_23ea682f-a60a-412a-b37d-7df10a976508
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-Length: 121

DELETE /api/data/v9.2/contacts(d76e86e2-a228-ed11-9db1-000d3a320482)
HTTP/1.1

--batch_23ea682f-a60a-412a-b37d-7df10a976508
Content-Type: application/http

```
Content-Transfer-Encoding: binary
Content-Length: 121

DELETE /api/data/v9.2/accounts(cf6e86e2-a228-ed11-9db1-000d3a320482)
HTTP/1.1

--batch_23ea682f-a60a-412a-b37d-7df10a976508
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-Length: 121

DELETE /api/data/v9.2/contacts(d36e86e2-a228-ed11-9db1-000d3a320482)
HTTP/1.1

--batch_23ea682f-a60a-412a-b37d-7df10a976508
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-Length: 124

DELETE /api/data/v9.2/userqueries(f76e86e2-a228-ed11-9db1-000d3a320482)
HTTP/1.1

--batch_23ea682f-a60a-412a-b37d-7df10a976508--
```

Respuesta:

```
HTTP

HTTP/1.1 200 OK
OData-Version: 4.0

--batchresponse_1e45a745-8b68-401f-a1c0-0081ec083cdc
Content-Type: application/http
Content-Transfer-Encoding: binary

HTTP/1.1 204 No Content
OData-Version: 4.0

--batchresponse_1e45a745-8b68-401f-a1c0-0081ec083cdc
Content-Type: application/http
Content-Transfer-Encoding: binary

HTTP/1.1 204 No Content
OData-Version: 4.0

--batchresponse_1e45a745-8b68-401f-a1c0-0081ec083cdc
Content-Type: application/http
```



```
Content-Type: application/http
Content-Transfer-Encoding: binary

HTTP/1.1 204 No Content
OData-Version: 4.0


--batchresponse_1e45a745-8b68-401f-a1c0-0081ec083cdc
Content-Type: application/http
Content-Transfer-Encoding: binary

HTTP/1.1 204 No Content
OData-Version: 4.0


--batchresponse_1e45a745-8b68-401f-a1c0-0081ec083cdc--
```

Consulte también

[Usar la API web de Dataverse](#)

[Consultar datos utilizando la API web](#)

[Recuperar y ejecutar consultas predefinidas](#)

[Ejemplo de datos de consulta API \(C#\)](#)

[Ejemplo de datos de consulta de la API web \(JavaScript del lado del cliente\)](#)

Nota

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta. [↗](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad ↗](#)).

Ejemplo de operaciones condicionales de la API web

Artículo • 24/08/2023

Esta colección de muestras demuestra cómo realizar operaciones que se basan condicionalmente en la versión de la fila de la tabla contenida en el servidor de Microsoft Dataverse y/o mantenido actualmente por el cliente. Para obtener más información, consulte [realizar operaciones condicionales mediante la API web](#). Este ejemplo se implementa como proyecto independiente para los siguientes idiomas:

- [Ejemplo de operaciones condicionales de la API web \(C#\)](#)
- [Ejemplo de operaciones condicionales de la API web \(JavaScript del lado del cliente\)](#)

La API web de Dataverse sigue las convenciones del protocolo [OData v4.0](#), que usa [ETags](#) para implementar el control de versiones del recurso. Las operaciones condicionales de la API web dependen de este mecanismo de versiones.

En este tema se explican la estructura y el contenido de los ejemplos en un nivel superior de lenguaje neutro. Detalla las solicitudes y las respuestas HTTP, y la salida del programa asociada, en el caso correspondiente. Revise los temas de ejemplo vinculados anteriores para obtener implementaciones específicas del idioma y detalles relacionados sobre cómo realizar las operaciones descritas en este tema.

Demostraciones

Este ejemplo se divide en tres secciones principales, enumeradas en la tabla siguiente. Cada sección contiene un conjunto de operaciones API Web relacionadas que se analizan con más detalle en la sección conceptual asociada del tema [realizar operaciones condicionales mediante la API Web](#).

Sección de código	Temas conceptuales asociados
Sección 0: Crear registros de muestra	Crear una fila de tabla usando la API web
Sección 1: GET condicional	Recuperaciones condicionales
Sección 2: Simultaneidad optimista al eliminar y actualizar	Aplicar simultaneidad optimista Limitar operaciones de upsert
Sección 3: Eliminar registros de muestra	Eliminación básica Ejecute las operaciones por lotes mediante

Sección de código	Temas conceptuales asociados
	API web

Las siguientes secciones contienen una breve explicación de las operaciones de la API web de Dataverse realizadas, junto con mensajes HTTP correspondientes y la salida asociada de la consola, que es la misma para cada implementación de idioma. Para razones de brevedad, se han omitido los encabezados HTTP menos pertinentes. Los URI de las filas de tabla variarán con la dirección de la organización base y el id. de la fila asignada por su servidor de Dataverse.

Sección 0: Crear registros de muestra

El ejemplo crea la siguiente fila de la tabla antes de que se ejecuten las secciones de código principal.

Solicitud:

```
HTTP

POST [Organization Uri]/api/data/v9.2/accounts?
$select=name,revenue,telephone1,description HTTP/1.1
Prefer: return=representation
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
    "name": "Contoso Ltd",
    "telephone1": "555-0000",
    "revenue": 5000000,
    "description": "Parent company of Contoso Pharmaceuticals, etc."
}
```

Respuesta:

```
HTTP

HTTP/1.1 201 Created
Preference-Applied: return=representation
OData-Version: 4.0

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#accounts(name,revenue,telephone1,description)/$entity",
    "@odata.etag": "W/\"72965013\"",
```

```
"name": "Contoso Ltd",
"revenue": 5000000.0,
"telephone1": "555-0000",
"description": "Parent company of Contoso Pharmaceuticals, etc.",
"accountid": "59d88f5e-6629-ed11-9db1-0022482746b6",
"_transactioncurrencyid_value": "228f42f8-e646-e111-8eb7-78e7d162ced1"
}
```

Salida de la consola:

```
Created and retrieved the initial account, shown below:
{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#accounts(name,revenue,telephone1:description)/$entity",
  "@odata.etag": "W/\"72965013\"",
  "name": "Contoso Ltd",
  "revenue": 5000000.0,
  "telephone1": "555-0000",
  "description": "Parent company of Contoso Pharmaceuticals, etc.",
  "accountid": "59d88f5e-6629-ed11-9db1-0022482746b6",
  "_transactioncurrencyid_value": "228f42f8-e646-e111-8eb7-78e7d162ced1"
}
```

Sección 1: GET condicional

Esta sección del programa demuestra cómo realizar recuperaciones condicionales para optimizar el ancho de banda de la red y el procesamiento del servidor mientras se mantiene el estado de fila más actual en el cliente. Más información:[Recuperaciones condicionales](#).

1. Intenta recuperar la cuenta `Contoso Ltd.` solo si *no* coincide con la versión actual, identificada por el valor ETag inicial que se devolvió cuando se creó la fila de la cuenta. Esta condición se representa por el encabezado `If-None-Match`.

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/accounts(59d88f5e-6629-ed11-9db1-
0022482746b6)?$select=name,revenue,telephone1:description HTTP/1.1
If-None-Match: W/"72965013"
OData-MaxVersion: 4.0
OData-Version: 4.0
Accept: application/json
```

Respuesta:

```
HTTP
```

```
HTTP/1.1 304 NotModified
```

Salida de la consola:

```
Account record retrieved using ETag: W/"72965013"  
Expected outcome: Entity was not modified so nothing was returned.
```

El valor de respuesta, `304 NotModified`, indica que la fila de la tabla actual es la más actual, por lo que el servidor *no* devuelve la fila solicitada en el cuerpo de la respuesta.

2. Actualice la cuenta modificando su propiedad de número de teléfono principal.

Solicitud:

```
HTTP
```

```
PUT [Organization Uri]/api/data/v9.2/accounts(59d88f5e-6629-ed11-9db1-  
0022482746b6)/telephone1 HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json  
  
{ "value": "555-0001"}
```

Respuesta:

```
HTTP
```

```
HTTP/1.1 204 NoContent  
OData-Version: 4.0
```

Salida de la consola:

```
Modified account record retrieved using ETag: W/"72965013"
```

3. Reintente la misma operación GET condicional, de nuevo con el valor de ETag original. Esta vez la operación devuelve los datos solicitados porque la versión en el servidor es diferente (y más reciente) que la versión identificada en la solicitud. Como en todas las recuperaciones de filas de la tabla, la respuesta incluye un encabezado ETag que identifica la versión actual.

Solicitud:

```
HTTP

GET [Organization Uri]/api/data/v9.2/accounts(59d88f5e-6629-ed11-9db1-0022482746b6)?$select=name,revenue,telephone1,description HTTP/1.1
If-None-Match: W/"72965013"
OData-MaxVersion: 4.0
OData-Version: 4.0
Accept: application/json
```

Respuesta:

```
HTTP

HTTP/1.1 200 OK
ETag: W/"72965025"
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#accounts(name,revenue,telephone1,description)/$entity",
  "@odata.etag": "W/\"72965025\"",
  "name": "Contoso Ltd",
  "revenue": 5000000.0,
  "telephone1": "555-0001",
  "description": "Parent company of Contoso Pharmaceuticals, etc.",
  "accountid": "59d88f5e-6629-ed11-9db1-0022482746b6",
  "_transactioncurrencyid_value": "228f42f8-e646-e111-8eb7-78e7d162ced1"
}
```

Salida de la consola:

```
Modified account record retrieved using ETag: W/"72965013"
Notice the updated ETag value and telephone number
{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#accounts(name,revenue,telephone1,description)/$entity",
  "@odata.etag": "W/\"72965025\"",
```

```
"name": "Contoso Ltd",
"revenue": 5000000.0,
"telephone1": "555-0001",
"description": "Parent company of Contoso Pharmaceuticals, etc.",
"accountid": "59d88f5e-6629-ed11-9db1-0022482746b6",
"_transactioncurrencyid_value": "228f42f8-e646-e111-8eb7-78e7d162ced1"
}
```

Sección 2: Simultaneidad optimista al eliminar y actualizar

Esta sección del programa muestra cómo realizar operaciones de eliminación y actualización condicionales. El uso más común de tales operaciones es implementar un enfoque de simultaneidad optimista para el procesamiento de filas en un entorno multiusuario. Más información:[Aplicar simultaneidad optimista](#)

1. Intente eliminar la cuenta original si y solo si coincide con la versión original (valor de ETag). Esta condición se representa por el encabezado `If-Match`. Esta operación falla porque la fila de la cuenta se actualizó en la sección anterior, por lo que su versión se actualizó en el servidor.

Solicitud:

```
HTTP

DELETE [Organization Uri]/api/data/v9.2/accounts(59d88f5e-6629-ed11-9db1-0022482746b6) HTTP/1.1
If-Match: W/"72965013"
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

```
HTTP

HTTP/1.1 412 PreconditionFailed
OData-Version: 4.0

{
"error": {
    "code": "0x80060882",
    "message": "The version of the existing record doesn't match the RowVersion property provided."
```

```
}
```

Salida de la consola:

```
Attempting to delete the account using the original ETag value  
Expected Error: The version of the existing record doesn't match the  
RowVersion property provided.  
Account not deleted using ETag 'W/"72965013"', status code:  
'PreconditionFailed'.
```

2. Intente actualizar la cuenta si y solo si coincide con el valor de ETag original. Una vez más esta condición está representada por el encabezado `If-Match` y la operación produce un error por la misma razón.

Solicitud:

```
HTTP  
  
PATCH [Organization Uri]/api/data/v9.2/accounts(59d88f5e-6629-ed11-  
9db1-0022482746b6) HTTP/1.1  
If-Match: W/"72965013"  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json  
  
{  
    "telephone1": "555-0002",  
    "revenue": 6000000  
}
```

Respuesta:

```
HTTP  
  
HTTP/1.1 412 PreconditionFailed  
OData-Version: 4.0  
  
{  
    "error": {  
        "code": "0x80060882",  
        "message": "The version of the existing record doesn't match the  
RowVersion property provided."  
    }  
}
```

Salida de la consola:

```
Attempting to update the account using the original ETag value  
Expected Error: The version of the existing record doesn't match the  
RowVersion property provided.  
Account not updated using ETag 'W/"72965013"', status code:  
'PreconditionFailed'.
```

3. Vuelva a intentar una actualización, pero en su lugar utilice el valor ETag actual obtenido de la última recuperación de fila en la sección anterior.

Solicitud:

```
HTTP  
  
PATCH [Organization Uri]/api/data/v9.2/accounts(59d88f5e-6629-ed11-  
9db1-0022482746b6) HTTP/1.1  
If-Match: W/"72965025"  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json  
  
{  
    "telephone1": "555-0003",  
    "revenue": 6000000  
}
```

Respuesta:

```
HTTP  
  
HTTP/1.1 204 NoContent  
OData-Version: 4.0  
OData-EntityId: [Organization Uri]/api/data/v9.2/accounts(59d88f5e-  
6629-ed11-9db1-0022482746b6)
```

Salida de la consola:

```
Attempting to update the account using the current ETag value  
Account successfully updated using ETag: W/"72965025".
```

4. Confirme que la actualización se realizó satisfactoriamente recuperando y generando el estado de cuenta actual. Esto usa una solicitud `GET` básica.

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/accounts(59d88f5e-6629-ed11-9db1-0022482746b6)?$select=name,revenue,telephone1,description HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
ETag: W/"72965035"
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#accounts(name,revenue,telephone1,description)/$entity",
  "@odata.etag": "W/\\"72965035\\\"",
  "name": "Contoso Ltd",
  "revenue": 6000000.0,
  "telephone1": "555-0003",
  "description": "Parent company of Contoso Pharmaceuticals, etc.",
  "accountid": "59d88f5e-6629-ed11-9db1-0022482746b6",
  "_transactioncurrencyid_value": "228f42f8-e646-e111-8eb7-78e7d162ced1"
}
```

Salida de la consola:

```
Below is the final state of the account
{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#accounts(name,revenue,telephone1,description)/$entity",
  "@odata.etag": "W/\\"72965035\\\"",
  "name": "Contoso Ltd",
  "revenue": 6000000.0,
  "telephone1": "555-0003",
  "description": "Parent company of Contoso Pharmaceuticals, etc.",
  "accountid": "59d88f5e-6629-ed11-9db1-0022482746b6",
```

```
        "_transactioncurrencyid_value": "228f42f8-e646-e111-8eb7-78e7d162ced1"  
    }  

```

Sección 3: Eliminar registros de muestra

Esta sección simplemente elimina el registro creado en la [Sección 0: Crear registros de muestra](#). Utiliza una solicitud `$batch`.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/$batch HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json  
  
--batch_98e0fdc2-a298-4f42-85a8-da0536140b78  
Content-Type: application/http  
Content-Transfer-Encoding: binary  
Content-Length: 115  
  
DELETE /api/data/v9.2/accounts(59d88f5e-6629-ed11-9db1-0022482746b6)  
HTTP/1.1  
  
--batch_98e0fdc2-a298-4f42-85a8-da0536140b78--
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK  
OData-Version: 4.0  
  
--batchresponse_7bf5a9a8-5b97-4fb0-9243-668f3113e6eb  
Content-Type: application/http  
Content-Transfer-Encoding: binary  
  
HTTP/1.1 204 No Content  
OData-Version: 4.0  
  
--batchresponse_7bf5a9a8-5b97-4fb0-9243-668f3113e6eb--
```

Salida de la consola:

```
Deleting created records.
```

Consulte también

[Usar la API web de Dataverse](#)

[Realizar operaciones condicionales mediante la API web](#)

[Ejemplo de operaciones condicionales de la API web \(C#\)](#)

[Ejemplo de operaciones condicionales de la API web \(JavaScript del lado del cliente\)](#)

Nota

¿Puede indicarnos sus preferencias de idioma de documentación? [Realice una breve encuesta.](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad](#)).

Ejemplo de funciones y acciones de la API web

Artículo • 24/08/2023

Este grupo de ejemplos muestra cómo realizar funciones y acciones enlazadas y sin enlazar, incluidas acciones personalizadas, mediante la API web de Microsoft Dataverse. Este ejemplo se implementa como proyecto independiente para los siguientes idiomas:

- [Ejemplo de funciones y acciones de la API web \(C#\)](#)
- [Ejemplo de funciones y acciones \(JavaScript del lado del cliente\)](#)

En este artículo se explican la estructura y el contenido del ejemplo en un nivel superior de lenguaje neutro. Revise los artículos de ejemplo vinculados anteriores para detalles de implementación específicos del idioma sobre cómo realizar las operaciones descritas en este artículo.

Demostraciones

Este ejemplo se divide en las siguientes secciones principales, que contienen operaciones de funciones y acciones de la API web que se describen minuciosamente en los artículos conceptuales asociados.

Sección del artículo	Vista asociada de los artículos
Sección 1: Función independiente WhoAmI	WhoAmI Function Funciones sin enlazar
Sección 2: Función independiente FormatAddress	FormatAddress Function Paso de parámetros a una función
Sección 3: Función independiente InitializeFrom	InitializeFrom Function Crear un registro a partir de otro registro Asignar columnas de tabla Personalizar las asignaciones de tablas y columnas
Sección 4: Función independiente RetrieveCurrentOrganization	RetrieveCurrentOrganization Function
Sección 6: Función vinculada de API personalizada IsSystemAdmin	Funciones enlazadas Usar funciones enlazadas Ejemplo: API personalizada IsSystemAdmin Crear y usar API personalizadas

Sección del artículo	Vista asociada de los artículos
Sección 7: Acción sin consolidar GrantAccess	GrantAccess Action Compartir y asignar RetrievePrincipalAccess Function
Sección 8: Acciones vinculadas AddPrivilegesRole	Acciones enlazadas Usar acciones enlazadas AddPrivilegesRole Action Referencia de tabla/entidad de Security Role (Role)
Sección 9: Eliminar registros de muestra	Eliminación básica Ejecute las operaciones por lotes mediante API web

Las siguientes secciones contienen una breve explicación de las operaciones de la API web de Dataverse realizadas, junto con mensajes HTTP correspondientes y la salida asociada de la consola.

Sección 1: Función independiente WhoAmI

[WhoAmI Function](#) es una función independiente simple y de uso común.

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/WhoAmI HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.WhoAmIResponse",
    "BusinessUnitId": "cca3985e-c618-ea11-a811-000d3a33f066",
    "UserId": "2138bd90-ec19-ea11-a811-000d3a334e11",
```

```
        "OrganizationId": "f2c9290b-0806-4d48-bf9c-3814d4286755"  
    }  
}
```

Salida de la consola:

```
WhoAmI tells us:  
WhoAmIResponse.BusinessUnitId:cca3985e-c618-ea11-a811-000d3a33f066  
WhoAmIResponse.UserId:2138bd90-ec19-ea11-a811-000d3a334e11  
WhoAmIResponse.OrganizationId:f2c9290b-0806-4d48-bf9c-3814d4286755
```

El valor `BusinessUnitId` recuperado aquí se utilizará en la [Sección 8: Acción enlazada AddPrivilegesRole](#).

Sección 2: Función independiente FormatAddress

La función `FormatAddress` es una función independiente que requiere que se establezcan parámetros. Devuelve una cadena que representa una dirección de acuerdo con los requisitos específicos de formato del país/región.

En este ejemplo, los parámetros se establecen mediante valores de parámetros de cadena de consulta.

1. Una solicitud de dirección en los Estados Unidos:

Solicitud:

```
HTTP  
  
GET [Organization  
Uri]/api/data/v9.2/FormatAddress(Line1=@p1,City=@p2,StateOrProvince=@p3  
,PostalCode=@p4,Country=@p5)?  
@p1='123%20Maple%20St.'&@p2='Seattle'&@p3='WA'&@p4='98007'&@p5='USA'  
HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json
```

Respuesta:

```
HTTP
```

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
"@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.FormatAddressRespon
se",
"Address": "123 Maple St.\r\nSeattle, WA 98007\r\nUSA"
}
```

Salida de la consola:

```
USA Formatted Address:
123 Maple St.
Seattle, WA 98007
USA
```

2. Una solicitud de una dirección en Japón.

Solicitud:

```
HTTP

GET [Organization
Uri]/api/data/v9.2/FormatAddress(Line1=@p1,City=@p2,StateOrProvince=@p3
,PostalCode=@p4,Country=@p5)?@p1='1-2-
3%20Sakura'&@p2='Nagoya'&@p3='Aichi'&@p4='455-2345'&@p5='JAPAN'
HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

```
HTTP

HTTP/1.1 200 OK
OData-Version: 4.0

{
"@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.FormatAddressRespon
se",
"Address": "455-2345\r\nAichi\r\nNagoya\r\n1-2-3 Sakura\r\nJAPAN"
}
```

Salida de la consola:

```
JAPAN Formatted Address:  
455-2345  
Aichi  
Nagoya  
1-2-3 Sakura  
JAPAN
```

Sección 3: Función independiente InitializeFrom

La función [InitializeFrom](#) es una función independiente que requiere parámetros. Esta función devuelve los datos para crear un nuevo registro en el contexto de un registro existente. Según los datos de configuración para controlar qué datos se copian, los datos de registro devueltos incluyen datos copiados del registro original.

Más información:

- [Crear un registro a partir de otro registro](#)
- [Asignar columnas de tabla](#)
- [Personalizar las asignaciones de tablas y columnas](#)

1. Cree un registro para que sea el registro original:

Solicitud:

```
HTTP  
  
POST [Organization Uri]/api/data/v9.2/accounts HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json  
  
{  
    "accountcategorycode": 1,  
    "address1_addressstypecode": 3,  
    "address1_city": "Redmond",  
    "address1_country": "USA",  
    "address1_line1": "123 Maple St.",  
    "address1_name": "Corporate Headquarters",  
    "address1_postalcode": "98000",  
    "address1_shippingmethodcode": 4,  
    "address1_stateorprovince": "WA",
```

```
"address1_telephone1": "555-1234",
"customertypecode": 3,
"description": "Contoso is a business consulting company.",
"emailaddress1": "info@contoso.com",
"industrycode": 7,
"name": "Contoso Consulting",
"numberofemployees": 150,
"ownershipcode": 2,
"preferredcontactmethodcode": 2,
"telephone1": "(425) 555-1234"
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization Uri]/api/data/v9.2/accounts(98d463e4-6d29-ed11-9db1-00224804f8e2)
```

2. Use `InitializeFrom` para obtener los datos de un nuevo registro a partir del registro original.

Solicitud:

HTTP

```
GET [Organization
Uri]/api/data/v9.2/InitializeFrom(EntityMoniker=@p1, TargetEntityName=@p2, TargetFieldType=@p3)?@p1={'@odata.id':'accounts(98d463e4-6d29-ed11-9db1-00224804f8e2)' }&@p2='account'&@p3=Microsoft.Dynamics.CRM.TargetFieldType'ValidForCreate' HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
Preference-Applied: return=representation
OData-Version: 4.0

{
"@odata.context": "[Organization
```

```
Uri]/api/data/v9.2/$metadata#accounts/$entity",
"@odata.type": "#Microsoft.Dynamics.CRM.account",
"parentaccountid@odata.bind": "accounts(98d463e4-6d29-ed11-9db1-
00224804f8e2)"
}
```

Salida de la consola:

```
New data based on original record:
{
  "@odata.context": "[Organization
URI]/api/data/v9.2/$metadata#accounts/$entity",
  "@odata.type": "#Microsoft.Dynamics.CRM.account",
  "parentaccountid@odata.bind": "accounts(98d463e4-6d29-ed11-9db1-
00224804f8e2)"
}
```

ⓘ Nota

Si no hay columnas asignadas para esta relación, solo se incluyen los valores de columna mínimos, como se muestra arriba. En este caso, solo la búsqueda `parentaccountid` para asociar el nuevo registro con el original.

Si todas las columnas disponibles están asignadas para esta relación, el valor devuelto incluye más datos del registro original, por ejemplo:

JSON

```
{
  "@odata.context": "[Organization
URI]/api/data/v9.2/$metadata#accounts/$entity",
  "@odata.type": "#Microsoft.Dynamics.CRM.account",
  "territorycode": 1,
  "address2_freighttermscode": 1,
  "address2_shippingmethodcode": 1,
  "address1_telephone1": "555-1234",
  "accountclassificationcode": 1,
  "creditonhold": false,
  "donotbulkemail": false,
  "donotsendmm": false,
  "emailaddress1": "info@contoso.com",
  "address1_line1": "123 Maple St.",
  "customertypecode": 3,
  "ownershipcode": 2,
  "businesstypecode": 1,
  "donotpostalmail": false,
```

```

    "donotbulkpostalmail": false,
    "name": "Contoso Consulting",
    "address1_city": "Redmond",
    "description": "Contoso is a business consulting company.",
    "donotemail": false,
    "address2_addressstypecode": 1,
    "donotphone": false,
    "statuscode": 1,
    "address1_name": "Corporate Headquarters",
    "followemail": true,
    "preferredcontactmethodcode": 2,
    "numberofemployees": 150,
    "industrycode": 7,
    "telephone1": "(425) 555-1234",
    "address1_shippingmethodcode": 4,
    "donotfax": false,
    "address1_addressstypecode": 3,
    "customersizecode": 1,
    "marketingonly": false,
    "accountratingcode": 1,
    "shippingmethodcode": 1,
    "address1_country": "USA",
    "participatesinworkflow": false,
    "accountcategorycode": 1,
    "address1_postalcode": "98000",
    "address1_stateorprovince": "WA",
    "parentaccountid@odata.bind": "accounts(fe9873ac-2f1b-ed11-b83e-00224837179f)"
}

```

3. Cree un nuevo registro con los datos devueltos de `InitializeFrom`.

Solicitud:

HTTP

```

POST [Organization Uri]/api/data/v9.2/accounts HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#accounts/$entity",
    "@odata.type": "#Microsoft.Dynamics.CRM.account",
    "parentaccountid@odata.bind": "accounts(98d463e4-6d29-ed11-9db1-00224804f8e2)",
    "name": "Contoso Consulting Chicago Branch",
    "address1_city": "Chicago",
    "address1_line1": "456 Elm St.",
    "address1_name": "Chicago Branch Office",
    "address1_postalcode": "60007",
}

```

```
"address1_stateorprovince": "IL",
"address1_telephone1": "(312) 555-3456",
"numberofemployees": 12
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization Uri]/api/data/v9.2/accounts(9ad463e4-6d29-ed11-9db1-00224804f8e2)
```

Salida de la consola:

New Record:

```
{
"@odata.context": "[Organization
URI]/api/data/v9.2/$metadata#accounts/$entity",
"@odata.type": "#Microsoft.Dynamics.CRM.account",
"parentaccountid@odata.bind": "accounts(98d463e4-6d29-ed11-9db1-00224804f8e2)",
"name": "Contoso Consulting Chicago Branch",
"address1_city": "Chicago",
"address1_line1": "456 Elm St.",
"address1_name": "Chicago Branch Office",
"address1_postalcode": "60007",
"address1_stateorprovince": "IL",
"address1_telephone1": "(312) 555-3456",
"numberofemployees": 12
}
```

Sección 4: Función independiente RetrieveCurrentOrganization

La función [RetrieveCurrentOrganization](#) devuelve información sobre la organización actual. Requiere un valor de tipo de enumeración [EndpointAccessType](#) como parámetro.

[RetrieveCurrentOrganization](#) devuelve un tipo complejo

[RetrieveCurrentOrganizationResponse](#) que incluye una propiedad [Detail](#) que es un tipo complejo [DetalleOrganización](#), que tiene propiedades complejas que utilizan el tipo complejo [EndpointCollection](#), tipo de enumeración [EndpointType](#) y Tipo de enumeración de estado de organización

ⓘ Nota

Observe cómo se pasa el valor del parámetro `AccessType` tipo de enumeración `EndpointAccessType` en la URL. Se requiere el nombre completo con el nombre de miembro seleccionado.

Solicitud:

HTTP

```
GET [Organization
Uri]/api/data/v9.2/RetrieveCurrentOrganization(AccessType=@p1)?
@p1=Microsoft.Dynamics.CRM.EndpointAccessType'Default' HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.RetrieveCurrentOrganizationResponse",
    "Detail": {
        "OrganizationId": "f2c9290b-0806-4d48-bf9c-3814d4286755",
        "FriendlyName": "[Organization Name]",
        "OrganizationVersion": "9.2.22074.168",
        "EnvironmentId": "Default-f6976b02-5a42-4e90-ac3d-8bf516ce0859",
        "DatacenterId": "695014e1-baf4-4d7e-9d3d-2261d4aa780",
        "Geo": "NA",
        "TenantId": "f6976b02-5a42-4e90-ac3d-8bf516ce0859",
        "UrlName": "org619726b5",
        "UniqueName": "org0335df44",
        "State": "Enabled",
        "Endpoints": {
            "Count": 3,
            "IsReadOnly": false,
            "Keys": [
                "WebApplication",
                "OrganizationService",
                "OrganizationDataService"
            ],
            "Values": [

```

```

        "[Organization URI]/",
        "[Organization URI]/XRMServices/2011/Organization.svc",
        "[Organization URI]/XRMServices/2011/OrganizationData.svc"
    ]
}
}
}

```

Salida de la consola:

```

Data returned with RetrieveCurrentOrganizationResponse:
{
    "OrganizationId": "f2c9290b-0806-4d48-bf9c-3814d4286755",
    "FriendlyName": "[Organization Name]",
    "OrganizationVersion": "9.2.22074.168",
    "EnvironmentId": "Default-f6976b02-5a42-4e90-ac3d-8bf516ce0859",
    "DatacenterId": "695014e1-baf8-4d7e-9d3d-2261d4aaf780",
    "Geo": "NA",
    "TenantId": "f6976b02-5a42-4e90-ac3d-8bf516ce0859",
    "UrlName": "org619726b5",
    "UniqueName": "org0335df44",
    "Endpoints": {
        "Count": 3,
        "IsReadOnly": false,
        "Keys": [
            "WebApplication",
            "OrganizationService",
            "OrganizationDataService"
        ],
        "Values": [
            "[Organization URI]/",
            "[Organization URI]/XRMServices/2011/Organization.svc",
            "[Organization URI]/XRMServices/2011/OrganizationData.svc"
        ]
    },
    "State": "Enabled"
}

```

Sección 5: Función independiente RetrieveTotalRecordCount

La función [RetrieveTotalRecordCount](#) devuelve datos sobre el número total de registros para entidades específicas. Los datos recuperados son de una instantánea de las últimas 24 horas, por lo que no es un recuento exacto en un momento dado.

Solicitud:

HTTP

```
GET [Organization
Uri]/api/data/v9.2/RetrieveTotalRecordCount(EntityNames=@p1)?@p1=
["account","contact"] HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.RetrieveTotalRecordCount
Response",
  "EntityRecordCountCollection": {
    "Count": 2,
    "IsReadOnly": false,
    "Keys": [
      "account",
      "contact"
    ],
    "Values": [
      19,
      3
    ]
  }
}
```

Salida de la consola:

```
The number of records for each table according to RetrieveTotalRecordCount:
  account:19
  contact:3
```

Sección 6: Función vinculada de API personalizada IsSystemAdmin

Para demostrar una función enlazada, este ejemplo importa un mensaje personalizado definido dentro de una solución antes de ejecutar esta parte del ejemplo.

El ejemplo usa el mensaje personalizado `sample_IsSystemAdmin` que se define mediante una API [personalizada](#). Puede encontrar detalles sobre esta API personalizada aquí: [Ejemplo: API personalizada IsSystemAdmin](#).

ⓘ Nota

Al usar una función o acción enlazada, debe incluir el nombre completo, que incluye `Microsoft.Dynamics.CRM.` + <nombre de la función o acción> en la URL.

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/systemusers(ce31e691-f559-ec11-8f8f-000d3a308de4)/Microsoft.Dynamics.CRM.sample_IsSystemAdmin HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.sample_IsSystemAdminResponse",
    "HasRole": false
}
```

Esta muestra recupera 10 registros de usuario y prueba cada uno para determinar si cada usuario tiene el Sistema Administrador rol de seguridad.

Salida de la consola:

Los nombres reales dependen de las personas de su entorno.

Top 10 users and whether they have System Administrator role.
Gediminas Matulis does not have the System Administrator role.
Gaby Frost does not have the System Administrator role.
Henrikas Martinkus does not have the System Administrator role.
Alain Davignon HAS the System Administrator role.
Isobel Macintyre HAS the System Administrator role.
Ale Laukaitiene HAS the System Administrator role.
Rudabeh Yekta HAS the System Administrator role.
Grazina Januliene HAS the System Administrator role.
Pranciskus Sukys HAS the System Administrator role.
Asha Sawant HAS the System Administrator role.

Para ver otro ejemplo de una función enlazada, vea el uso de la función [RetrievePrincipalAccess](#) en el siguiente ejemplo.

Sección 7: Acción sin consolidar GrantAccess

La acción [GrantAccess](#) es una acción independiente que permite a las personas compartir privilegios específicos con otros usuarios en su entorno.

El código de ejemplo demuestra las operaciones siguientes:

1. Crear un registro para compartir.
2. Encuentre un usuario habilitado que no sea el usuario actual.
3. Utilice la función [RetrievePrincipalAccess](#) para determinar qué derechos de acceso tiene el usuario para el registro creado.

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/systemusers(ce31e691-f559-ec11-8f8f-000d3a308de4)/Microsoft.Dynamics.CRM.RetrievePrincipalAccess(Target=@p1)?@p1={'@odata.id':'accounts(659876fd-6d29-ed11-9db1-00224804f8e2)'}  
HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
"@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.RetrievePrincipalAccessResponse",
"AccessRights": "ShareAccess"
}
```

Salida de la consola:

```
Testing user: Gediminas Matulis
Current users access: ShareAccess
```

4. Si el usuario no tiene `AccessRights`.`DeleteAccess`, otorgue al usuario este acceso mediante la acción `GrantAccess`.

Solicitud:

```
HTTP

POST [Organization Uri]/api/data/v9.2/GrantAccess HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
"Target": {
    "accountid": "659876fd-6d29-ed11-9db1-00224804f8e2",
    "@odata.type": "Microsoft.Dynamics.CRM.account"
},
"PrincipalAccess": {
    "AccessMask": "DeleteAccess",
    "Principal": {
        "systemuserid": "ce31e691-f559-ec11-8f8f-000d3a308de4",
        "@odata.type": "Microsoft.Dynamics.CRM.systemuser"
    }
}
}
```

Respuesta:

```
HTTP
```

```
HTTP/1.1 204 NoContent
```

```
OData-Version: 4.0
```

5. Una vez que `DeleteAccess` se ha concedido, la misma llamada a [RetrievePrincipalAccess Function](#) muestra que ahora tienen acceso para eliminar este registro:

Solicitud:

```
HTTP
```

```
GET [Organization Uri]/api/data/v9.2/systemusers(ce31e691-f559-ec11-  
8f8f-  
000d3a308de4)/Microsoft.Dynamics.CRM.RetrievePrincipalAccess(Target=@p1  
)?@p1={'@odata.id':'accounts(659876fd-6d29-ed11-9db1-00224804f8e2)'}  
HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json
```

Respuesta:

```
HTTP
```

```
HTTP/1.1 200 OK  
OData-Version: 4.0
```

```
{  
"@odata.context": "[Organization  
Uri]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.RetrievePrincipalAc  
cessResponse",  
"AccessRights": "DeleteAccess, ShareAccess"  
}
```

Salida de la consola:

```
Gediminas Matulis was granted DeleteAccess
```

Sección 8: Acción vinculada AddPrivilegesRole

La acción `AddPrivilegesRole` es una acción asociada al [tipo de entidad de rol](#). Es la forma de agregar privilegios a un rol de seguridad.

El código de ejemplo realiza las operaciones siguientes:

1. Crear un rol de seguridad. El rol debe asociarse a una unidad de negocio. El valor de identificación de la unidad de negocio se recuperó en la [Sección 1: Función independiente WhoAmI](#).

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/roles HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
  "businessunitid@odata.bind": "businessunits(cca3985e-c618-ea11-a811-
  000d3a33f066)",
  "name": "Test Role"
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization Uri]/api/data/v9.2/roles(669876fd-6d29-
ed11-9db1-00224804f8e2)
```

2. Recuperar el rol, expandiendo la propiedad de navegación con valor de colección `roleprivileges_association` para incluir los privilegios incluidos con el rol.

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/roles(669876fd-6d29-ed11-9db1-
00224804f8e2)?
$select=roleid&$expand=roleprivileges_association($select=name)
HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
ETag: W/"13278232"
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#roles(roleid,roleprivileges_association(name))/$entity",
  "@odata.etag": "W/\"13278232\",
  "roleid": "669876fd-6d29-ed11-9db1-00224804f8e2",
  "roleprivileges_association": [
    {
      "@odata.etag": "W/\"142279\",
      "name": "prvReadSharePointData",
      "privilegeid": "fecbd29c-df64-4ede-a611-47226b402c22"
    },
    {
      "@odata.etag": "W/\"142304\",
      "name": "prvReadSdkMessage",
      "privilegeid": "94c3ac2c-eb23-41cb-a903-4e2e49e910b4"
    },
    {
      "@odata.etag": "W/\"142421\",
      "name": "prvWriteSharePointData",
      "privilegeid": "cfdd12cf-090b-4599-8302-771962d2350a"
    },
    {
      "@odata.etag": "W/\"142477\",
      "name": "prvReadSdkMessageProcessingStepImage",
      "privilegeid": "122e085f-8c52-47e8-8415-875dee1c961e"
    },
    {
      "@odata.etag": "W/\"142695\",
      "name": "prvReadSdkMessageProcessingStep",
      "privilegeid": "db10a828-ec49-4035-8b7e-c58efaf169ec"
    },
    {
      "@odata.etag": "W/\"142713\",
      "name": "prvReadPluginAssembly",
      "privilegeid": "f5b50296-a212-488a-be92-cbccaa8971717"
    },
    {
      "@odata.etag": "W/\"142735\",
      "name": "prvCreateSharePointData",
      "privilegeid": "5eb85025-363b-46ea-a77e-ce24159cd231"
    },
    {
      "@odata.etag": "W/\"142740\",
      "name": "prvReadPluginType",
      "privilegeid": "9365005c-4703-473b-8d3c-d073cf8670c"
    },
    {
      "@odata.etag": "W/\"142741\",
      "name": "prvUpdateSharePointData"
    }
  ]
}
```

```
        "@odata.etag": "W/\"142761\"",  
        "name": "prvReadSharePointDocument",  
        "privilegeid": "d71fc8d0-99bc-430e-abd7-d95c64f11e9c"  
    }  
]  
}
```

3. Muestra el número de privilegios creados por defecto para el nuevo rol.

Salida de la consola:

```
Number of privileges in new role: 9  
    prvReadSharePointData  
    prvReadSdkMessage  
    prvWriteSharePointData  
    prvReadSdkMessageProcessingStepImage  
    prvReadSdkMessageProcessingStep  
    prvReadPluginAssembly  
    prvCreateSharePointData  
    prvReadPluginType  
    prvReadSharePointDocument
```

4. Recuperar la definición de los privilegios `prvCreateAccount` y `prvReadAccount` de [privilege EntityType](#).

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/privileges?  
$select=name&$filter=name eq 'prvCreateAccount' or name eq  
'prvReadAccount' HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK  
OData-Version: 4.0  
  
{  
    "@odata.context": "[Organization  
Uri]/api/data/v9.2/$metadata#privileges(name)",
```

```

"value": [
  {
    "@odata.etag": "W/\"142189\"",
    "name": "prvReadAccount",
    "privilegeid": "886b280c-6396-4d56-a0a3-2c1b0a50ceb0"
  },
  {
    "@odata.etag": "W/\"142359\"",
    "name": "prvCreateAccount",
    "privilegeid": "d26fe964-230b-42dd-ad93-5cc879de411e"
  }
]
}

```

5. Preparar una lista de instancias de [RolePrivilege ComplexType](#) para los privilegios `prvCreateAccount` y `prvReadAccount` con la propiedad `Depth` establecida en [PrivilegeDepth.Basic](#).

6. Pasar la lista como el parámetro `AddPrivilegesRole.Privileges`.

Solicitud:

HTTP

```

POST [Organization Uri]/api/data/v9.2/roles(669876fd-6d29-ed11-9db1-
00224804f8e2)/Microsoft.Dynamics.CRM.AddPrivilegesRole HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

```

```

{
  "Privileges": [
    {
      "Depth": "Basic",
      "PrivilegeId": "886b280c-6396-4d56-a0a3-2c1b0a50ceb0",
      "BusinessUnitId": "cca3985e-c618-ea11-a811-000d3a33f066",
      "PrivilegeName": "prvReadAccount"
    },
    {
      "Depth": "Basic",
      "PrivilegeId": "d26fe964-230b-42dd-ad93-5cc879de411e",
      "BusinessUnitId": "cca3985e-c618-ea11-a811-000d3a33f066",
      "PrivilegeName": "prvCreateAccount"
    }
  ]
}

```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent  
OData-Version: 4.0
```

7. Recupere los privilegios asociados con el rol nuevamente para confirmar que se agregaron.

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/roles(669876fd-6d29-ed11-9db1-  
00224804f8e2)?  
$select=roleid&$expand=roleprivileges_association($select=name)  
HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK  
ETag: W/"13278248"  
OData-Version: 4.0  
  
{  
    "@odata.context": "[Organization  
Uri]/api/data/v9.2/$metadata#roles(roleid,roleprivileges_association(na  
me))/$entity",  
    "@odata.etag": "W/\\"13278248\\\"",  
    "roleid": "669876fd-6d29-ed11-9db1-00224804f8e2",  
    "roleprivileges_association": [  
        {  
            "@odata.etag": "W/\\"142189\\\"",  
            "name": "prvReadAccount",  
            "privilegeid": "886b280c-6396-4d56-a0a3-2c1b0a50ceb0"  
        },  
        {  
            "@odata.etag": "W/\\"142279\\\"",  
            "name": "prvReadSharePointData",  
            "privilegeid": "fecbd29c-df64-4ede-a611-47226b402c22"  
        },  
        {  
            "@odata.etag": "W/\\"142304\\\"",  
            "name": "prvReadSdkMessage",  
            "privilegeid": "94c3ac2c-eb23-41cb-a903-4e2e49e910b4"  
        },  
    ]  
}
```

```

{
    "@odata.etag": "W/\"142359\",
    "name": "prvCreateAccount",
    "privilegeid": "d26fe964-230b-42dd-ad93-5cc879de411e"
},
{
    "@odata.etag": "W/\"142421\",
    "name": "prvWriteSharePointData",
    "privilegeid": "cfdd12cf-090b-4599-8302-771962d2350a"
},
{
    "@odata.etag": "W/\"142477\",
    "name": "prvReadSdkMessageProcessingStepImage",
    "privilegeid": "122e085f-8c52-47e8-8415-875dee1c961e"
},
{
    "@odata.etag": "W/\"142695\",
    "name": "prvReadSdkMessageProcessingStep",
    "privilegeid": "db10a828-ec49-4035-8b7e-c58efaf169ec"
},
{
    "@odata.etag": "W/\"142713\",
    "name": "prvReadPluginAssembly",
    "privilegeid": "f5b50296-a212-488a-be92-cbcca8971717"
},
{
    "@odata.etag": "W/\"142735\",
    "name": "prvCreateSharePointData",
    "privilegeid": "5eb85025-363b-46ea-a77e-ce24159cd231"
},
{
    "@odata.etag": "W/\"142740\",
    "name": "prvReadPluginType",
    "privilegeid": "9365005c-4703-473b-8d3c-d073cf8670c"
},
{
    "@odata.etag": "W/\"142761\",
    "name": "prvReadSharePointDocument",
    "privilegeid": "d71fc8d0-99bc-430e-abd7-d95c64f11e9c"
}
]
}

```

Salida de la consola:

```

Number of privileges after: 11
prvReadAccount
prvReadSharePointData
prvReadSdkMessage
prvCreateAccount
prvWriteSharePointData

```

```
prvReadSdkMessageProcessingStepImage  
prvReadSdkMessageProcessingStep  
prvReadPluginAssembly  
prvCreateSharePointData  
prvReadPluginType  
prvReadSharePointDocument
```

Sección 9: Eliminar registros de muestra

Cada registro creado en esta muestra se agregó a una lista para ser eliminado al final. Estos registros se eliminan mediante una solicitud `$batch`.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/$batch HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json  
  
--batch_d6010246-cd97-429f-bc05-cf20054cf8e8a  
Content-Type: application/http  
Content-Transfer-Encoding: binary  
Content-Length: 121  
  
DELETE /api/data/v9.2/accounts(98d463e4-6d29-ed11-9db1-00224804f8e2)  
HTTP/1.1  
  
--batch_d6010246-cd97-429f-bc05-cf20054cf8e8a  
Content-Type: application/http  
Content-Transfer-Encoding: binary  
Content-Length: 121  
  
DELETE /api/data/v9.2/accounts(9ad463e4-6d29-ed11-9db1-00224804f8e2)  
HTTP/1.1  
  
--batch_d6010246-cd97-429f-bc05-cf20054cf8e8a  
Content-Type: application/http  
Content-Transfer-Encoding: binary  
Content-Length: 122  
  
DELETE /api/data/v9.2/solutions(b37bc86a-4c3a-41be-b35d-ddfd129276c5)  
HTTP/1.1  
  
--batch_d6010246-cd97-429f-bc05-cf20054cf8e8a  
Content-Type: application/http
```

```
Content-Transfer-Encoding: binary
Content-Length: 121

DELETE /api/data/v9.2/accounts(659876fd-6d29-ed11-9db1-00224804f8e2)
HTTP/1.1
```

```
--batch_d6010246-cd97-429f-bc05-cf20054cf8a
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-Length: 118
```

```
DELETE /api/data/v9.2/roles(669876fd-6d29-ed11-9db1-00224804f8e2) HTTP/1.1
```

```
--batch_d6010246-cd97-429f-bc05-cf20054cf8a--
```

Respuesta:

```
HTTP
```

```
HTTP/1.1 200 OK
OData-Version: 4.0

--batchresponse_cb852192-c300-4ed7-a54f-dc5fc7ee27c3
Content-Type: application/http
Content-Transfer-Encoding: binary
```

```
HTTP/1.1 204 No Content
OData-Version: 4.0
```

```
--batchresponse_cb852192-c300-4ed7-a54f-dc5fc7ee27c3
Content-Type: application/http
Content-Transfer-Encoding: binary
```

```
HTTP/1.1 204 No Content
OData-Version: 4.0
```

```
--batchresponse_cb852192-c300-4ed7-a54f-dc5fc7ee27c3
Content-Type: application/http
Content-Transfer-Encoding: binary
```

```
HTTP/1.1 204 No Content
OData-Version: 4.0
```

```
--batchresponse_cb852192-c300-4ed7-a54f-dc5fc7ee27c3
Content-Type: application/http
Content-Transfer-Encoding: binary
```

```
HTTP/1.1 204 No Content
```

```
OData-Version: 4.0
```

```
--batchresponse_cb852192-c300-4ed7-a54f-dc5fc7ee27c3
```

```
Content-Type: application/http
```

```
Content-Transfer-Encoding: binary
```

```
HTTP/1.1 204 No Content
```

```
OData-Version: 4.0
```

```
--batchresponse_cb852192-c300-4ed7-a54f-dc5fc7ee27c3--
```

Salida de la consola:

```
Deleting created records.
```

Consulte también

[Usar la API web de Dataverse](#)

[Usar funciones de la API web](#)

[Usar acciones de la API web](#)

[Ejemplo de funciones y acciones de la API web \(C#\)](#)

 **Nota**

¿Puede indicarnos sus preferencias de idioma de documentación? [Realice una breve encuesta.](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad](#)).

Ejemplo de operaciones de metadatos de API web

Artículo • 24/08/2023

Esta colección de solicitudes y respuestas http demuestra cómo realizar operaciones seleccionadas que modifican el esquema de Dataverse, o metadatos, usando la [muestra de operaciones de metadatos de API web \(C#\)](#)

En este artículo se describe un conjunto común de operaciones implementadas por cada ejemplo en este grupo. Este artículo describe las solicitudes y respuestas HTTP y la salida de texto que realiza cada muestra sin los detalles específicos del idioma. Consulte las descripciones específicas del idioma y los ejemplos individuales para obtener más información acerca de cómo se realizan estas operaciones.

Demostraciones

Este ejemplo se divide en las siguientes secciones, que contienen operaciones de datos de consulta de la API web de Dataverse que se describen minuciosamente en los artículos conceptuales asociados especificados.

Sección de código	Artículos conceptuales y de referencia asociados
Sección 0: Crear un editor y una solución	Crear una fila de tabla publisher EntityType solucion EntityType
Sección 1: Crear, recuperar y actualizar una tabla	Crear y actualizar definiciones de tablas EntityMetadata EntityType
Sección 2: Crear, recuperar y actualizar columnas - Columna booleana - Valores de opción de actualización - Columna DateTime - Columna decimal - Columnas de enteros - Columna de notas - Columna de moneda - Columna de lista de selección - Agregar una opción al conjunto de opciones local - Opciones para reordenar la columna de opciones - Eliminar valor de opción local	Crear columnas Recuperar atributos Acción InsertOptionValue Acción OrderOption Acción DeleteOptionValue Acción InsertStatusValue

Sección de código	Artículos conceptuales y de referencia asociados
<ul style="list-style-type: none"> - Columna de lista de selección múltiple - Insertar valor de estado 	
Sección 3: Crear y usar OptionSet global	Crear y actualizar opciones (conjuntos de opciones)
Sección 4: Crear relación con el cliente	CreateCustomerRelationships Action
Sección 5: Crear y recuperar una relación de uno a muchos	Idoneidad para relaciones Crear nueva relación de uno a varios Consultar metadatos de relaciones
Sección 6: Crear y recuperar una relación de muchos a uno	Crear nueva relación de uno a varios Consultar metadatos de relaciones
Sección 7: Crear y recuperar una relación de muchos a muchos	Crear nueva relación de varios a varios Consultar metadatos de relaciones
Sección 8: Exportar una solución administrada	Exportar soluciones
Sección 9: Eliminar registros de muestra	Eliminación básica
Sección 10: Importar y eliminar una solución administrada	Importar soluciones

! Nota

Para razones de brevedad, se han omitido los encabezados HTTP menos pertinentes. Las URL de los registros variarán con la dirección de la organización base y los id. establecidos por el servidor de Dataverse.

Sección 0: Crear un editor y una solución

1. Cree primero el editor, ya que la solución debe estar relacionada con él. Todos los elementos creados o modificados en esta muestra utilizan los valores Publisher `customizationprefix` y `customizationoptionvalueprefix`.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/publishers HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
```

```
If-None-Match: null  
Accept: application/json  
  
{  
  "friendlyname": "Example Publisher",  
  "uniquename": "examplepublisher",  
  "description": "An example publisher for samples",  
  "customizationprefix": "sample",  
  "customizationoptionvalueprefix": 72700  
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent  
OData-Version: 4.0  
OData-EntityId: [Organization Uri]/api/data/v9.2/publishers(a78ab7fc-  
102a-ed11-9db1-00224804f8e2)
```

Salida de la consola:

```
Created publisher Example Publisher
```

2. A continuación, cree la solución relacionada con el editor.

(!) Nota

Muchos de los elementos creados o actualizados en este ejemplo utilizarán el valor `uniquename` de esta solución con el encabezado de solicitud `MSCRM.SolutionUniqueName` para que los cambios se incluyan como parte de esta solución. Algunas acciones tienen un parámetro `SolutionUniqueName` que hace lo mismo. Al final de este ejemplo, esta solución se exportará y contendrá las definiciones de todos los elementos creados y modificados en este ejemplo.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/solutions HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0
```

```
If-None-Match: null  
Accept: application/json  
  
{  
  "friendlyname": "Example Solution",  
  "uniquename": "examplesolution",  
  "description": "An example solution for samples",  
  "version": "1.0.0.0",  
  "publisherid@odata.bind": "publishers(a78ab7fc-102a-ed11-9db1-00224804f8e2)"  
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent  
OData-Version: 4.0  
OData-EntityId: [Organization Uri]/api/data/v9.2/solutions(5472b902-112a-ed11-9db1-00224804f8e2)
```

Salida de la consola:

```
Created solution Example Solution
```

Sección 1: Crear, recuperar y actualizar una tabla

1. Cree la tabla `sample_BankAccount`.

Estas propiedades son requeridas: `SchemaName`, `DisplayName`, `DisplayCollectionName`, `HasNotes`, `HasActivities` y `PrimaryNameAttribute`, que deben incluir el valor `LogicalName` de la columna de nombre principal.

La tabla también debe incluir una columna `StringAttributeMetadata` en la colección `Attributes` que sea la columna de nombre principal de la tabla. Esa definición de columna debe tener valores `SchemaName`, `MaxLength` y `DisplayName`, y `IsPrimaryName` debe establecerse en verdadero.

ⓘ Nota

El encabezado de solicitud `MSCRM.SolutionUniqueName: examplesolution` asocia esta tabla a la solución. El valor `SchemaName` (`sample_BankAccount`) incluye el prefijo de personalización del editor.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/EntityDefinitions HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
  "@odata.type": "Microsoft.Dynamics.CRM.EntityMetadata",
  "Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
      {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "A table to store information about customer bank
accounts",
        "LanguageCode": 1033,
        "IsManaged": false
      }
    ],
    "UserLocalizedLabel": {
      "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
      "Label": "A table to store information about customer bank
accounts",
      "LanguageCode": 1033,
      "IsManaged": false
    }
  },
  "DisplayCollectionName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
      {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Bank Accounts",
        "LanguageCode": 1033,
        "IsManaged": false
      }
    ],
    "UserLocalizedLabel": {
      "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
      "Label": "Bank Accounts",
      "LanguageCode": 1033,
      "IsManaged": false
    }
  }
}
```

```
},
"DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Bank Account",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Bank Account",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"HasActivities": false,
"HasNotes": false,
"OwnershipType": "UserOwned",
"PrimaryNameAttribute": "sample_name",
"SchemaName": "sample_BankAccount",
"Attributes": [
    {
        "@odata.type": "Microsoft.Dynamics.CRM.StringAttributeMetadata",
        "AttributeType": "String",
        "AttributeTypeName": {
            "Value": "StringType"
        },
        "MaxLength": 100,
        "Description": {
            "@odata.type": "Microsoft.Dynamics.CRM.Label",
            "LocalizedLabels": [
                {
                    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                    "Label": "The primary attribute for the Bank Account entity.",
                    "LanguageCode": 1033,
                    "IsManaged": false
                }
            ],
            "UserLocalizedLabel": {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "The primary attribute for the Bank Account entity.",
                "LanguageCode": 1033,
                "IsManaged": false
            }
        }
    },
    {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [
            {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Account Name",
            }
        ]
    }
],
```

```

        "LanguageCode": 1033,
        "IsManaged": false
    },
],
"UserLocalizedLabel": {
    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
    "Label": "Account Name",
    "LanguageCode": 1033,
    "IsManaged": false
}
},
"IsPrimaryName": true,
"RequiredLevel": {
    "Value": "None",
    "CanBeChanged": false,
    "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
},
"SchemaName": "sample_Name"
}
]
}

```

Respuesta:

HTTP

```

HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/EntityDefinitions(5872b902-112a-ed11-9db1-
00224804f8e2)

```

Salida de la consola:

```

Sending request to create the sample_BankAccount table...
Created sample_BankAccount table.

```

2. Recupere la definición de tabla `sample_BankAccount`.

- Esta operación de recuperación no incluye ninguna `$select` para filtrar las propiedades devueltas porque estos datos se modifican y se envían para actualizar la definición de la tabla usando `PUT`, que sobrescribe el valor existente.
- Esta consulta tampoco incluye `$expand` para incluir datos relacionados, como atributos, porque los datos relacionados deben actualizarse por separado.

(!) Nota

Esta solicitud y otras en esta muestra usan el encabezado `Consistency: Strong`. Utilice este encabezado cuando recupere los cambios de definición de metadatos directamente después de aplicarlos. Los cambios de metadatos se almacenan en caché por razones de rendimiento y una solicitud de un elemento recién creado puede devolver un 404 porque aún no se ha almacenado en caché. El almacenamiento en caché puede tardar 30 segundos. Este encabezado obligará al servidor a leer la última versión, incluidos sus cambios. Al usar este encabezado, niega la ganancia de rendimiento que proporciona el almacenamiento en caché, por lo que solo debe usarlo en escenarios como este ejemplo donde está recuperando los cambios que acaba de realizar. Más información: [Encabezados HTTP > Otros encabezados](#).

Solicitud:

HTTP

```
GET [Organization  
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')  
HTTP/1.1  
Consistency: Strong  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK  
OData-Version: 4.0  
  
{  
"@odata.context": "[Organization  
Uri]/api/data/v9.2/$metadata#EntityDefinitions/$entity",  
"ActivityTypeMask": 0,  
"AutoRouteToOwnerQueue": false,  
"CanTriggerWorkflow": true,  
"EntityHelpUrlEnabled": false,  
"EntityHelpUrl": null,  
"IsDocumentManagementEnabled": false,  
"IsOneNoteIntegrationEnabled": false,  
"IsInteractionCentricEnabled": false,
```

```
"IsKnowledgeManagementEnabled": false,
"IsSLAEnabled": false,
"IsBPFEntity": false,
"IsDocumentRecommendationsEnabled": false,
"IsMSTeamsIntegrationEnabled": false,
"SettingOf": null,
"DataProviderId": null,
"DataSourceId": null,
"AutoCreateAccessTeams": false,
"IsActivity": false,
"IsActivityParty": false,
"IsRetrieveAuditEnabled": false,
"IsRetrieveMultipleAuditEnabled": false,
"IsArchivalEnabled": false,
"IsAvailableOffline": false,
"IsChildEntity": false,
"IsAIRUpdated": false,
"IconLargeName": null,
"IconMediumName": null,
"IconSmallName": null,
"IconVectorName": null,
"IsCustomEntity": true,
"IsBusinessProcessEnabled": false,
"SyncToExternalSearchIndex": false,
"IsOptimisticConcurrencyEnabled": true,
"ChangeTrackingEnabled": false,
"IsImportable": true,
"IsIntersect": false,
"IsManaged": false,
".IsEnabledForCharts": true,
".IsEnabledForTrace": false,
"IsValidForAdvancedFind": true,
"DaysSinceRecordLastModified": 0,
"MobileOfflineFilters": "",
"IsReadingPaneEnabled": true,
"IsQuickCreateEnabled": false,
"LogicalName": "sample_bankaccount",
"ObjectTypeCode": 10393,
"OwnershipType": "UserOwned",
"PrimaryNameAttribute": "sample_name",
"PrimaryImageAttribute": null,
"PrimaryIdAttribute": "sample_bankaccountid",
"RecurrenceBaseEntityLogicalName": null,
"ReportViewName": "Filteredsample_BankAccount",
"SchemaName": "sample_BankAccount",
"IntroducedVersion": "1.0.0.0",
"IsStateModelAware": true,
"EnforceStateTransitions": false,
"ExternalName": null,
"EntityColor": null,
"LogicalCollectionName": "sample_bankaccounts",
"ExternalCollectionName": null,
"CollectionSchemaName": "sample_BankAccounts",
"EntitySetName": "sample_bankaccounts",
".IsEnabledForExternalChannels": false,
```

```
"IsPrivate": false,
"UsesBusinessDataLabelTable": false,
"IsLogicalEntity": false,
"HasNotes": false,
"HasActivities": false,
"HasFeedback": false,
"IsSolutionAware": false,
"CreatedOn": "2022-09-01T16:13:40Z",
"ModifiedOn": "2022-09-01T16:13:40Z",
"HasEmailAddresses": false,
"OwnerId": null,
"OwnerIdType": 8,
"OwningBusinessUnit": null,
"MetadataId": "5872b902-112a-ed11-9db1-00224804f8e2",
"HasChanged": null,
"Description": {
    "LocalizedLabels": [
        {
            "Label": "A table to store information about customer bank accounts",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "daf026b7-dfde-4b7b-8e52-91f31b098a9d",
            "HasChanged": null
        }
    ],
    "UserLocalizedLabel": {
        "Label": "A table to store information about customer bank accounts",
        "LanguageCode": 1033,
        "IsManaged": false,
        "MetadataId": "daf026b7-dfde-4b7b-8e52-91f31b098a9d",
        "HasChanged": null
    }
},
"DisplayCollectionName": {
    "LocalizedLabels": [
        {
            "Label": "Bank Accounts",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "5c598c79-b89d-4679-8a40-9562d0a1e4fb",
            "HasChanged": null
        }
    ],
    "UserLocalizedLabel": {
        "Label": "Bank Accounts",
        "LanguageCode": 1033,
        "IsManaged": false,
        "MetadataId": "5c598c79-b89d-4679-8a40-9562d0a1e4fb",
        "HasChanged": null
    }
},
"DisplayName": {
    "LocalizedLabels": [
```

```
{
    "Label": "Bank Account",
    "LanguageCode": 1033,
    "IsManaged": false,
    "MetadataId": "4e4c3fdc-7711-4b43-8eba-9155bb7100c0",
    "HasChanged": null
}
],
"UserLocalizedLabel": {
    "Label": "Bank Account",
    "LanguageCode": 1033,
    "IsManaged": false,
    "MetadataId": "4e4c3fdc-7711-4b43-8eba-9155bb7100c0",
    "HasChanged": null
}
},
"IsAuditEnabled": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyauditsettings"
},
"IsValidForQueue": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyqueuesettings"
},
"IsConnectionsEnabled": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyconnectionsettings"
},
"IsCustomizable": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "iscustomizable"
},
"IsRenameable": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "isrenameable"
},
"IsMappable": {
    "Value": true,
    "CanBeChanged": false,
    "ManagedPropertyLogicalName": "ismappable"
},
"IsDuplicateDetectionEnabled": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifieduplicatedetectionsettings"
},
"CanCreateAttributes": {
    "Value": true,
    "CanBeChanged": false,
    "ManagedPropertyLogicalName": "cancreateattributes"
```

```
},
  "CanCreateForms": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "cancreateforms"
  },
  "CanCreateViews": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "cancreateviews"
  },
  "CanCreateCharts": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "cancreatecharts"
  },
  "CanBeRelatedEntityInRelationship": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canberelatedentityinrelationship"
  },
  "CanBePrimaryEntityInRelationship": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canbeprimaryentityinrelationship"
  },
  "CanBeInManyToMany": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canbeinmanytomany"
  },
  "CanBeInCustomEntityAssociation": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canbeincustomentityassociation"
  },
  "CanEnableSyncToExternalSearchIndex": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canenablesynctoexternalsearchindex"
  },
  "CanModifyAdditionalSettings": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyadditionalsettings"
  },
  "CanChangeHierarchicalRelationship": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canchangehierarchicalrelationship"
  },
  "CanChangeTrackingBeEnabled": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canchangetrackingbeenabled"
```

```
},
  "IsMailMergeEnabled": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifymailmergesettings"
  },
  "IsVisibleInMobile": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifymobilevisibility"
  },
  "IsVisibleInMobileClient": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifymobileclientvisibility"
  },
  "IsReadOnlyInMobileClient": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifymobileclientreadonly"
  },
  "IsOfflineInMobileClient": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifymobileclientoffline"
  },
  "Privileges": [
    {
      "CanBeBasic": true,
      "CanBeDeep": true,
      "CanBeGlobal": true,
      "CanBeLocal": true,
      "CanBeEntityReference": false,
      "CanBeParentEntityReference": false,
      "Name": "prvCreatesample_BankAccount",
      "PrivilegeId": "44f00701-716e-4584-8bab-cb0d263c070b",
      "PrivilegeType": "Create"
    },
    {
      "CanBeBasic": true,
      "CanBeDeep": true,
      "CanBeGlobal": true,
      "CanBeLocal": true,
      "CanBeEntityReference": false,
      "CanBeParentEntityReference": false,
      "Name": "prvReadsample_BankAccount",
      "PrivilegeId": "9cad3243-d0fe-467e-a731-c8b3416a6252",
      "PrivilegeType": "Read"
    },
    {
      "CanBeBasic": true,
      "CanBeDeep": true,
      "CanBeGlobal": true,
      "CanBeLocal": true,
      "CanBeEntityReference": false,
```

```
        "CanBeParentEntityReference": false,
        "Name": "prvWritesample_BankAccount",
        "PrivilegeId": "dc5465ed-223f-4b13-a272-fff25e5b5270",
        "PrivilegeType": "Write"
    },
    {
        "CanBeBasic": true,
        "CanBeDeep": true,
        "CanBeGlobal": true,
        "CanBeLocal": true,
        "CanBeEntityReference": false,
        "CanBeParentEntityReference": false,
        "Name": "prvDeletesample_BankAccount",
        "PrivilegeId": "9a409df2-ca4a-4ad9-8218-df88424dd7a0",
        "PrivilegeType": "Delete"
    },
    {
        "CanBeBasic": true,
        "CanBeDeep": true,
        "CanBeGlobal": true,
        "CanBeLocal": true,
        "CanBeEntityReference": false,
        "CanBeParentEntityReference": false,
        "Name": "prvAssignsample_BankAccount",
        "PrivilegeId": "73bf7dd3-f532-4468-abfe-84bbf0eae058",
        "PrivilegeType": "Assign"
    },
    {
        "CanBeBasic": true,
        "CanBeDeep": true,
        "CanBeGlobal": true,
        "CanBeLocal": true,
        "CanBeEntityReference": false,
        "CanBeParentEntityReference": false,
        "Name": "prvSharesample_BankAccount",
        "PrivilegeId": "292f6e27-9603-4835-882d-e28c175432ed",
        "PrivilegeType": "Share"
    },
    {
        "CanBeBasic": true,
        "CanBeDeep": true,
        "CanBeGlobal": true,
        "CanBeLocal": true,
        "CanBeEntityReference": false,
        "CanBeParentEntityReference": false,
        "Name": "prvAppendsample_BankAccount",
        "PrivilegeId": "42401aa6-6447-4fdc-9679-bcb89b62bd76",
        "PrivilegeType": "Append"
    },
    {
        "CanBeBasic": true,
        "CanBeDeep": true,
        "CanBeGlobal": true,
        "CanBeLocal": true,
        "CanBeEntityReference": false,
```

```

        "CanBeParentEntityReference": false,
        "Name": "prvAppendTosample_BankAccount",
        "PrivilegeId": "847ba62d-2f33-4208-87e6-52532b331f60",
        "PrivilegeType": "AppendTo"
    }
],
"Settings": []
}

```

Salida de la consola:

El ejemplo muestra el JSON recuperado del servidor.

3. Actualice la tabla `sample_BankAccount`. Los únicos valores que se modifican son `HasActivities` y `Description`, pero debe enviar la definición completa con `PUT`.

Solicitud:

HTTP

```

PUT [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')
HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
MSCRM.MergeLabels: true
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
"@odata.type": "Microsoft.Dynamics.CRM.EntityMetadata",
"ActivityTypeMask": 0,
"AutoCreateAccessTeams": false,
"AutoRouteToOwnerQueue": false,
"CanBeInCustomEntityAssociation": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canbeincustomentityassociation"
},
"CanBeInManyToMany": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canbeinmanytomany"
},
"CanBePrimaryEntityInRelationship": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canbeprimaryentityinrelationship"
}
}

```

```
},
  "CanBeRelatedEntityInRelationship": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canberelatedentityinrelationship"
  },
  "CanChangeHierarchicalRelationship": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canchangehierarchicalrelationship"
  },
  "CanChangeTrackingBeEnabled": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canchangetrackingbeenabled"
  },
  "CanCreateAttributes": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": true,
    "CanBeChanged": false,
    "ManagedPropertyLogicalName": "cancreateattributes"
  },
  "CanCreateCharts": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "cancreatecharts"
  },
  "CanCreateForms": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "cancreateforms"
  },
  "CanCreateViews": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "cancreateviews"
  },
  "CanEnableSyncToExternalSearchIndex": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canenablesynctoexternalsearchindex"
  },
  "CanModifyAdditionalSettings": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyadditionalsettings"
  }
},
```

```
"CanTriggerWorkflow": true,
"ChangeTrackingEnabled": false,
"CollectionSchemaName": "sample_BankAccounts",
"DaysSinceRecordLastModified": 0,
"Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Contains information about customer bank accounts",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Contains information about customer bank accounts",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"DisplayCollectionName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Bank Accounts",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "5c598c79-b89d-4679-8a40-9562d0a1e4fb"
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Bank Accounts",
        "LanguageCode": 1033,
        "IsManaged": false,
        "MetadataId": "5c598c79-b89d-4679-8a40-9562d0a1e4fb"
    }
},
"DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Bank Account",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "4e4c3fdc-7711-4b43-8eba-9155bb7100c0"
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Bank Account",
        "LanguageCode": 1033,
```

```
        "IsManaged": false,
        "MetadataId": "4e4c3fdc-7711-4b43-8eba-9155bb7100c0"
    },
},
"EnforceStateTransitions": false,
"EntityHelpUrlEnabled": false,
"EntitySetName": "sample_bankaccounts",
"HasActivities": true,
"HasFeedback": false,
"HasNotes": false,
"IntroducedVersion": "1.0.0.0",
"IsActivity": false,
"IsActivityParty": false,
"IsAIRUpdated": false,
"IsAuditEnabled": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyauditsettings"
},
"IsAvailableOffline": false,
"IsBPFEentity": false,
"IsBusinessProcessEnabled": false,
"IsChildEntity": false,
"IsConnectionsEnabled": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyconnectionsettings"
},
"IsCustomEntity": true,
"IsCustomizable": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "iscustomizable"
},
"IsDocumentManagementEnabled": false,
"IsDocumentRecommendationsEnabled": false,
"IsDuplicateDetectionEnabled": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifieduplicatedetectionsettings"
},
"IsEnabledForCharts": true,
"IsEnabledForExternalChannels": false,
"IsEnabledForTrace": false,
"IsImportable": true,
"IsInteractionCentricEnabled": false,
"IsIntersect": false,
"IsKnowledgeManagementEnabled": false,
"IsLogicalEntity": false,
"IsMailMergeEnabled": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
```

```
        "Value": false,
        "CanBeChanged": true,
        "ManagedPropertyLogicalName": "canmodifymailmergesettings"
    },
    "IsManaged": false,
    "IsMappable": {
        "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
        "Value": true,
        "CanBeChanged": false,
        "ManagedPropertyLogicalName": "ismappable"
    },
    "IsMSTeamsIntegrationEnabled": false,
    "IsOfflineInMobileClient": {
        "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
        "Value": false,
        "CanBeChanged": true,
        "ManagedPropertyLogicalName": "canmodifymobileclientoffline"
    },
    "IsOneNoteIntegrationEnabled": false,
    "IsOptimisticConcurrencyEnabled": true,
    "IsPrivate": false,
    "IsQuickCreateEnabled": false,
    "IsReadingPaneEnabled": true,
    "IsReadOnlyInMobileClient": {
        "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
        "Value": false,
        "CanBeChanged": true,
        "ManagedPropertyLogicalName": "canmodifymobileclientreadonly"
    },
    "IsRenameable": {
        "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
        "Value": true,
        "CanBeChanged": true,
        "ManagedPropertyLogicalName": "isrenameable"
    },
    "IsSLAEnabled": false,
    "IsSolutionAware": false,
    "IsStateModelAware": true,
    "IsValidForAdvancedFind": true,
    "IsValidForQueue": {
        "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
        "Value": false,
        "CanBeChanged": true,
        "ManagedPropertyLogicalName": "canmodifyqueuesettings"
    },
    "IsVisibleInMobile": {
        "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
        "Value": false,
        "CanBeChanged": true,
        "ManagedPropertyLogicalName": "canmodifymobilevisibility"
    },
    "IsVisibleInMobileClient": {
        "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
        "Value": false,
        "CanBeChanged": true,
```

```
    "ManagedPropertyLogicalName": "canmodifymobileclientvisibility"
},
"LogicalCollectionName": "sample_bankaccounts",
"LogicalName": "sample_bankaccount",
"MobileOfflineFilters": "",
"ObjectTypeCode": 10393,
"OwnershipType": "UserOwned",
"PrimaryIdAttribute": "sample_bankaccountid",
"PrimaryNameAttribute": "sample_name",
"Privileges": [
{
    "@odata.type":
"Microsoft.Dynamics.CRM.SecurityPrivilegeMetadata",
    "CanBeBasic": true,
    "CanBeDeep": true,
    "CanBeGlobal": true,
    "CanBeLocal": true,
    "CanBeEntityReference": false,
    "CanBeParentEntityReference": false,
    "Name": "prvCreatesample_BankAccount",
    "PrivilegeId": "44f00701-716e-4584-8bab-cb0d263c070b",
    "PrivilegeType": "Create"
},
{
    "@odata.type":
"Microsoft.Dynamics.CRM.SecurityPrivilegeMetadata",
    "CanBeBasic": true,
    "CanBeDeep": true,
    "CanBeGlobal": true,
    "CanBeLocal": true,
    "CanBeEntityReference": false,
    "CanBeParentEntityReference": false,
    "Name": "prvReadsample_BankAccount",
    "PrivilegeId": "9cad3243-d0fe-467e-a731-c8b3416a6252",
    "PrivilegeType": "Read"
},
{
    "@odata.type":
"Microsoft.Dynamics.CRM.SecurityPrivilegeMetadata",
    "CanBeBasic": true,
    "CanBeDeep": true,
    "CanBeGlobal": true,
    "CanBeLocal": true,
    "CanBeEntityReference": false,
    "CanBeParentEntityReference": false,
    "Name": "prvWritessample_BankAccount",
    "PrivilegeId": "dc5465ed-223f-4b13-a272-fff25e5b5270",
    "PrivilegeType": "Write"
},
{
    "@odata.type":
"Microsoft.Dynamics.CRM.SecurityPrivilegeMetadata",
    "CanBeBasic": true,
    "CanBeDeep": true,
    "CanBeGlobal": true,
```

```
    "CanBeLocal": true,
    "CanBeEntityReference": false,
    "CanBeParentEntityReference": false,
    "Name": "prvDeletesample_BankAccount",
    "PrivilegeId": "9a409df2-ca4a-4ad9-8218-df88424dd7a0",
    "PrivilegeType": "Delete"
},
{
    "@odata.type":
"Microsoft.Dynamics.CRM.SecurityPrivilegeMetadata",
    "CanBeBasic": true,
    "CanBeDeep": true,
    "CanBeGlobal": true,
    "CanBeLocal": true,
    "CanBeEntityReference": false,
    "CanBeParentEntityReference": false,
    "Name": "prvAssignsample_BankAccount",
    "PrivilegeId": "73bf7dd3-f532-4468-abfe-84bbf0eae058",
    "PrivilegeType": "Assign"
},
{
    "@odata.type":
"Microsoft.Dynamics.CRM.SecurityPrivilegeMetadata",
    "CanBeBasic": true,
    "CanBeDeep": true,
    "CanBeGlobal": true,
    "CanBeLocal": true,
    "CanBeEntityReference": false,
    "CanBeParentEntityReference": false,
    "Name": "prvSharesample_BankAccount",
    "PrivilegeId": "292f6e27-9603-4835-882d-e28c175432ed",
    "PrivilegeType": "Share"
},
{
    "@odata.type":
"Microsoft.Dynamics.CRM.SecurityPrivilegeMetadata",
    "CanBeBasic": true,
    "CanBeDeep": true,
    "CanBeGlobal": true,
    "CanBeLocal": true,
    "CanBeEntityReference": false,
    "CanBeParentEntityReference": false,
    "Name": "prvAppendsample_BankAccount",
    "PrivilegeId": "42401aa6-6447-4fdc-9679-bcb89b62bd76",
    "PrivilegeType": "Append"
},
{
    "@odata.type":
"Microsoft.Dynamics.CRM.SecurityPrivilegeMetadata",
    "CanBeBasic": true,
    "CanBeDeep": true,
    "CanBeGlobal": true,
    "CanBeLocal": true,
    "CanBeEntityReference": false,
    "CanBeParentEntityReference": false,
```

```
        "Name": "prvAppendTosample_BankAccount",
        "PrivilegeId": "847ba62d-2f33-4208-87e6-52532b331f60",
        "PrivilegeType": "AppendTo"
    },
],
"ReportViewName": "Filteredsample_BankAccount",
"SchemaName": "sample_BankAccount",
"Settings": [],
"SyncToExternalSearchIndex": false,
"UsesBusinessDataLabelTable": false,
"MetadataId": "5872b902-112a-ed11-9db1-00224804f8e2"
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')
```

Salida de la consola:

```
Sending request to update the sample_BankAccount table...
Updated the Bank Account table
```

Sección 2: Crear, recuperar y actualizar columnas

Esta sección crea y recupera un grupo seleccionado de definiciones de columna. Cada uno de estos tipos se derivan de [AttributeMetadata EntityType](#) por lo que comparten la mayoría de las mismas propiedades comunes. Sin embargo, cada tipo derivado tiene algunas propiedades especiales.

Columna booleana

1. Cree una columna booleana usando [BooleanAttributeMetadata EntityType](#). A pesar del nombre, las columnas booleanas tienen una propiedad `OptionSet` al igual que las columnas de elección. Sin embargo, siempre tienen solo dos opciones: `TrueOption` con valor 1 y `FalseOption` con valor 0.

Solicitud:

HTTP

```
POST [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanAttributeMetadata",
    "AttributeType": "Boolean",
    "AttributeName": {
        "Value": "BooleanType"
    },
    "DefaultValue": false,
    "OptionSet": {
        "@odata.type": "Microsoft.Dynamics.CRM.BooleanOptionSetMetadata",
        "TrueOption": {
            "Value": 1,
            "Label": {
                "@odata.type": "Microsoft.Dynamics.CRM.Label",
                "LocalizedLabels": [
                    {
                        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                        "Label": "True",
                        "LanguageCode": 1033,
                        "IsManaged": false
                    }
                ],
                "UserLocalizedLabel": {
                    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                    "Label": "True",
                    "LanguageCode": 1033,
                    "IsManaged": false
                }
            }
        },
        "FalseOption": {
            "Value": 0,
            "Label": {
                "@odata.type": "Microsoft.Dynamics.CRM.Label",
                "LocalizedLabels": [
                    {
                        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                        "Label": "False",
                        "LanguageCode": 1033,
                        "IsManaged": false
                    }
                ],
            }
        }
    }
},
```

```

    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "False",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
{
},
},
"OptionsetType": "Boolean"
},
"Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Boolean Attribute",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Boolean Attribute",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Sample Boolean",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Sample Boolean",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"RequiredLevel": {
    "Value": "None",
    "CanBeChanged": false,
    "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
},
"SchemaName": "sample_Boolean"
}

```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(73f33b3d-112a-ed11-9db1-00224804f8e2)
```

2. Recupere la columna booleana que incluye `$expand=OptionSet` para que se puedan recuperar las opciones.

ⓘ Nota

La URL de esta solicitud incluye

`/Microsoft.Dynamics.CRM.BooleanAttributeMetadata`, que realiza una operación de conversión necesaria para devolver cualquier propiedad que no esté definida en **AttributeMetadata EntityType**. Sin esto, no es posible la expansión de `OptionSet`.

Solicitud:

HTTP

```
GET [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(LogicalName='sample_boolean')/Microsoft.Dynamics.CRM.Boolean
AttributeMetadata?$expand=OptionSet HTTP/1.1
Consistency: Strong
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
"@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#EntityDefinitions('sample_bankaccount')/At
tributes/Microsoft.Dynamics.CRM.BooleanAttributeMetadata(OptionSet())/$
entity",
"MetadataId": "73f33b3d-112a-ed11-9db1-00224804f8e2",
"HasChanged": null,
```

```
"AttributeOf": null,
"AttributeType": "Boolean",
"ColumnNumber": 35,
"DeprecatedVersion": null,
"IntroducedVersion": "1.0.0.0",
"EntityLogicalName": "sample_bankaccount",
"IsCustomAttribute": true,
"IsPrimaryId": false,
"IsValidODataAttribute": true,
"IsPrimaryName": false,
"IsValidForCreate": true,
"IsValidForRead": true,
"IsValidForUpdate": true,
"CanBeSecuredForRead": true,
"CanBeSecuredForCreate": true,
"CanBeSecuredForUpdate": true,
"IsSecured": false,
"IsRetrievable": false,
"IsFilterable": false,
"IsSearchable": false,
"IsManaged": false,
"LinkedAttributeId": null,
"LogicalName": "sample_boolean",
"IsValidForForm": true,
"IsRequiredForForm": false,
"IsValidForGrid": true,
"SchemaName": "sample_Boolean",
"ExternalName": null,
"IsLogical": false,
"IsDataSourceSecret": false,
"InheritsFrom": null,
"CreatedOn": "2022-09-01T16:15:08Z",
"ModifiedOn": "2022-09-01T16:15:08Z",
"SourceType": 0,
"AutoNumberFormat": null,
"DefaultValue": false,
"FormulaDefinition": "",
"SourceTypeMask": 0,
"AttributeTypeName": {
    "Value": "BooleanType"
},
"Description": {
    "LocalizedLabels": [
        {
            "Label": "Boolean Attribute",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "ea50b52d-53e4-4f8d-82ce-8f74a7554800",
            "HasChanged": null
        }
    ],
    "UserLocalizedLabel": {
        "Label": "Boolean Attribute",
        "LanguageCode": 1033,
        "IsManaged": false,
```

```
        "MetadataId": "ea50b52d-53e4-4f8d-82ce-8f74a7554800",
        "HasChanged": null
    },
},
"DisplayName": {
    "LocalizedLabels": [
        {
            "Label": "Sample Boolean",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "9e4daa21-8774-4de9-b467-d046389459dc",
            "HasChanged": null
        }
    ],
    "UserLocalizedLabel": {
        "Label": "Sample Boolean",
        "LanguageCode": 1033,
        "IsManaged": false,
        "MetadataId": "9e4daa21-8774-4de9-b467-d046389459dc",
        "HasChanged": null
    }
},
"IsAuditEnabled": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyauditsettings"
},
"IsGlobalFilterEnabled": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyglobalfiltersettings"
},
"IsSortableEnabled": {
    "Value": false,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyissortablesettings"
},
"IsCustomizable": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "iscustomizable"
},
"IsRenameable": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "isrenameable"
},
"IsValidForAdvancedFind": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifysearchsettings"
},
"RequiredLevel": {
    "Value": "None",
    "CanBeChanged": false,
```

```
        "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
    },
    "CanModifyAdditionalSettings": {
        "Value": true,
        "CanBeChanged": true,
        "ManagedPropertyLogicalName": "canmodifyadditionalsettings"
    },
    "Settings": [],
    "OptionSet": {
        "MetadataId": "74f33b3d-112a-ed11-9db1-00224804f8e2",
        "HasChanged": null,
        "IsCustomOptionSet": true,
        "IsGlobal": false,
        "IsManaged": false,
        "Name": "sample_bankaccount_sample_boolean",
        "ExternalTypeName": null,
        "OptionsetType": "Boolean",
        "IntroducedVersion": "1.0.0.0",
        "Description": {
            "LocalizedLabels": [
                {
                    "Label": "Boolean Attribute",
                    "LanguageCode": 1033,
                    "IsManaged": false,
                    "MetadataId": "76f33b3d-112a-ed11-9db1-00224804f8e2",
                    "HasChanged": null
                }
            ],
            "UserLocalizedLabel": {
                "Label": "Boolean Attribute",
                "LanguageCode": 1033,
                "IsManaged": false,
                "MetadataId": "76f33b3d-112a-ed11-9db1-00224804f8e2",
                "HasChanged": null
            }
        },
        "DisplayName": {
            "LocalizedLabels": [
                {
                    "Label": "Sample Boolean",
                    "LanguageCode": 1033,
                    "IsManaged": false,
                    "MetadataId": "75f33b3d-112a-ed11-9db1-00224804f8e2",
                    "HasChanged": null
                }
            ],
            "UserLocalizedLabel": {
                "Label": "Sample Boolean",
                "LanguageCode": 1033,
                "IsManaged": false,
                "MetadataId": "75f33b3d-112a-ed11-9db1-00224804f8e2",
                "HasChanged": null
            }
        },
        "IsCustomizable": {
```

```
        "Value": true,
        "CanBeChanged": true,
        "ManagedPropertyLogicalName": "iscustomizable"
    },
    "TrueOption": {
        "Value": 1,
        "Color": null,
        "IsManaged": false,
        "ExternalValue": "",
        "ParentValues": [],
        "MetadataId": null,
        "HasChanged": null,
        "Label": {
            "LocalizedLabels": [
                {
                    "Label": "True",
                    "LanguageCode": 1033,
                    "IsManaged": false,
                    "MetadataId": "12049c5f-e99d-453f-8315-3933512539a1",
                    "HasChanged": null
                }
            ],
            "UserLocalizedLabel": {
                "Label": "True",
                "LanguageCode": 1033,
                "IsManaged": false,
                "MetadataId": "12049c5f-e99d-453f-8315-3933512539a1",
                "HasChanged": null
            }
        },
        "Description": {
            "LocalizedLabels": [],
            "UserLocalizedLabel": null
        }
    },
    "FalseOption": {
        "Value": 0,
        "Color": null,
        "IsManaged": false,
        "ExternalValue": "",
        "ParentValues": [],
        "MetadataId": null,
        "HasChanged": null,
        "Label": {
            "LocalizedLabels": [
                {
                    "Label": "False",
                    "LanguageCode": 1033,
                    "IsManaged": false,
                    "MetadataId": "e3d4c2b1-ad54-4d3a-8e01-f759da0e476f",
                    "HasChanged": null
                }
            ],
            "UserLocalizedLabel": {
                "Label": "False",

```

```

        "LanguageCode": 1033,
        "IsManaged": false,
        "MetadataId": "e3d4c2b1-ad54-4d3a-8e01-f759da0e476f",
        "HasChanged": null
    }
},
"Description": {
    "LocalizedLabels": [],
    "UserLocalizedLabel": null
}
}
}
}

```

Salida de la consola:

```

Original Option Labels:
True Option Label:'True' Value: 1
False Option Label:'False' Value: 0

```

3. Actualice la columna booleana. Los únicos cambios son en las propiedades

`DisplayName`, `Description` y `RequiredLevel`, pero se incluye toda la definición porque se usa `PUT`.

ⓘ Nota

A pesar de que la propiedad `OptionSet` está incluida en esta carga útil, los cambios en las opciones no se aplicarán porque no se consideran parte de la definición de la columna. Deben actualizarse por separado y este ejemplo le mostrará cómo hacerlo en los siguientes pasos.

Solicitud:

HTTP

```

PUT [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(LogicalName='sample_boolean') HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
MSCRM.MergeLabels: true
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

```

```
{
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanAttributeMetadata",
    "AttributeType": "Boolean",
    "AttributeName": {
        "Value": "BooleanType"
    },
    "DefaultValue": false,
    "FormulaDefinition": "",
    "SourceTypeMask": 0,
    "OptionSet": {
        "@odata.type": "Microsoft.Dynamics.CRM.BooleanOptionSetMetadata",
        "TrueOption": {
            "Value": 1,
            "Label": {
                "@odata.type": "Microsoft.Dynamics.CRM.Label",
                "LocalizedLabels": [
                    {
                        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                        "Label": "True",
                        "LanguageCode": 1033,
                        "IsManaged": false,
                        "MetadataId": "12049c5f-e99d-453f-8315-3933512539a1"
                    }
                ],
                "UserLocalizedLabel": {
                    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                    "Label": "True",
                    "LanguageCode": 1033,
                    "IsManaged": false,
                    "MetadataId": "12049c5f-e99d-453f-8315-3933512539a1"
                }
            }
        },
        "FalseOption": {
            "Value": 0,
            "Label": {
                "@odata.type": "Microsoft.Dynamics.CRM.Label",
                "LocalizedLabels": [
                    {
                        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                        "Label": "False",
                        "LanguageCode": 1033,
                        "IsManaged": false,
                        "MetadataId": "e3d4c2b1-ad54-4d3a-8e01-f759da0e476f"
                    }
                ]
            }
        }
    },
    "IsManaged": false,
    "ExternalValue": ""
},
```

```
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "False",
        "LanguageCode": 1033,
        "IsManaged": false,
        "MetadataId": "e3d4c2b1-ad54-4d3a-8e01-f759da0e476f"
    }
},
{
    "Description": {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [],
        "UserLocalizedLabel": {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "LanguageCode": 0,
            "IsManaged": false
        }
    },
    "IsManaged": false,
    "ExternalValue": ""
},
{
    "OptionsetType": "Boolean",
    "Description": {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [
            {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Boolean Attribute",
                "LanguageCode": 1033,
                "IsManaged": false,
                "MetadataId": "76f33b3d-112a-ed11-9db1-00224804f8e2"
            }
        ],
        "UserLocalizedLabel": {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Boolean Attribute",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "76f33b3d-112a-ed11-9db1-00224804f8e2"
        }
    }
},
{
    "DisplayName": {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [
            {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Sample Boolean",
                "LanguageCode": 1033,
                "IsManaged": false,
                "MetadataId": "75f33b3d-112a-ed11-9db1-00224804f8e2"
            }
        ],
        "UserLocalizedLabel": {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Sample Boolean",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "75f33b3d-112a-ed11-9db1-00224804f8e2"
        }
    }
}
```

```
        "Label": "Sample Boolean",
        "LanguageCode": 1033,
        "IsManaged": false,
        "MetadataId": "75f33b3d-112a-ed11-9db1-00224804f8e2"
    }
},
"IntroducedVersion": "1.0.0.0",
"IsCustomizable": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "iscustomizable"
},
"IsCustomOptionSet": true,
"IsGlobal": false,
"IsManaged": false,
"Name": "sample_bankaccount_sample_boolean",
"MetadataId": "74f33b3d-112a-ed11-9db1-00224804f8e2"
},
"CanBeSecuredForCreate": true,
"CanBeSecuredForRead": true,
"CanBeSecuredForUpdate": true,
"CanModifyAdditionalSettings": {
    "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "canmodifyadditionalsettings"
},
"ColumnNumber": 35,
"Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Boolean Attribute Updated",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Boolean Attribute Updated",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Sample Boolean Updated",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ]
}
```

```
],
  "UserLocalizedLabel": {
    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
    "Label": "Sample Boolean Updated",
    "LanguageCode": 1033,
    "IsManaged": false
  }
},
"EntityLogicalName": "sample_bankaccount",
"IntroducedVersion": "1.0.0.0",
"IsAuditEnabled": {
  "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
  "Value": true,
  "CanBeChanged": true,
  "ManagedPropertyLogicalName": "canmodifyauditsettings"
},
"IsCustomAttribute": true,
"IsCustomizable": {
  "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
  "Value": true,
  "CanBeChanged": true,
  "ManagedPropertyLogicalName": "iscustomizable"
},
"IsDataSourceSecret": false,
"IsFilterable": false,
"IsGlobalFilterEnabled": {
  "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
  "Value": false,
  "CanBeChanged": true,
  "ManagedPropertyLogicalName": "canmodifyglobalfiltersettings"
},
"IsLogical": false,
"IsManaged": false,
"IsPrimaryId": false,
"IsPrimaryName": false,
"IsRenameable": {
  "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
  "Value": true,
  "CanBeChanged": true,
  "ManagedPropertyLogicalName": "isrenameable"
},
"IsRequiredForForm": false,
"IsRetrievable": false,
"IsSearchable": false,
"IsSecured": false,
"IsSortableEnabled": {
  "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
  "Value": false,
  "CanBeChanged": true,
  "ManagedPropertyLogicalName": "canmodifyissortablesettings"
},
"IsValidForAdvancedFind": {
  "@odata.type": "Microsoft.Dynamics.CRM.BooleanManagedProperty",
  "Value": true,
  "CanBeChanged": true,
```

```
        "ManagedPropertyLogicalName": "canmodifysearchsettings"
    },
    "IsValidForCreate": true,
    "IsValidForForm": true,
    "IsValidForGrid": true,
    "IsValidForRead": true,
    "IsValidForUpdate": true,
    "LogicalName": "sample_boolean",
    "RequiredLevel": {
        "Value": "ApplicationRequired",
        "CanBeChanged": false,
        "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
    },
    "SchemaName": "sample_Boolean",
    "SourceType": 0,
    "MetadataId": "73f33b3d-112a-ed11-9db1-00224804f8e2"
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(LogicalName='sample_boolean')
```

Salida de la consola:

```
Updated Boolean Column properties
```

Valores de opción de actualización

Actualice cada una de las opciones booleanas usando [UpdateOptionValue Action](#).

ⓘ Nota

Aquí estamos aplicando cambios a las opciones en un atributo booleano, pero usará [UpdateOptionValue](#) para opciones en cualquier tipo de columna que las use excepto columnas `status`, donde debe utilizar [UpdateStateValue Action](#).

1. Cambie el valor de la etiqueta `TrueOption` a 'Arriba'.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/UpdateOptionValue HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
  "AttributeLogicalName": "sample_boolean",
  "EntityLogicalName": "sample_bankaccount",
  "Value": 1,
  "Label": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
      {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Up",
        "LanguageCode": 1033,
        "IsManaged": false
      }
    ],
    "UserLocalizedLabel": {
      "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
      "Label": "Up",
      "LanguageCode": 1033,
      "IsManaged": false
    }
  },
  "MergeLabels": true
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
```

2. Cambie el valor de la etiqueta `FalseOption` a 'Abajo'.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/UpdateOptionValue HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
```

```
Accept: application/json
```

```
{
  "AttributeLogicalName": "sample_boolean",
  "EntityLogicalName": "sample_bankaccount",
  "Value": 0,
  "Label": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
      {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Down",
        "LanguageCode": 1033,
        "IsManaged": false
      }
    ],
    "UserLocalizedLabel": {
      "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
      "Label": "Down",
      "LanguageCode": 1033,
      "IsManaged": false
    }
  },
  "MergeLabels": true
}
```

Respuesta:

```
HTTP
```

```
HTTP/1.1 204 NoContent
```

```
OData-Version: 4.0
```

Salida de la consola:

```
Updated option labels
```

3. Recupere los valores de opción modificados para la columna booleana usando la misma consulta que antes:

Salida de la consola:

```
Updated Option Labels:
Updated True Option Label:'Up' Value: 1
Updated False Option Label:'Down' Value: 0
```

Columna DateTime

1. Cree una columna DateTime mediante [DateTimeAttributeMetadata EntityType](#).

Solicitud:

HTTP

```
POST [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
  "@odata.type": "Microsoft.Dynamics.CRM.DateTimeAttributeMetadata",
  "AttributeType": "DateTime",
  "AttributeTypeName": {
    "Value": "DateTimeType"
  },
  "Format": "DateOnly",
  "ImeMode": "Disabled",
  "DateTimeBehavior": {
    "Value": "DateOnly"
  },
  "Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
      {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "DateTime Attribute",
        "LanguageCode": 1033,
        "IsManaged": false
      }
    ],
    "UserLocalizedLabel": {
      "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
      "Label": "DateTime Attribute",
      "LanguageCode": 1033,
      "IsManaged": false
    }
  },
  "DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
      {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Sample DateTime",
        "LanguageCode": 1033,
        "IsManaged": false
      }
    ]
  }
}
```

```
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Sample DateTime",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"RequiredLevel": {
    "Value": "None",
    "CanBeChanged": false,
    "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
},
"SchemaName": "sample_DateTime"
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(f1db3d43-112a-ed11-9db1-00224804f8e2)
```

Salida de la consola:

```
Created DateTime column with id:f1db3d43-112a-ed11-9db1-00224804f8e2
```

2. Recupere los valores seleccionados de la columna DateTime.

Solicitud:

HTTP

```
GET [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(LogicalName='sample_datetime')/Microsoft.Dynamics.CRM.DateTime
meAttributeMetadata?$select=SchemaName,Format,DateTimeBehavior HTTP/1.1
Consistency: Strong
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#EntityDefinitions('sample_bankaccount')/At
tributes/Microsoft.Dynamics.CRM.DateTimeAttributeMetadata(SchemaName,Fo
rmat,DateTimeBehavior)/$entity",
  "SchemaName": "sample_DateTime",
  "Format": "DateOnly",
  "MetadataId": "f1db3d43-112a-ed11-9db1-00224804f8e2",
  "DateTimeBehavior": {
    "Value": "DateOnly"
  }
}
```

Salida de la consola:

```
Retrieved Datetime column properties:
  DateTime Format:'DateOnly'
  DateTime DateTimeBehavior:'DateOnly'
```

Columna decimal

1. Cree una columna Decimal usando [DecimalAttributeMetadata EntityType](#).

Solicitud:

HTTP

```
POST [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
  "@odata.type": "Microsoft.Dynamics.CRM.DecimalAttributeMetadata",
  "AttributeType": "Decimal",
  "AttributeTypeName": {
```

```

    "Value": "DecimalType"
},
".MaxValue": 100.0,
".MinValue": 0.0,
"Precision": 1,
"Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Decimal Attribute",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Decimal Attribute",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Sample Decimal",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Sample Decimal",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"RequiredLevel": {
    "Value": "None",
    "CanBeChanged": false,
    "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
},
"SchemaName": "sample_Decimal"
}

```

Respuesta:

HTTP

HTTP/1.1 204 NoContent
OData-Version: 4.0

```
OData-EntityId: [Organization  
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/  
Attributes(f2db3d43-112a-ed11-9db1-00224804f8e2)
```

Salida de la consola:

```
Created Decimal column with id:f2db3d43-112a-ed11-9db1-00224804f8e2
```

2. Recupere los valores seleccionados de la columna Decimal.

Solicitud:

HTTP

```
GET [Organization  
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/  
Attributes(LogicalName='sample_decimal')/Microsoft.Dynamics.CRM.Decimal  
AttributeMetadata?$select=SchemaName,.MaxValue,.MinValue,Precision  
HTTP/1.1  
Consistency: Strong  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK  
OData-Version: 4.0  
  
{  
"@odata.context": "[Organization  
Uri]/api/data/v9.2/$metadata#EntityDefinitions('sample_bankaccount')/At  
tributes/Microsoft.Dynamics.CRM.DecimalAttributeMetadata(SchemaName,Max  
Value,MinValue,Precision)/$entity",  
"SchemaName": "sample.Decimal",  
".MaxValue": 100,  
".MinValue": 0,  
"Precision": 1,  
"MetadataId": "f2db3d43-112a-ed11-9db1-00224804f8e2"  
}
```

Salida de la consola:

```
Retrieved Decimal column properties:  
Decimal MaxValue:100  
Decimal MinValue:0  
Decimal Precision:1
```

Columnas de enteros

1. Cree una columna de enteros usando [IntegerAttributeMetadata EntityType](#).

Solicitud:

HTTP

```
POST [Organization  
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/  
Attributes HTTP/1.1  
MSCRM.SolutionUniqueName: examplesolution  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json  
  
{  
    "@odata.type": "Microsoft.Dynamics.CRM.IntegerAttributeMetadata",  
    "AttributeType": "Integer",  
    "AttributeTypeName": {  
        "Value": "IntegerType"  
    },  
    ".MaxValue": 100,  
    ".MinValue": 0,  
    "Format": "None",  
    "SourceTypeMask": 0,  
    "Description": {  
        "@odata.type": "Microsoft.Dynamics.CRM.Label",  
        "LocalizedLabels": [  
            {  
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",  
                "Label": "Integer Attribute",  
                "LanguageCode": 1033,  
                "IsManaged": false  
            }  
        ],  
        "UserLocalizedLabel": {  
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",  
            "Label": "Integer Attribute",  
            "LanguageCode": 1033,  
            "IsManaged": false  
        }  
    },  
    "DisplayName": {  
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
```

```
"LocalizedLabels": [
  {
    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
    "Label": "Sample Integer",
    "LanguageCode": 1033,
    "IsManaged": false
  }
],
"UserLocalizedLabel": {
  "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
  "Label": "Sample Integer",
  "LanguageCode": 1033,
  "IsManaged": false
}
},
"RequiredLevel": {
  "Value": "None",
  "CanBeChanged": false,
  "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
},
"SchemaName": "sample_Integer"
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(f5db3d43-112a-ed11-9db1-00224804f8e2)
```

Salida de la consola:

```
Created Integer column with id:f5db3d43-112a-ed11-9db1-00224804f8e2
```

2. Recupere los valores seleccionados de la columna de enteros.

Solicitud:

HTTP

```
GET [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(LogicalName='sample_integer')/Microsoft.Dynamics.CRM.Integer
AttributeMetadata?$select=SchemaName,.MaxValue,.MinValue,Format HTTP/1.1
Consistency: Strong
```

```
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#EntityDefinitions('sample_bankaccount')/At
tributes/Microsoft.Dynamics.CRM.IntegerAttributeMetadata(SchemaName,Max
Value,MinValue,Format)/$entity",
  "SchemaName": "sample_Integer",
  ".MaxValue": 100,
  ".MinValue": 0,
  "Format": "None",
  "MetadataId": "f5db3d43-112a-ed11-9db1-00224804f8e2"
}
```

Salida de la consola:

```
Retrieved Integer column properties:
Integer MaxValue:100
Integer MinValue:0
Integer Format:None
```

Columna de notas

1. Cree una columna de notas usando [MemoAttributeMetadata EntityType](#).

Solicitud:

HTTP

```
POST [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

```
{
  "@odata.type": "Microsoft.Dynamics.CRM.MemoAttributeMetadata",
  "AttributeType": "Memo",
  "AttributeName": {
    "Value": "MemoType"
  },
  "Format": "TextArea",
  "ImeMode": "Disabled",
  "MaxLength": 500,
  "IsLocalizable": false,
  "Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
      {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Memo Attribute",
        "LanguageCode": 1033,
        "IsManaged": false
      }
    ],
    "UserLocalizedLabel": {
      "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
      "Label": "Memo Attribute",
      "LanguageCode": 1033,
      "IsManaged": false
    }
  },
  "DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
      {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Sample Memo",
        "LanguageCode": 1033,
        "IsManaged": false
      }
    ],
    "UserLocalizedLabel": {
      "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
      "Label": "Sample Memo",
      "LanguageCode": 1033,
      "IsManaged": false
    }
  },
  "RequiredLevel": {
    "Value": "None",
    "CanBeChanged": false,
    "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
  },
  "SchemaName": "sample_Memo"
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(f6db3d43-112a-ed11-9db1-00224804f8e2)
```

Salida de la consola:

```
Created Memo column with id:f6db3d43-112a-ed11-9db1-00224804f8e2
```

2. Recupere los valores seleccionados de la columna de notas.

Solicitud:

HTTP

```
GET [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(LogicalName='sample_memo')/Microsoft.Dynamics.CRM.MemoAttributeMetadata?$select=SchemaName,Format,ImeMode,MaxLength HTTP/1.1
Consistency: Strong
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
"@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#EntityDefinitions('sample_bankaccount')/Attributes/Microsoft.Dynamics.CRM.MemoAttributeMetadata(SchemaName,Format
,ImeMode,MaxLength)/$entity",
"SchemaName": "sample_Memo",
"Format": "TextArea",
"ImeMode": "Disabled",
"MaxLength": 500,
"MetadataId": "f6db3d43-112a-ed11-9db1-00224804f8e2"
}
```

Salida de la consola:

```
Retrieved Memo column properties:  
Memo Format:TextArea  
Memo ImeMode:Disabled  
Memo MaxLength:500
```

Columna de moneda

1. Crear una columna de moneda mediante [MoneyAttributeMetadata EntityType](#).

Solicitud:

HTTP

```
POST [Organization  
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/  
Attributes HTTP/1.1  
MSCRM.SolutionUniqueName: examplesolution  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json  
  
{  
    "@odata.type": "Microsoft.Dynamics.CRM.MoneyAttributeMetadata",  
    "AttributeType": "Money",  
    "AttributeTypeName": {  
        "Value": "MoneyType"  
    },  
    "ImeMode": "Disabled",  
    ".MaxValue": 1000.0,  
    ".MinValue": 0.0,  
    "Precision": 1,  
    "PrecisionSource": 1,  
    "SourceTypeMask": 0,  
    "IsBaseCurrency": false,  
    "Description": {  
        "@odata.type": "Microsoft.Dynamics.CRM.Label",  
        "LocalizedLabels": [  
            {  
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",  
                "Label": "Money Attribute",  
                "LanguageCode": 1033,  
                "IsManaged": false  
            }  
        ],  
        "UserLocalizedLabel": {  
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",  
            "Label": "Moneda",  
            "LanguageCode": 1033,  
            "IsManaged": false  
        }  
    }  
}
```

```

        "Label": "Money Attribute",
        "LanguageCode": 1033,
        "IsManaged": false
    },
},
"DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Sample Money",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Sample Money",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"RequiredLevel": {
    "Value": "None",
    "CanBeChanged": false,
    "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
},
"SchemaName": "sample_Money"
}

```

Respuesta:

HTTP

```

HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(fddb3d43-112a-ed11-9db1-00224804f8e2)

```

Salida de la consola:

```
Created Money column with id:fddb3d43-112a-ed11-9db1-00224804f8e2
```

2. Recupere los valores seleccionados de la columna de moneda.

Solicitud:

HTTP

```
GET [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(LogicalName='sample_money')/Microsoft.Dynamics.CRM.MoneyAttr
ibuteMetadata?
$select=SchemaName,.MaxValue,.MinValue,Precision,PrecisionSource,ImeMode
HTTP/1.1
Consistency: Strong
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#EntityDefinitions('sample_bankaccount')/At
tributes/Microsoft.Dynamics.CRM.MoneyAttributeMetadata(SchemaName,MaxVa
lue,MinValue,Precision,PrecisionSource,ImeMode)/$entity",
  "SchemaName": "sample_Money",
  ".MaxValue": 1000.0,
  ".MinValue": 0.0,
  "Precision": 1,
  "PrecisionSource": 1,
  "ImeMode": "Disabled",
  "MetadataId": "fdedb3d43-112a-ed11-9db1-00224804f8e2"
}
```

Salida de la consola:

```
Retrieved Money column properties:
Money MaxValue:1000
Money MinValue:0
Money Precision:1
Money PrecisionSource:1
Money ImeMode:Disabled
```

Columna de lista de selección

1. Cree una columna de opciones (lista de selección) usando [PicklistAttributeMetadata EntityType](#) con un conjunto de opciones local.

Solicitud:

HTTP

```
POST [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
"@odata.type": "Microsoft.Dynamics.CRM.PicklistAttributeMetadata",
"AttributeType": "Picklist",
"AttributeName": {
    "Value": "PicklistType"
},
"SourceTypeMask": 0,
"OptionSet": {
    "@odata.type": "Microsoft.Dynamics.CRM.OptionSetMetadata",
    "Options": [
        {
            "Value": 727000000,
            "Label": {
                "@odata.type": "Microsoft.Dynamics.CRM.Label",
                "LocalizedLabels": [
                    {
                        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                        "Label": "Bravo",
                        "LanguageCode": 1033,
                        "IsManaged": false
                    }
                ],
                "UserLocalizedLabel": {
                    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                    "Label": "Bravo",
                    "LanguageCode": 1033,
                    "IsManaged": false
                }
            }
        },
        {
            "Value": 727000001,
            "Label": {
                "@odata.type": "Microsoft.Dynamics.CRM.Label",
                "LocalizedLabels": [
                    {
                        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                        "Label": "Bravo"
                    }
                ]
            }
        }
    ]
}
```

```
        "Label": "Delta",
        "LanguageCode": 1033,
        "IsManaged": false
    }
],
"UserLocalizedLabel": {
    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
    "Label": "Delta",
    "LanguageCode": 1033,
    "IsManaged": false
}
},
{
"Value": 727000002,
"Label": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Alpha",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Alpha",
        "LanguageCode": 1033,
        "IsManaged": false
    }
}
},
{
"Value": 727000003,
"Label": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Charlie",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Charlie",
        "LanguageCode": 1033,
        "IsManaged": false
    }
}
},
{
"Value": 727000004,
```

```
"Label": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Foxtrot",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Foxtrot",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"IsGlobal": false,
"OptionsetType": "Picklist"
},
"Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Choice Attribute",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Choice Attribute",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Sample Choice",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Sample Choice",
        "LanguageCode": 1033,
        "IsManaged": false
    }
}
```

```
},
"RequiredLevel": {
    "Value": "None",
    "CanBeChanged": false,
    "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
},
"SchemaName": "sample_Choice"
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(4a154e49-112a-ed11-9db1-00224804f8e2)
```

Salida de la consola:

```
Created Choice column with id:4a154e49-112a-ed11-9db1-00224804f8e2
```

2. Recuperar opciones de la columna de elección usando

```
$select=SchemaName&$expand=OptionSet.
```

Solicitud:

HTTP

```
GET [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(LogicalName='sample_choice')/Microsoft.Dynamics.CRM.Picklist
AttributeMetadata?$select=SchemaName&$expand=OptionSet HTTP/1.1
Consistency: Strong
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0
```

```
{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#EntityDefinitions('sample_bankaccount')/At
tributes/Microsoft.Dynamics.CRM.PicklistAttributeMetadata(SchemaName,Op
tionSet())/$entity",
  "SchemaName": "sample_Choice",
  "MetadataId": "4a154e49-112a-ed11-9db1-00224804f8e2",
  "OptionSet": {
    "MetadataId": "4b154e49-112a-ed11-9db1-00224804f8e2",
    "HasChanged": null,
    "IsCustomOptionSet": true,
    "IsGlobal": false,
    "IsManaged": false,
    "Name": "sample_bankaccount_sample_choice",
    "ExternalTypeName": null,
    "OptionsetType": "Picklist",
    "IntroducedVersion": "1.0.0.0",
    "ParentOptionSetName": null,
    "Description": {
      "LocalizedLabels": [
        {
          "Label": "Choice Attribute",
          "LanguageCode": 1033,
          "IsManaged": false,
          "MetadataId": "4d154e49-112a-ed11-9db1-00224804f8e2",
          "HasChanged": null
        }
      ],
      "UserLocalizedLabel": {
        "Label": "Choice Attribute",
        "LanguageCode": 1033,
        "IsManaged": false,
        "MetadataId": "4d154e49-112a-ed11-9db1-00224804f8e2",
        "HasChanged": null
      }
    },
    "DisplayName": {
      "LocalizedLabels": [
        {
          "Label": "Sample Choice",
          "LanguageCode": 1033,
          "IsManaged": false,
          "MetadataId": "4c154e49-112a-ed11-9db1-00224804f8e2",
          "HasChanged": null
        }
      ],
      "UserLocalizedLabel": {
        "Label": "Sample Choice",
        "LanguageCode": 1033,
        "IsManaged": false,
        "MetadataId": "4c154e49-112a-ed11-9db1-00224804f8e2",
        "HasChanged": null
      }
    }
  }
},
```

```
"IsCustomizable": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "iscustomizable"
},
"Options": [
    {
        "Value": 727000000,
        "Color": null,
        "IsManaged": false,
        "ExternalValue": "",
        "ParentValues": [],
        "MetadataId": null,
        "HasChanged": null,
        "Label": {
            "LocalizedLabels": [
                {
                    "Label": "Bravo",
                    "LanguageCode": 1033,
                    "IsManaged": false,
                    "MetadataId": "bc8d1815-75b7-4c13-b618-7959aaf4abb6",
                    "HasChanged": null
                }
            ],
            "UserLocalizedLabel": {
                "Label": "Bravo",
                "LanguageCode": 1033,
                "IsManaged": false,
                "MetadataId": "bc8d1815-75b7-4c13-b618-7959aaf4abb6",
                "HasChanged": null
            }
        },
        "Description": {
            "LocalizedLabels": [],
            "UserLocalizedLabel": null
        }
    },
    {
        "Value": 727000001,
        "Color": null,
        "IsManaged": false,
        "ExternalValue": "",
        "ParentValues": [],
        "MetadataId": null,
        "HasChanged": null,
        "Label": {
            "LocalizedLabels": [
                {
                    "Label": "Delta",
                    "LanguageCode": 1033,
                    "IsManaged": false,
                    "MetadataId": "c3613791-85a0-41ac-8575-91aca4bb91e8",
                    "HasChanged": null
                }
            ],
            "UserLocalizedLabel": {
                "Label": "Delta",
                "LanguageCode": 1033,
                "IsManaged": false,
                "MetadataId": "c3613791-85a0-41ac-8575-91aca4bb91e8",
                "HasChanged": null
            }
        }
    }
],
```

```
        "UserLocalizedLabel": {
            "Label": "Delta",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "c3613791-85a0-41ac-8575-91aca4bb91e8",
            "HasChanged": null
        }
    },
    "Description": {
        "LocalizedLabels": [],
        "UserLocalizedLabel": null
    }
},
{
    "Value": 727000002,
    "Color": null,
    "IsManaged": false,
    "ExternalValue": "",
    "ParentValues": [],
    "MetadataId": null,
    "HasChanged": null,
    "Label": {
        "LocalizedLabels": [
            {
                "Label": "Alpha",
                "LanguageCode": 1033,
                "IsManaged": false,
                "MetadataId": "8db04562-9ec3-4014-a170-0482bbb94e44",
                "HasChanged": null
            }
        ],
        "UserLocalizedLabel": {
            "Label": "Alpha",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "8db04562-9ec3-4014-a170-0482bbb94e44",
            "HasChanged": null
        }
    },
    "Description": {
        "LocalizedLabels": [],
        "UserLocalizedLabel": null
    }
},
{
    "Value": 727000003,
    "Color": null,
    "IsManaged": false,
    "ExternalValue": "",
    "ParentValues": [],
    "MetadataId": null,
    "HasChanged": null,
    "Label": {
        "LocalizedLabels": [
            {

```

```
        "Label": "Charlie",
        "LanguageCode": 1033,
        "IsManaged": false,
        "MetadataId": "d00dc11e-ed91-478b-ac78-86b6784326ad",
        "HasChanged": null
    }
],
"UserLocalizedLabel": {
    "Label": "Charlie",
    "LanguageCode": 1033,
    "IsManaged": false,
    "MetadataId": "d00dc11e-ed91-478b-ac78-86b6784326ad",
    "HasChanged": null
}
},
"Description": {
    "LocalizedLabels": [],
    "UserLocalizedLabel": null
}
},
{
    "Value": 727000004,
    "Color": null,
    "IsManaged": false,
    "ExternalValue": "",
    "ParentValues": [],
    "MetadataId": null,
    "HasChanged": null,
    "Label": {
        "LocalizedLabels": [
            {
                "Label": "Foxtrot",
                "LanguageCode": 1033,
                "IsManaged": false,
                "MetadataId": "36a565b7-cd21-4505-812b-5567c28eec23",
                "HasChanged": null
            }
        ],
        "UserLocalizedLabel": {
            "Label": "Foxtrot",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "36a565b7-cd21-4505-812b-5567c28eec23",
            "HasChanged": null
        }
    },
    "Description": {
        "LocalizedLabels": [],
        "UserLocalizedLabel": null
    }
}
]
}
```

Salida de la consola:

```
Retrieved Choice column options:  
    Value:727000000 Label:Bravo  
    Value:727000001 Label:Delta  
    Value:727000002 Label:Alpha  
    Value:727000003 Label:Charlie  
    Value:727000004 Label:Foxtrot
```

Agregar una opción al conjunto de opciones local

1. Agregue una opción a la columna de elección usando [InsertOptionValue Action](#).

⚠️ Nota

`InsertOptionValue` y las siguientes acciones para trabajar con opciones tiene un parámetro `SolutionUniqueName` para que establezca el nombre único de la solución en lugar de usar el encabezado de solicitud

`MSCRM.SolutionUniqueName`.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/InsertOptionValue HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json  
  
{  
    "AttributeLogicalName": "sample_choice",  
    "EntityLogicalName": "sample_bankaccount",  
    "Value": 727000005,  
    "Label": {  
        "@odata.type": "Microsoft.Dynamics.CRM.Label",  
        "LocalizedLabels": [  
            {  
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",  
                "Label": "Echo",  
                "LanguageCode": 1033,  
                "IsManaged": false  
            }  
        ],  
        "UserLocalizedLabel": {
```

```
    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
    "Label": "Echo",
    "LanguageCode": 1033,
    "IsManaged": false
  },
},
"SolutionUniqueName": "examplesolution"
}
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.InsertOptionValueRe
sponse",
  "NewOptionValue": 727000005
}
```

Salida de la consola:

```
Added new option with label 'Echo'
```

2. Recupere las opciones de la columna de elección nuevamente usando la misma consulta que antes:

Salida de la consola:

```
The option values for the picklist:
  Value: 727000000, Label:Bravo
  Value: 727000001, Label:Delta
  Value: 727000002, Label:Alpha
  Value: 727000003, Label:Charlie
  Value: 727000004, Label:Foxtrot
  Value: 727000005, Label:Echo
```

Opciones para reordenar la columna de opciones

1. Reordene las opciones de la columna de opciones mediante [OrderOption Action](#).

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/OrderOption HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
  "EntityLogicalName": "sample_bankaccount",
  "AttributeLogicalName": "sample_choice",
  "Values": [
    727000002,
    727000000,
    727000003,
    727000001,
    727000005,
    727000004
  ],
  "SolutionUniqueName": "examplesolution"
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
```

Salida de la consola:

```
Options re-ordered.
```

2. Recupere las opciones de la columna Elección nuevamente usando la misma consulta que antes para ver las opciones en el nuevo orden.

Salida de la consola:

```
The option values for the picklist in the new order:
Value: 727000002, Label:Alpha
Value: 727000000, Label:Bravo
Value: 727000003, Label:Charlie
Value: 727000001, Label:Delta
```

```
Value: 727000005, Label:Echo  
Value: 727000004, Label:Foxtrot
```

Eliminar valor de opción local

1. Elimine una opción mediante [DeleteOptionValue Action](#).

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/DeleteOptionValue HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json  
  
{  
    "AttributeLogicalName": "sample_choice",  
    "EntityLogicalName": "sample_bankaccount",  
    "Value": 727000004  
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent  
OData-Version: 4.0
```

Salida de la consola:

```
Deleting a local option value...  
Local OptionSet option value deleted.
```

Columna de lista de selección múltiple

1. Cree una columna de opciones de selección múltiple usando [MultiSelectPicklistAttributeMetadata EntityType](#).

Solicitud:

HTTP

```
POST [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
  "@odata.type": "Microsoft.Dynamics.CRM.MultiSelectPicklistAttributeMetadata",
  "AttributeType": "Virtual",
  "AttributeName": {
    "Value": "MultiSelectPicklistType"
  },
  "SourceTypeMask": 0,
  "OptionSet": {
    "@odata.type": "Microsoft.Dynamics.CRM.OptionSetMetadata",
    "Options": [
      {
        "Value": 727000000,
        "Label": {
          "@odata.type": "Microsoft.Dynamics.CRM.Label",
          "LocalizedLabels": [
            {
              "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
              "Label": "Appetizer",
              "LanguageCode": 1033,
              "IsManaged": false
            }
          ],
          "UserLocalizedLabel": {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Appetizer",
            "LanguageCode": 1033,
            "IsManaged": false
          }
        }
      },
      {
        "Value": 727000001,
        "Label": {
          "@odata.type": "Microsoft.Dynamics.CRM.Label",
          "LocalizedLabels": [
            {
              "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
              "Label": "Entree",
              "LanguageCode": 1033,
              "IsManaged": false
            }
          ],
          "UserLocalizedLabel": {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
```

```
        "Label": "Entree",
        "LanguageCode": 1033,
        "IsManaged": false
    }
}
},
{
"Value": 727000002,
"Label": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Dessert",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Dessert",
        "LanguageCode": 1033,
        "IsManaged": false
    }
}
}
],
"IsGlobal": false,
"OptionsetType": "Picklist"
},
"Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "MultiSelect Choice Attribute",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "MultiSelect Choice Attribute",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Sample MultiSelect Choice",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ]
}
```

```
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Sample MultiSelect Choice",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"RequiredLevel": {
    "Value": "None",
    "CanBeChanged": false,
    "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
},
"SchemaName": "sample_MultiSelectChoice"
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(2c1c3050-112a-ed11-9db1-00224804f8e2)
```

Salida de la consola:

```
Creating a MultiSelect Choice column...
Created MultiSelect Choice column with id:2c1c3050-112a-ed11-9db1-
00224804f8e2
```

2. Recupere las opciones de la columna de selección múltiple usando `GET`

```
EntityDefinitions(LogicalName='sample_bankaccount')/Attributes(LogicalName='sa
mple_multiselectchoice')/Microsoft.Dynamics.CRM.MultiSelectPicklistAttributeMe
tadata?$select=SchemaName&$expand=OptionSet
```

Salida de la consola:

```
The option values for the multi-select choice column:
Value: 727000000, Label:Appetizer
```

```
Value: 727000001, Label:Entree  
Value: 727000002, Label:Dessert
```

Insertar valor de estado

Use [InsertStatusValue Action](#) para agregar una nueva opción a una columna `statuscode`. Debe especificar para qué `StateCode` es válido.

ⓘ Nota

Observe que el valor devuelto se aplica al valor del editor `customizationoptionvalueprefix` (72700) automáticamente.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/InsertStatusValue HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
    "AttributeLogicalName": "statuscode",
    "EntityLogicalName": "sample_bankaccount",
    "Label": {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [
            {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Frozen",
                "LanguageCode": 1033,
                "IsManaged": false
            }
        ],
        "UserLocalizedLabel": {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Frozen",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    },
    "StateCode": 1,
    "SolutionUniqueName": "examplesolution"
}
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.InsertStatusValueRespons
e",
    "NewOptionValue": 727000000
}
```

Salida de la consola:

```
Created new Status value:727000000
```

Sección 3: Crear y usar OptionSet global

1. Cree una opción global (conjunto de opciones). Este conjunto de opciones se llama `sample_colors` y contiene opciones para rojo, amarillo y verde.

Nota

La solicitud especifica los valores de opción utilizando el editor `customizationoptionvalueprefix`.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/GlobalOptionSetDefinitions
HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
    "@odata.type": "Microsoft.Dynamics.CRM.OptionSetMetadata",
    "Options": [
        {
            "Name": "Red"
        },
        {
            "Name": "Yellow"
        },
        {
            "Name": "Green"
        }
    ]
}
```

```
"Value": 727000000,
"Label": {
"@odata.type": "Microsoft.Dynamics.CRM.Label",
"LocalizedLabels": [
{
"@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
"Label": "Red",
"LanguageCode": 1033,
"IsManaged": false
}
],
"UserLocalizedLabel": {
"@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
"Label": "Red",
"LanguageCode": 1033,
"IsManaged": false
}
}
},
{
"Value": 727000001,
"Label": {
"@odata.type": "Microsoft.Dynamics.CRM.Label",
"LocalizedLabels": [
{
"@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
"Label": "Yellow",
"LanguageCode": 1033,
"IsManaged": false
}
],
"UserLocalizedLabel": {
"@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
"Label": "Yellow",
"LanguageCode": 1033,
"IsManaged": false
}
}
},
{
"Value": 727000002,
"Label": {
"@odata.type": "Microsoft.Dynamics.CRM.Label",
"LocalizedLabels": [
{
"@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
"Label": "Green",
"LanguageCode": 1033,
"IsManaged": false
}
],
"UserLocalizedLabel": {
"@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
"Label": "Green",
"LanguageCode": 1033,
```

```

        "IsManaged": false
    }
}
]
,
"Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Color Choice",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Color Choice",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Colors",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Colors",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"Name": "sample_colors",
"OptionsetType": "Picklist"
}

```

Respuesta:

HTTP

```

HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/GlobalOptionSetDefinitions(7cf8c56-112a-ed11-9db1-
00224804f8e2)

```

Salida de la consola:

```
Created a new global option set with id:7cf8c56-112a-ed11-9db1-00224804f8e2
```

2. Recupere las opciones de elección global.

Solicitud:

HTTP

```
GET [Organization  
Uri]/api/data/v9.2/GlobalOptionSetDefinitions(7cf8c56-112a-ed11-9db1-00224804f8e2)/Microsoft.Dynamics.CRM.OptionSetMetadata HTTP/1.1  
Consistency: Strong  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK  
OData-Version: 4.0  
  
{  
    "@odata.context": "[Organization  
Uri]/api/data/v9.2/$metadata#GlobalOptionSetDefinitions/Microsoft.Dynamics.CRM.OptionSetMetadata/$entity",  
    "ParentOptionSetName": null,  
    "IsCustomOptionSet": true,  
    "IsGlobal": true,  
    "IsManaged": false,  
    "Name": "sample_colors",  
    "ExternalTypeName": null,  
    "OptionSetType": "Picklist",  
    "IntroducedVersion": "1.0.0.0",  
    "MetadataId": "7cf8c56-112a-ed11-9db1-00224804f8e2",  
    "HasChanged": null,  
    "Options": [  
        {  
            "Value": 727000000,  
            "Color": null,  
            "IsManaged": false,  
            "ExternalValue": "",  
            "ParentValues": [],  
            "MetadataId": null,
```

```
"HasChanged": null,
"Label": {
"LocalizedLabels": [
{
"Label": "Red",
"LanguageCode": 1033,
"IsManaged": false,
"MetadataId": "2c1fa94f-3714-4615-995b-690158d0d989",
"HasChanged": null
}
],
"UserLocalizedLabel": {
"Label": "Red",
"LanguageCode": 1033,
"IsManaged": false,
"MetadataId": "2c1fa94f-3714-4615-995b-690158d0d989",
"HasChanged": null
}
},
"Description": {
"LocalizedLabels": [],
"UserLocalizedLabel": null
}
},
{
"Value": 727000001,
"Color": null,
"IsManaged": false,
"ExternalValue": "",
"ParentValues": [],
"MetadataId": null,
"HasChanged": null,
"Label": {
"LocalizedLabels": [
{
"Label": "Yellow",
"LanguageCode": 1033,
"IsManaged": false,
"MetadataId": "a499c2fe-c13a-4c1e-b190-db8ae74396f5",
"HasChanged": null
}
],
"UserLocalizedLabel": {
"Label": "Yellow",
"LanguageCode": 1033,
"IsManaged": false,
"MetadataId": "a499c2fe-c13a-4c1e-b190-db8ae74396f5",
"HasChanged": null
}
},
"Description": {
"LocalizedLabels": [],
"UserLocalizedLabel": null
}
},
```

```
{
    "Value": 727000002,
    "Color": null,
    "IsManaged": false,
    "ExternalValue": "",
    "ParentValues": [],
    "MetadataId": null,
    "HasChanged": null,
    "Label": {
        "LocalizedLabels": [
            {
                "Label": "Green",
                "LanguageCode": 1033,
                "IsManaged": false,
                "MetadataId": "8378af2c-4b68-4ea4-ad37-e676f696e1ba",
                "HasChanged": null
            }
        ],
        "UserLocalizedLabel": {
            "Label": "Green",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "8378af2c-4b68-4ea4-ad37-e676f696e1ba",
            "HasChanged": null
        }
    },
    "Description": {
        "LocalizedLabels": [],
        "UserLocalizedLabel": null
    }
},
],
"Description": {
    "LocalizedLabels": [
        {
            "Label": "Color Choice",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "7efd8c56-112a-ed11-9db1-00224804f8e2",
            "HasChanged": null
        }
    ],
    "UserLocalizedLabel": {
        "Label": "Color Choice",
        "LanguageCode": 1033,
        "IsManaged": false,
        "MetadataId": "7efd8c56-112a-ed11-9db1-00224804f8e2",
        "HasChanged": null
    }
},
},
"DisplayName": {
    "LocalizedLabels": [
        {
            "Label": "Colors",
            "LanguageCode": 1033,
```

```

        "IsManaged": false,
        "MetadataId": "7dfd8c56-112a-ed11-9db1-00224804f8e2",
        "HasChanged": null
    }
],
"UserLocalizedLabel": {
    "Label": "Colors",
    "LanguageCode": 1033,
    "IsManaged": false,
    "MetadataId": "7dfd8c56-112a-ed11-9db1-00224804f8e2",
    "HasChanged": null
}
},
"IsCustomizable": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "iscustomizable"
}
}

```

Salida de la consola:

```

List the retrieved options for the colors global option set:
Value: 727000000 Label:Red
Value: 727000001 Label:Yellow
Value: 727000002 Label:Green

```

- Cree una columna de opciones que el conjunto de opciones global. Asocie la definición de columna al conjunto de opciones global usando:

```
"GlobalOptionSet@odata.bind": "/GlobalOptionSetDefinitions(7cf8c56-112a-ed11-9db1-00224804f8e2)".
```

Solicitud:

HTTP

```

POST [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
"@odata.type": "Microsoft.Dynamics.CRM.PicklistAttributeMetadata",
"AttributeType": "Picklist",

```

```

"AttributeType": {
    "Value": "PicklistType"
},
"SourceTypeMask": 0,
"GlobalOptionSet@odata.bind": "/GlobalOptionSetDefinitions(7cf8c56-112a-ed11-9db1-00224804f8e2)",
"Description": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Colors Global Picklist Attribute",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Colors Global Picklist Attribute",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"DisplayName": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Sample Colors",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Sample Colors",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"RequiredLevel": {
    "Value": "None",
    "CanBeChanged": false,
    "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
},
"SchemaName": "sample_Colors"
}

```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(81fd8c56-112a-ed11-9db1-00224804f8e2)
```

Salida de la consola:

```
Created Choice column with id:81fd8c56-112a-ed11-9db1-00224804f8e2
using colors global optionset.
```

Sección 4: Crear relación con el cliente

1. Use [CreateCustomerRelationships Action](#) para crear una relación con el cliente.

Esta acción agrega una columna de búsqueda para la tabla `sample_BankAccount` que permite establecer un registro `account` o `contact`.

`CreateCustomerRelationships` tiene un parámetro `Lookup` [ComplexLookupAttributeMetadata ComplexType](#) y un parámetro `OneToManyRelationships` que contienen un par de relaciones definidas mediante [ComplexOneToManyRelationshipMetadata ComplexType](#).

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/CreateCustomerRelationships
HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
  "Lookup": {
    "@odata.type":
      "Microsoft.Dynamics.CRM.ComplexLookupAttributeMetadata",
    "AttributeType": "Lookup",
    "AttributeName": {
      "Value": "LookupType"
    },
    "Format": "None",
    "Targets": [
      "account",
      "contact"
    ]
  }
}
```

```
        "contact"
    ],
    "ColumnNumber": 0,
    "Description": {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [
            {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "The owner of the bank account",
                "LanguageCode": 1033,
                "IsManaged": false
            }
        ],
        "UserLocalizedLabel": {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "The owner of the bank account",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    },
    "DisplayName": {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [
            {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Bank Account owner",
                "LanguageCode": 1033,
                "IsManaged": false
            }
        ],
        "UserLocalizedLabel": {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Bank Account owner",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    },
    "RequiredLevel": {
        "Value": "ApplicationRequired",
        "CanBeChanged": false,
        "ManagedPropertyLogicalName": "canmodifyrequirementlevelsettings"
    },
    "SchemaName": "sample_CustomerId",
    "SourceType": 0
},
"OneToManyRelationships": [
    {
        "@odata.type": "Microsoft.Dynamics.CRM.ComplexOneToManyRelationshipMetadata",
        "ReferencedEntity": "account",
        "ReferencingEntity": "sample_bankaccount",
        "RelationshipBehavior": 0,
        "RelationshipType": "OneToManyRelationship",
        "SchemaName": "sample_BankAccount_Customer_Account",
        "SecurityTypes": "None"
```

```

},
{
  "@odata.type": "Microsoft.Dynamics.CRM.ComplexOneToManyRelationshipMetadata",
  "ReferencedEntity": "contact",
  "ReferencingEntity": "sample_bankaccount",
  "RelationshipBehavior": 0,
  "RelationshipType": "OneToManyRelationship",
  "SchemaName": "sample_BankAccount_Customer_Contact",
  "SecurityTypes": "None"
}
],
"SolutionUniqueName": "examplesolution"
}

```

Respuesta:

HTTP

```

HTTP/1.1 200 OK
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.CreateCustomerRelat
ionshipsResponse",
  "RelationshipIds": [
    "84fd8c56-112a-ed11-9db1-00224804f8e2",
    "8dfd8c56-112a-ed11-9db1-00224804f8e2"
  ],
  "AttributeId": "59478264-16af-4bcc-8baa-b154df0d6767"
}

```

2. Use el `CreateCustomerRelationshipsResponse.AttributeId` Valor para recuperar la propiedad `Targets` de la columna de búsqueda para la relación con el cliente.

Solicitud:

HTTP

```

GET [Organization
Uri]/api/data/v9.2/EntityDefinitions(LogicalName='sample_bankaccount')/
Attributes(LogicalName='sample_customerid')/Microsoft.Dynamics.CRM.Look
upAttributeMetadata?$select=Targets HTTP/1.1
Consistency: Strong
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#EntityDefinitions('sample_bankaccount')/At
tributes/Microsoft.Dynamics.CRM.LookupAttributeMetadata(Targets)/$entit
y",
  "Targets": [
    "account",
    "contact"
  ],
  "MetadataId": "59478264-16af-4bcc-8baa-b154df0d6767"
}
```

Salida de la consola:

```
The Target values of the Lookup column created:
  account
  contact
```

3. Use el `CreateCustomerRelationshipsResponse.RelationshipIds` valores para recuperar las relaciones para la columna del cliente.

Primero para la relación con `account`:

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/RelationshipDefinitions(84fd8c56-
112a-ed11-9db1-
00224804f8e2)/Microsoft.Dynamics.CRM.OneToManyRelationshipMetadata
HTTP/1.1
Consistency: Strong
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#RelationshipDefinitions/Microsoft.Dynamics
.CRM.OneToManyRelationshipMetadata/$entity",
    "ReferencedAttribute": "accountid",
    "ReferencedEntity": "account",
    "ReferencingAttribute": "sample_customerid",
    "ReferencingEntity": "sample_bankaccount",
    "IsHierarchical": false,
    "EntityKey": null,
    "IsRelationshipAttributeDenormalized": false,
    "ReferencedEntityNavigationPropertyName":
        "sample_BankAccount_Customer_Account",
    "ReferencingEntityNavigationPropertyName": "sample_CustomerId_account",
    "RelationshipBehavior": 1,
    "IsDenormalizedLookup": null,
    "DenormalizedAttributeName": null,
    "IsCustomRelationship": true,
    "IsValidForAdvancedFind": true,
    "SchemaName": "sample_BankAccount_Customer_Account",
    "SecurityTypes": "Append",
    "IsManaged": false,
    "RelationshipType": "OneToManyRelationship",
    "IntroducedVersion": "1.0.0.0",
    "MetadataId": "84fd8c56-112a-ed11-9db1-00224804f8e2",
    "HasChanged": null,
    "AssociatedMenuConfiguration": {
        "Behavior": "UseCollectionName",
        "Group": "Details",
        "Order": 10000,
        "IsCustomizable": true,
        "Icon": null,
        "ViewId": "00000000-0000-0000-0000-000000000000",
        "AvailableOffline": true,
        "MenuItem": null,
        "QueryApi": null,
        "Label": {
            "LocalizedLabels": [],
            "UserLocalizedLabel": null
        }
    },
    "CascadeConfiguration": {
        "Assign": "NoCascade",
        "Delete": "RemoveLink",
        "Archive": "RemoveLink",
        "Merge": "Cascade",
        "Reparent": "NoCascade",
        "Share": "NoCascade",
        "Unshare": "NoCascade"
    }
}
```

```
        "RollupView": "NoCascade"
    },
    "RelationshipAttributes": [],
    "IsCustomizable": {
        "Value": true,
        "CanBeChanged": true,
        "ManagedPropertyLogicalName": "iscustomizable"
    }
}
```

Luego para la relación con `contact`:

Solicitud:

```
HTTP

GET [Organization Uri]/api/data/v9.2/RelationshipDefinitions(8dfd8c56-112a-ed11-9db1-00224804f8e2)/Microsoft.Dynamics.CRM.OneToManyRelationshipMetadata
HTTP/1.1
Consistency: Strong
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

```
HTTP

HTTP/1.1 200 OK
OData-Version: 4.0

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#RelationshipDefinitions/Microsoft.Dynamics
.CRM.OneToManyRelationshipMetadata/$entity",
    "ReferencedAttribute": "contactid",
    "ReferencedEntity": "contact",
    "ReferencingAttribute": "sample_customerid",
    "ReferencingEntity": "sample_bankaccount",
    "IsHierarchical": false,
    "EntityKey": null,
    "IsRelationshipAttributeDenormalized": false,
    "ReferencedEntityNavigationPropertyName":
        "sample_BankAccount_Customer_Contact",
    "ReferencingEntityNavigationPropertyName": "sample_CustomerId_contact",
    "RelationshipBehavior": 1,
    "IsDenormalizedLookup": null,
    "DenormalizedAttributeName": null,
    "IsCustomRelationship": true,
```

```

    "IsValidForAdvancedFind": true,
    "SchemaName": "sample_BankAccount_Customer_Contact",
    "SecurityTypes": "Append",
    "IsManaged": false,
    "RelationshipType": "OneToManyRelationship",
    "IntroducedVersion": "1.0.0.0",
    "MetadataId": "8dfd8c56-112a-ed11-9db1-00224804f8e2",
    "HasChanged": null,
    "AssociatedMenuConfiguration": {
        "Behavior": "UseCollectionName",
        "Group": "Details",
        "Order": 10000,
        "IsCustomizable": true,
        "Icon": null,
        "ViewId": "00000000-0000-0000-0000-000000000000",
        "AvailableOffline": true,
        "MenuItem": null,
        "QueryApi": null,
        "Label": {
            "LocalizedLabels": [],
            "UserLocalizedLabel": null
        }
    },
    "CascadeConfiguration": {
        "Assign": "NoCascade",
        "Delete": "RemoveLink",
        "Archive": "RemoveLink",
        "Merge": "Cascade",
        "Reparent": "NoCascade",
        "Share": "NoCascade",
        "Unshare": "NoCascade",
        "RollupView": "NoCascade"
    },
    "RelationshipAttributes": [],
    "IsCustomizable": {
        "Value": true,
        "CanBeChanged": true,
        "ManagedPropertyLogicalName": "iscustomizable"
    }
}

```

Salida de la consola:

```

The Schema Names of the relationships created:
sample_BankAccount_Customer_Account
sample_BankAccount_Customer_Contact

```

Sección 5: Crear y recuperar una relación de uno a muchos

Antes de crear una relación mediante código, debe confirmar que la relación es válida. Los diseñadores de [Power Apps](#) usan funciones especiales para mostrarle qué combinaciones son válidas. Puede usar las mismas funciones en su código para detectar si una relación en particular se puede crear o no.

Validar la idoneidad de la relación de 1:N

1. [CanBeReferenced Function](#) le indica si una tabla puede ser la tabla principal (uno) en una relación de uno a varios.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/CanBeReferenced HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
  "EntityName": "sample_bankaccount"
}
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.CanBeReferencedRespon
se",
  "CanBeReferenced": true
}
```

Salida de la consola:

The sample_BankAccount table is eligible to be a primary table in a one-to-many relationship.

2. [CanBeReferencing Function](#) indica si la tabla especificada puede ser la tabla de referencia en una relación de uno a varios. La tabla de referencia es la tabla a la que se le ha agregado una columna de búsqueda para que sea "varios" en la relación de uno a varios.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/CanBeReferencing HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
  "EntityName": "contact"
}
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.CanBeReferencingRes
ponse",
  "CanBeReferencing": true
}
```

Salida de la consola:

The contact table is eligible to be a related table in a one-to-many relationship.

Identificar entidades de referencia potenciales

En el contexto de una tabla específica que puede ser la tabla parmary en una relación de uno a muchos, use [GetValidReferencingEntities Function](#) para identificar qué otras tablas pueden ser las relacionadas con él.

Solicitud:

```
HTTP

GET [Organization
Uri]/api/data/v9.2/GetValidReferencingEntities(ReferencedEntityName='sample_
bankaccount') HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

```
HTTP

HTTP/1.1 200 OK
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.GetValidReferencingEntit
iesResponse",
  "EntityNames": [
    "msdyn_slakpi",
    "workflowbinary",
    "apisettings",
    "flowsession",
    "knowledgearticle",
    "socialprofile",
    "goal",
    "metric",
    "goalrollupquery",
    "mailbox",
    "position",
    "channelaccessprofile",
    "externalparty",
    "channelaccessprofilerule",
    "channelaccessprofileruleitem",
    "sample_bankaccount",
    "privilegesremovalsetting",
    "knowledgebaserecord",
    "msdyn_insightsstorevirtualentity",
    "aaduser",
    "sharepointdocument",
    "msfp_unsubscribedrecipient",
    "msdyn_dataflow",
    "flowmachineimage",
```

"queueitem",
"appointment",
"msdyn_federatedarticleincident",
"msfp_surveyresponse",
"msdyn_dataflowrefreshhistory",
"mailmergetemplate",
"contact",
"organizationdatasyncstate",
"bot",
"knowledgearticleviews",
"slaitem",
"msfp_question",
"category",
"connection",
"newprocess",
"msfp_survey",
"emailserverprofile",
"appnotification",
"feedback",
"activityfileattachment",
"organizationdatasyncsubscriptionentity",
"msdyn_nonrelationalds",
"expiredprocess",
"msfp_surveyinvite",
"msfp_alert",
"businessunit",
"msfp_alertrule",
"slakpiinstance",
"email",
"datasyncstate",
"msdyn_entityrefreshhistory",
"msdyn_componentlayerdatasource",
"account",
"kbarticle",
"systemuser",
"task",
"letter",
"reportcategory",
"phonecall",
"actioncard",
"msdyn_kmfederatedsearchconfig",
"featurecontrolsetting",
"translationprocess",
"recurringappointmentmaster",
"externalpartyitem",
"msdyn_aibdatasetfile",
"socialactivity",
"flowmachineimageversion",
"fax",
"msdyn_kbattachment",
"serviceplanmapping",
"msdyn_knowledgearticletemplate",
"msfp_emailtemplate",
"conversationtranscript",
"sharepointsite",

```

    "processstage",
    "msfp_localizedemailtemplate",
    "queue",
    "msdyn_richtextfile",
    "msdyn_serviceconfiguration",
    "team",
    "sharedlinksetting",
    "territory",
    "msdyn_federatedarticle",
    "msdyn_knowledgepersonalfilter",
    "sharepointdocumentlocation",
    "chat",
    "msfp_fileresponse",
    "msfp_satisfactionmetric",
    "msdyn_aibfeedbackloop",
    "msdyn_customcontrolextendedsettings",
    "msfp_surveyreminder",
    "msfp_questionresponse",
    "msfp_project"
]
}

```

Salida de la consola:

```
The contact table is in the list of potential referencing entities for sample_BankAccount.
```

Crear una relación de 1:N

La siguiente solicitud crea una relación de uno a varios entre `sample_BankAccount` y las tablas de contactos con una columna de búsqueda agregada a la tabla `contact`.

Solicitud:

HTTP

```

POST [Organization Uri]/api/data/v9.2/RelationshipDefinitions HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
    "@odata.type": "Microsoft.Dynamics.CRM.OneToManyRelationshipMetadata",
    "AssociatedMenuConfiguration": {
        "Behavior": "UseLabel",
        "Group": "Details",

```

```
"Label": {
    "@odata.type": "Microsoft.Dynamics.CRM.Label",
    "LocalizedLabels": [
        {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Cardholders",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    ],
    "UserLocalizedLabel": {
        "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
        "Label": "Cardholders",
        "LanguageCode": 1033,
        "IsManaged": false
    }
},
"Order": 10000,
"ViewId": "00000000-0000-0000-0000-000000000000"
},
"CascadeConfiguration": {
    "Assign": "NoCascade",
    "Delete": "RemoveLink",
    "Merge": "Cascade",
    "Reparent": "NoCascade",
    "Share": "NoCascade",
    "Unshare": "NoCascade",
    "RollupView": "NoCascade"
},
"IsHierarchical": false,
"ReferencedAttribute": "sample_bankaccountid",
"ReferencedEntity": "sample_bankaccount",
"ReferencingEntity": "contact",
"Lookup": {
    "@odata.type": "Microsoft.Dynamics.CRM.LookupAttributeMetadata",
    "AttributeType": "Lookup",
    "AttributeTypeName": {
        "Value": "LookupType"
    },
    "Format": "None",
    "Description": {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [
            {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "The bank account this contact has access to.",
                "LanguageCode": 1033,
                "IsManaged": false
            }
        ],
        "UserLocalizedLabel": {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "The bank account this contact has access to.",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    }
}
```

```

        },
        "DisplayName": {
            "@odata.type": "Microsoft.Dynamics.CRM.Label",
            "LocalizedLabels": [
                {
                    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                    "Label": "Bank Account",
                    "LanguageCode": 1033,
                    "IsManaged": false
                }
            ],
            "UserLocalizedLabel": {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Bank Account",
                "LanguageCode": 1033,
                "IsManaged": false
            }
        },
        "SchemaName": "sample_BankAccountId"
    },
    "IsCustomRelationship": false,
    "IsManaged": false,
    "IsValidForAdvancedFind": false,
    "RelationshipType": "OneToManyRelationship",
    "SchemaName": "sample_BankAccount_Contacts",
    "SecurityTypes": "None"
}

```

Respuesta:

HTTP

```

HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/RelationshipDefinitions(991efd5f-112a-ed11-9db1-
00224804f8e2)

```

Salida de la consola:

```

Creating a one-to-many relationship...
Created one-to-many relationship: RelationshipDefinitions(991efd5f-112a-
ed11-9db1-00224804f8e2)

```

Recuperar una relación de 1:N

La siguiente solicitud recupera la relación creada por la solicitud anterior.

ⓘ Nota

Porque `RelationshipDefinitions` contiene definiciones de relación de uno a muchos y de muchos a muchos, debe incluir lo siguiente en la URL para convertir al tipo que desea recuperar: `/Microsoft.Dynamics.CRM.OneToManyRelationshipMetadata`. De lo contrario, el valor devuelto será `RelationshipMetadataBase EntityType` y no incluirá las propiedades específicas del `OneToManyRelationshipMetadata EntityType`.

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/RelationshipDefinitions(991efd5f-112a-ed11-9db1-00224804f8e2)/Microsoft.Dynamics.CRM.OneToManyRelationshipMetadata
HTTP/1.1
Consistency: Strong
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#RelationshipDefinitions/Microsoft.Dynamics.CRM.
OneToManyRelationshipMetadata/$entity",
    "ReferencedAttribute": "sample_bankaccountid",
    "ReferencedEntity": "sample_bankaccount",
    "ReferencingAttribute": "sample_bankaccountid",
    "ReferencingEntity": "contact",
    "IsHierarchical": false,
    "EntityKey": null,
    "IsRelationshipAttributeDenormalized": false,
    "ReferencedEntityNavigationPropertyName": "sample_BankAccount_Contacts",
    "ReferencingEntityNavigationPropertyName": "sample_BankAccountId",
    "RelationshipBehavior": 1,
    "IsDenormalizedLookup": null,
    "DenormalizedAttributeName": null,
    "IsCustomRelationship": true,
```

```
"IsValidForAdvancedFind": false,
"SchemaName": "sample_BankAccount_Contacts",
"SecurityTypes": "Append",
"IsManaged": false,
"RelationshipType": "OneToManyRelationship",
"IntroducedVersion": "1.0.0.0",
"MetadataId": "991efd5f-112a-ed11-9db1-00224804f8e2",
"HasChanged": null,
"AssociatedMenuConfiguration": {
    "Behavior": "UseLabel",
    "Group": "Details",
    "Order": 10000,
    "IsCustomizable": true,
    "Icon": null,
    "ViewId": "00000000-0000-0000-0000-000000000000",
    "AvailableOffline": true,
    "MenuItem": null,
    "QueryApi": null,
    "Label": {
        "LocalizedLabels": [
            {
                "Label": "Cardholders",
                "LanguageCode": 1033,
                "IsManaged": false,
                "MetadataId": "a99a420f-6778-4f2f-b42b-64bc84b2c2d2",
                "HasChanged": null
            }
        ],
        "UserLocalizedLabel": {
            "Label": "Cardholders",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "a99a420f-6778-4f2f-b42b-64bc84b2c2d2",
            "HasChanged": null
        }
    }
},
"CascadeConfiguration": {
    "Assign": "NoCascade",
    "Delete": "RemoveLink",
    "Archive": "RemoveLink",
    "Merge": "NoCascade",
    "Reparent": "NoCascade",
    "Share": "NoCascade",
    "Unshare": "NoCascade",
    "RollupView": "NoCascade"
},
"RelationshipAttributes": [],
"IsCustomizable": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "iscustomizable"
}
}
```

Salida de la consola:

```
Retrieved relationship: sample_BankAccount_Contacts
```

Sección 6: Crear y recuperar una relación de muchos a uno

Una relación de varios a uno es simplemente una relación uno a varios vista desde la otra dirección. Los siguientes ejemplos crean una columna de búsqueda denominada `sample_relatedaccountid` en la tabla `sample_BankAccount` que hace referencia a una fila en la tabla `account`.

Crear una relación de N:1

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/RelationshipDefinitions HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
    "@odata.type": "Microsoft.Dynamics.CRM.OneToManyRelationshipMetadata",
    "AssociatedMenuConfiguration": {
        "Behavior": "UseLabel",
        "Group": "Details",
        "Label": {
            "@odata.type": "Microsoft.Dynamics.CRM.Label",
            "LocalizedLabels": [
                {
                    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                    "Label": "Related Bank Accounts",
                    "LanguageCode": 1033,
                    "IsManaged": false
                }
            ],
            "UserLocalizedLabel": {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Related Bank Accounts",
                "LanguageCode": 1033,
                "IsManaged": false
            }
        }
    }
}
```

```
        },
    },
    "Order": 10000,
    "ViewId": "00000000-0000-0000-0000-000000000000"
},
"CascadeConfiguration": {
    "Assign": "NoCascade",
    "Delete": "RemoveLink",
    "Merge": "Cascade",
    "Reparent": "NoCascade",
    "Share": "NoCascade",
    "Unshare": "NoCascade",
    "RollupView": "NoCascade"
},
"IsHierarchical": false,
"ReferencedAttribute": "accountid",
"ReferencedEntity": "account",
"ReferencingEntity": "sample_bankaccount",
"Lookup": {
    "@odata.type": "Microsoft.Dynamics.CRM.LookupAttributeMetadata",
    "AttributeType": "Lookup",
    "AttributeName": {
        "Value": "LookupType"
    },
    "Format": "None",
    "Description": {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [
            {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "An Account related to the bank account.",
                "LanguageCode": 1033,
                "IsManaged": false
            }
        ],
        "UserLocalizedLabel": {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "An Account related to the bank account.",
            "LanguageCode": 1033,
            "IsManaged": false
        }
    },
    "DisplayName": {
        "@odata.type": "Microsoft.Dynamics.CRM.Label",
        "LocalizedLabels": [
            {
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
                "Label": "Related Account",
                "LanguageCode": 1033,
                "IsManaged": false
            }
        ],
        "UserLocalizedLabel": {
            "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",
            "Label": "Related Account",
            "LanguageCode": 1033
        }
    }
}
```

```
        "LanguageCode": 1033,  
        "IsManaged": false  
    }  
,  
    "SchemaName": "sample_RelatedAccountId"  
},  
"IsCustomRelationship": false,  
"IsManaged": false,  
"IsValidForAdvancedFind": false,  
"RelationshipType": "OneToManyRelationship",  
"SchemaName": "sample_Account_BankAccounts",  
"SecurityTypes": "None"  
}  
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent  
OData-Version: 4.0  
OData-EntityId: [Organization  
Uri]/api/data/v9.2/RelationshipDefinitions(0901c466-112a-ed11-9db1-  
00224804f8e2)
```

Salida de la consola:

```
Created many-to-one relationship: RelationshipDefinitions(0901c466-112a-  
ed11-9db1-00224804f8e2)
```

Recuperar una relación de N:1

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/RelationshipDefinitions(0901c466-112a-  
ed11-9db1-00224804f8e2)/Microsoft.Dynamics.CRM.OneToManyRelationshipMetadata  
HTTP/1.1  
Consistency: Strong  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#RelationshipDefinitions/Microsoft.Dynamics.CRM.
OneToManyRelationshipMetadata/$entity",
    "ReferencedAttribute": "accountid",
    "ReferencedEntity": "account",
    "ReferencingAttribute": "sample_relatedaccountid",
    "ReferencingEntity": "sample_bankaccount",
    "IsHierarchical": false,
    "EntityKey": null,
    "IsRelationshipAttributeDenormalized": false,
    "ReferencedEntityNavigationPropertyName": "sample_Account_BankAccounts",
    "ReferencingEntityNavigationPropertyName": "sample_RelatedAccountId",
    "RelationshipBehavior": 1,
    "IsDenormalizedLookup": null,
    "DenormalizedAttributeName": null,
    "IsCustomRelationship": true,
    "IsValidForAdvancedFind": false,
    "SchemaName": "sample_Account_BankAccounts",
    "SecurityTypes": "Append",
    "IsManaged": false,
    "RelationshipType": "OneToManyRelationship",
    "IntroducedVersion": "1.0.0.0",
    "MetadataId": "0901c466-112a-ed11-9db1-00224804f8e2",
    "HasChanged": null,
    "AssociatedMenuConfiguration": {
        "Behavior": "UseLabel",
        "Group": "Details",
        "Order": 10000,
        "IsCustomizable": true,
        "Icon": null,
        "ViewId": "00000000-0000-0000-0000-000000000000",
        "AvailableOffline": true,
        "MenuItem": null,
        "QueryApi": null,
        "Label": {
            "LocalizedLabels": [
                {
                    "Label": "Related Bank Accounts",
                    "LanguageCode": 1033,
                    "IsManaged": false,
                    "MetadataId": "b2ad2cce-5c7c-46aa-8bbf-7903d33ef019",
                    "HasChanged": null
                }
            ],
            "UserLocalizedLabel": {
                "Label": "Related Bank Accounts",
                "LanguageCode": 1033,
                "IsManaged": false,
            }
        }
    }
}
```

```
        "MetadataId": "b2ad2cce-5c7c-46aa-8bbf-7903d33ef019",
        "HasChanged": null
    }
},
{
"CascadeConfiguration": {
    "Assign": "NoCascade",
    "Delete": "RemoveLink",
    "Archive": "RemoveLink",
    "Merge": "Cascade",
    "Reparent": "NoCascade",
    "Share": "NoCascade",
    "Unshare": "NoCascade",
    "RollupView": "NoCascade"
},
"RelationshipAttributes": [],
"IsCustomizable": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "iscustomizable"
}
}
```

Salida de la consola:

```
Retrieved relationship: sample_Account_BankAccounts
```

Sección 7: Crear y recuperar una relación de muchos a muchos

Al igual que uno a muchos Relaciones, hay funciones especiales utilizadas por los diseñadores en [Power Apps](#) para evitar combinaciones no válidas al crear muchos a muchos Relaciones.

Validar la idoneidad de la relación de N:N

1. [CanManyToMany Function](#) indica si una tabla puede participar en una relación de varios a varios. Entonces, esta solicitud prueba la tabla `contact`.

Solicitud:

```
HTTP
```

```
POST [Organization Uri]/api/data/v9.2/CanManyToMany HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{{
"EntityName": "contact"
}}
```

Respuesta:

```
HTTP

HTTP/1.1 200 OK
OData-Version: 4.0

{{
"@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.CanManyToManyRespon
se",
"CanManyToMany": true
}}
```

Salida de la consola:

```
The contact table can participate in many-to-many relationships.
```

2. Esta solicitud realiza la misma prueba en la tabla `sample_bankaccount`.

Solicitud:

```
HTTP

POST [Organization Uri]/api/data/v9.2/CanManyToMany HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{{
"EntityName": "sample_bankaccount"
}}
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.CanManyToManyRespon
se",
  "CanManyToMany": true
}
```

Salida de la consola:

```
The sample_bankaccount table can participate in many-to-many
relationships.
```

Identificar entidades potenciales para relaciones de N:N

Use [GetValidManyToMany Function](#) para obtener una lista de tablas que pueden participar en relaciones de varios a varios.

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/GetValidManyToMany HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.GetValidManyToManyRespon
se",
  "EntityNames": [
    "msdyn_slakpi",
    "workflowbinary",
```

```
"apisettings",
"flowsession",
"theme",
"knowledgearticle",
"socialprofile",
"goal",
"position",
"externalparty",
"channelaccessprofileruleitem",
"routingruleitem",
"sample_bankaccount",
"synapsealinkexternaltablestate",
"synapsesdatabase",
"msdyn_aimodel",
"aaduser",
"applicationuser",
"msfp_unsubscribedrecipient",
"msdyn_aiconfiguration",
"msdyn_dataflow",
"flowmachineimage",
"queueitem",
"synapselinkschedule",
"msdyn_federatedarticleincident",
"flowmachine",
"synapsealinkprofile",
"msdyn_dataflowrefreshhistory",
"solutioncomponentrelationshipconfiguration",
"contact",
"organizationdatasyncstate",
"botcomponent",
"bot",
"msdyn_componentlayer",
"msdyn_odatav4ds",
"msfp_question",
"msdyn_aibfile",
"msdyn_solutionhistorydatasource",
"msdyn_solutionhealthruleset",
"newprocess",
"connectionreference",
"msdyn_knowledgemanagementsetting",
"msdyn_pmrecording",
"msfp_survey",
"msdyn_aibdatasetscontainer",
"package",
"msdyn_solutioncomponentsummary",
"msdyn_helppage",
"appnotification",
"organizationdatasyncsubscriptionentity",
"msdyn_aidotrainingboundingbox",
"msdyn_nonrelationalds",
"expiredprocess",
"msdyn_analysisresultdetail",
"msfp_alertrule",
"msdyn_solutioncomponentcountsummary",
"msdyn_kalanguagesetting",
```

"transactioncurrency",
"exportsolutionupload",
"msdyn_pmprocessusersettings",
"datasyncstate",
"msdyn_entityrefreshhistory",
"msdyn_analysisresult",
"msdyn_componentlayerdatasource",
"account",
"kbarticle",
"systemuser",
"task",
"msdyn_analysisjob",
"solutioncomponentconfiguration",
"msdyn_knowledgesearchfilter",
"stagesolutionupload",
"msdyn_pmttemplate",
"phonecall",
"msdyn_solutioncomponentdatasource",
"environmentvariablevalue",
"msdyn_aitemplate",
"userrating",
"synapsealinkprofileentity",
"featurecontrolsetting",
"translationprocess",
"msdyn_pminferredtask",
"customapirequestparameter",
"externalpartyitem",
"msdyn_aibdatasetfile",
"flowmachineingroup",
"flowmachineimageversion",
"msdyn_aibdatasetrecord",
"msdyn_kbattachment",
"msdyn_aifptrainingdocument",
"customapiresponseproperty",
"msdyn_knowledgearticletemplate",
"msdyn_aiodimage",
"msdyn_knowledgesearchinsight",
"msfp_emailtemplate",
"catalog",
"msdyn_knowledgeinteractioninsight",
"conversationtranscript",
"msdyn_pmanalysishistory",
"msdyn_datalakeds",
"canvasappextendedmetadata",
"msfp_localizedemailtemplate",
"msdynce_botcontent",
"queue",
"msdyn_solutionhealthruleargument",
"msdyn_aibfileattacheddata",
"msdyn_richtextfile",
"msdyn_kmpersonalizationsetting",
"msdyn_aiodtrainingimage",
"msdyn_serviceconfiguration",
"msdyn_knowledgearticleimage",
"team",

```

    "territory",
    "msdyn_solutioncomponentcountdatasource",
    "catalogassignment",
    "msdyn_federatedarticle",
    "msdyn_solutionhealthrule",
    "msdyn_solutionhistory",
    "msdyn_knowledgepersonalfilter",
    "organizationdatasyncsubscription",
    "solutioncomponentbatchconfiguration",
    "connector",
    "solutioncomponentattributeconfiguration",
    "synapsealinkprofileentitystate",
    "msdyn_aiodlabel",
    "customapi",
    "msdyn_aibdataset",
    "msfp_fileresponse",
    "environmentvariabledefinition",
    "msdyn_analysiscomponent",
    "msfp_satisfactionmetric",
    "msdyn_tour",
    "msdyn_customcontrolextendedsettings",
    "msfp_surveyreminder",
    "virtualentitymetadata",
    "msfp_questionresponse",
    "msfp_project"
]
}

```

Salida de la consola:

```

Contact is in the list of potential tables for N:N.
sample_BankAccount is in the list of potential tables for N:N.

```

Crear una relación de N:N

Esta solicitud crea una relación de varios a varios entre las tablas `sample_BankAccount` y `Contact`.

Solicitud:

HTTP

```

POST [Organization Uri]/api/data/v9.2/RelationshipDefinitions HTTP/1.1
MSCRM.SolutionUniqueName: examplesolution
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

```

```
{  
    "@odata.type": "Microsoft.Dynamics.CRM.ManyToManyRelationshipMetadata",  
    "Entity1AssociatedMenuConfiguration": {  
        "Behavior": "UseLabel",  
        "Group": "Details",  
        "Label": {  
            "@odata.type": "Microsoft.Dynamics.CRM.Label",  
            "LocalizedLabels": [  
                {  
                    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",  
                    "Label": "Bank Accounts",  
                    "LanguageCode": 1033,  
                    "IsManaged": false  
                }  
            ],  
            "UserLocalizedLabel": {  
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",  
                "Label": "Bank Accounts",  
                "LanguageCode": 1033,  
                "IsManaged": false  
            }  
        },  
        "Order": 10000,  
        "ViewId": "00000000-0000-0000-0000-000000000000"  
    },  
    "Entity1LogicalName": "sample_bankaccount",  
    "Entity2AssociatedMenuConfiguration": {  
        "Behavior": "UseLabel",  
        "Group": "Details",  
        "Label": {  
            "@odata.type": "Microsoft.Dynamics.CRM.Label",  
            "LocalizedLabels": [  
                {  
                    "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",  
                    "Label": "Contacts",  
                    "LanguageCode": 1033,  
                    "IsManaged": false  
                }  
            ],  
            "UserLocalizedLabel": {  
                "@odata.type": "Microsoft.Dynamics.CRM.LocalizedLabel",  
                "Label": "Contacts",  
                "LanguageCode": 1033,  
                "IsManaged": false  
            }  
        },  
        "Order": 10000,  
        "ViewId": "00000000-0000-0000-0000-000000000000"  
    },  
    "Entity2LogicalName": "contact",  
    "IntersectEntityName": "sample_sample_BankAccounts_Contacts",  
    "IsCustomRelationship": false,  
    "IsManaged": false,  
    "IsValidForAdvancedFind": false,  
}
```

```
"RelationshipType": "OneToManyRelationship",
"SchemaName": "sample_sample_BankAccounts_Contacts",
"SecurityTypes": "None"
}
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent
OData-Version: 4.0
OData-EntityId: [Organization
Uri]/api/data/v9.2/RelationshipDefinitions(55c9f86c-112a-ed11-9db1-
00224804f8e2)
```

Salida de la consola:

```
Created Many-to-Many relationship at: RelationshipDefinitions(55c9f86c-112a-
ed11-9db1-00224804f8e2)
```

Recuperar una relación N:N

Esta solicitud recupera la relación varios a varios creada por la solicitud anterior.

ⓘ Nota

Como se mencionaba antes, como `RelationshipDefinitions` contiene definiciones de relación de uno a muchos y de muchos a muchos, debe incluir lo siguiente en la URL para convertir al tipo que desea recuperar:

`/Microsoft.Dynamics.CRM.ManyToManyRelationshipMetadata`. De lo contrario, el valor devuelto será `RelationshipMetadataBase EntityType` y no incluirá las propiedades específicas del `ManyToManyRelationshipMetadata EntityType`.

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/RelationshipDefinitions(55c9f86c-112a-
ed11-9db1-
00224804f8e2)/Microsoft.Dynamics.CRM.ManyToManyRelationshipMetadata HTTP/1.1
Consistency: Strong
OData-MaxVersion: 4.0
```

```
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

HTTP/1.1 200 OK

OData-Version: 4.0

```
{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#RelationshipDefinitions/Microsoft.Dynamics.CRM.
ManyToManyRelationshipMetadata/$entity",
  "Entity1LogicalName": "sample_bankaccount",
  "Entity2LogicalName": "contact",
  "IntersectEntityName": "sample_sample_bankaccounts_contacts",
  "Entity1IntersectAttribute": "sample_bankaccountid",
  "Entity2IntersectAttribute": "contactid",
  "Entity1NavigationPropertyName": "sample_sample_BankAccounts_Contacts",
  "Entity2NavigationPropertyName": "sample_sample_BankAccounts_Contacts",
  "IsCustomRelationship": true,
  "IsValidForAdvancedFind": false,
  "SchemaName": "sample_sample_BankAccounts_Contacts",
  "SecurityTypes": "None",
  "IsManaged": false,
  "RelationshipType": "ManyToManyRelationship",
  "IntroducedVersion": "1.0.0.0",
  "MetadataId": "55c9f86c-112a-ed11-9db1-00224804f8e2",
  "HasChanged": null,
  "Entity1AssociatedMenuConfiguration": {
    "Behavior": "UseLabel",
    "Group": "Details",
    "Order": 10000,
    "IsCustomizable": true,
    "Icon": null,
    "ViewId": "00000000-0000-0000-0000-000000000000",
    "AvailableOffline": true,
    "MenuItemId": null,
    "QueryApi": null,
    "Label": {
      "LocalizedLabels": [
        {
          "Label": "Bank Accounts",
          "LanguageCode": 1033,
          "IsManaged": false,
          "MetadataId": "271999d1-6fd9-4413-b027-f2a1b231e0a4",
          "HasChanged": null
        }
      ],
      "UserLocalizedLabel": {
        "Label": "Bank Accounts",
        "LanguageCode": 1033
      }
    }
  }
}
```

```

        "LanguageCode": 1033,
        "IsManaged": false,
        "MetadataId": "271999d1-6fd9-4413-b027-f2a1b231e0a4",
        "HasChanged": null
    }
}
},
"Entity2AssociatedMenuConfiguration": {
    "Behavior": "UseLabel",
    "Group": "Details",
    "Order": 10000,
    "IsCustomizable": true,
    "Icon": null,
    "ViewId": "00000000-0000-0000-0000-000000000000",
    "AvailableOffline": true,
    "MenuItem": null,
    "QueryApi": null,
    "Label": {
        "LocalizedLabels": [
            {
                "Label": "Contacts",
                "LanguageCode": 1033,
                "IsManaged": false,
                "MetadataId": "1fcff441-9a41-42b1-a0d9-e92daa47230f",
                "HasChanged": null
            }
        ],
        "UserLocalizedLabel": {
            "Label": "Contacts",
            "LanguageCode": 1033,
            "IsManaged": false,
            "MetadataId": "1fcff441-9a41-42b1-a0d9-e92daa47230f",
            "HasChanged": null
        }
    }
},
"IsCustomizable": {
    "Value": true,
    "CanBeChanged": true,
    "ManagedPropertyLogicalName": "iscustomizable"
}
}

```

Salida de la consola:

```
Retrieved Many-to-Many relationship:sample_sample_BankAccounts_Contacts
```

Sección 8: Exportar una solución administrada

Use [ExportSolution Action](#) para exportar la solución como administrada. Esta acción incluye muchos modificadores que puede usar para incluir información adicional como parte de la solución, pero en este caso, todas esas opciones están desactivadas. Más información: [Trabajar con soluciones](#)

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/ExportSolution HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{
    "SolutionName": "examplesolution",
    "Managed": true,
    "ExportAutoNumberingSettings": false,
    "ExportCalendarSettings": false,
    "ExportCustomizationSettings": false,
    "ExportEmailTrackingSettings": false,
    "ExportGeneralSettings": false,
    "ExportMarketingSettings": false,
    "ExportOutlookSynchronizationSettings": false,
    "ExportRelationshipRoles": false,
    "ExportIsvConfig": false,
    "ExportSales": false,
    "ExportExternalApplications": false
}
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
    "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#Microsoft.Dynamics.CRM.ExportSolutionResponse",
    "ExportSolutionFile": "[ Binary content removed for brevity]"
}
```

Salida de la consola:

```
Solution Exported to E:\GitHub\PowerApps-Samples\dataverse\webapi\C#-
```

Sección 9: Eliminar registros de muestra

Las referencias a todos los registros creados en esta muestra se han agregado a una lista. En esta sección, todos los registros creados se eliminan mediante una operación `$batch`.

Solicitud:

HTTP

```
POST [Organization Uri]/api/data/v9.2/$batch HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

--batch_d6cb9873-6b53-490d-942d-0bea79b8ba9a
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-Length: 136

DELETE /api/data/v9.2/RelationshipDefinitions(991efd5f-112a-ed11-9db1-
00224804f8e2) HTTP/1.1

--batch_d6cb9873-6b53-490d-942d-0bea79b8ba9a
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-Length: 130

DELETE /api/data/v9.2/EntityDefinitions(5872b902-112a-ed11-9db1-
00224804f8e2) HTTP/1.1

--batch_d6cb9873-6b53-490d-942d-0bea79b8ba9a
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-Length: 122

DELETE /api/data/v9.2/solutions(5472b902-112a-ed11-9db1-00224804f8e2)
HTTP/1.1

--batch_d6cb9873-6b53-490d-942d-0bea79b8ba9a
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-Length: 123

DELETE /api/data/v9.2/publishers(a78ab7fc-102a-ed11-9db1-00224804f8e2)
```

HTTP/1.1

```
--batch_d6cb9873-6b53-490d-942d-0bea79b8ba9a
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-Length: 139

DELETE /api/data/v9.2/GlobalOptionSetDefinitions(7cf8c56-112a-ed11-9db1-
00224804f8e2) HTTP/1.1

--batch_d6cb9873-6b53-490d-942d-0bea79b8ba9a--
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

--batchresponse_9fa66f62-38d1-4890-91ce-4185fd556745
Content-Type: application/http
Content-Transfer-Encoding: binary

HTTP/1.1 204 No Content
OData-Version: 4.0

--batchresponse_9fa66f62-38d1-4890-91ce-4185fd556745
Content-Type: application/http
Content-Transfer-Encoding: binary

HTTP/1.1 204 No Content
OData-Version: 4.0

--batchresponse_9fa66f62-38d1-4890-91ce-4185fd556745
Content-Type: application/http
Content-Transfer-Encoding: binary

HTTP/1.1 204 No Content
OData-Version: 4.0
```

```
--batchresponse_9fa66f62-38d1-4890-91ce-4185fd556745
Content-Type: application/http
Content-Transfer-Encoding: binary

HTTP/1.1 204 No Content
OData-Version: 4.0

--batchresponse_9fa66f62-38d1-4890-91ce-4185fd556745--
```

Salida de la consola:

```
Deleting created records...
```

Sección 10: Importar y eliminar una solución administrada

- Este paso importa solución administrada exportado en [Sección 8: Exportar solución administrada](#) utilizando el [ImportSolution Action](#).

Solicitud:

```
HTTP
POST [Organization Uri]/api/data/v9.2/ImportSolution HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json

{{
    "OverwriteUnmanagedCustomizations": false,
    "PublishWorkflows": false,
    "CustomizationFile": "[ Binary content removed for brevity]",
    "ImportJobId": "00000000-0000-0000-0000-000000000000"
}}
```

Respuesta:

```
HTTP
HTTP/1.1 204 NoContent
```

OData-Version: 4.0

Salida de la consola:

```
Sending request to import the examplesolution solution...
Solution imported as a managed solution.
```

2. Obtenga el `solutionid` de la solución administrada por `uniquename` para que pueda eliminarlo.

Solicitud:

HTTP

```
GET [Organization Uri]/api/data/v9.2/solutions?
$select=solutionid&$filter=uniquename%20eq%20'examplesolution' HTTP/1.1
OData-MaxVersion: 4.0
OData-Version: 4.0
If-None-Match: null
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 200 OK
OData-Version: 4.0

{
  "@odata.context": "[Organization
Uri]/api/data/v9.2/$metadata#solutions(solutionid)",
  "value": [
    {
      "@odata.etag": "W/\\"13291034\\\"",
      "solutionid": "07439497-6992-4e30-81e0-628a91984af5"
    }
  ]
}
```

Eliminar una solución administrada

Este paso final elimina el solución administrada importado para devolver el sistema al estado original.

Solicitud:

HTTP

```
DELETE [Organization Uri]/api/data/v9.2/solutions(07439497-6992-4e30-81e0-  
628a91984af5) HTTP/1.1  
OData-MaxVersion: 4.0  
OData-Version: 4.0  
If-None-Match: null  
Accept: application/json
```

Respuesta:

HTTP

```
HTTP/1.1 204 NoContent  
OData-Version: 4.0
```

Salida de la consola:

```
Sending request to delete the examplesolution solution...  
Managed solution deleted.  
--Metadata Operations sample completed--
```

Consulte también

[Usar la API web de Dataverse](#)

[Utilizar la API web con definiciones de tabla](#)

[Consultar definiciones de tabla con la API web](#)

[Crear y actualizar definiciones de tabla mediante la API web](#)

[Crear y actualizar relaciones de tabla mediante la API web](#)

[Crear y actualizar selecciones \(conjuntos de opciones\) mediante la API web](#)

[Ejemplo de operaciones de metadatos de API web \(C#\)](#)

ⓘ Nota

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta. [↗](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad ↗](#)).

Ejemplos de operaciones de datos de API web (C#)

Artículo • 25/04/2023

En este tema se proporciona información sobre los ejemplos de la API web implementada con C# mediante .NET 6.0. Si bien cada ejemplo se centra en un aspecto distinto de la API web de Microsoft Dataverse, comparten características y estructura similares.

Requisitos previos

Los siguientes elementos son necesarios para crear y ejecutar los ejemplos en C# de la API web de Dataverse:

- Una versión más reciente de Microsoft Visual Studio 2022 o posterior. Una versión gratuita, [Visual Studio Community](#), está disponible para descarga [aquí](#).
- El acceso a Dataverse con una cuenta que tenga el rol de seguridad Administrador del sistema.

Biblioteca de clases WebAPIService

Estos ejemplos utilizan la [Biblioteca de clases WebAPIService \(C#\)](#) para proporcionar un código común reutilizable que maneje los errores de límites de protección del servicio que se puede volver a probar. Más información: [Límites de API de protección de servicios](#)

Lista de ejemplos de la API web (C#)

En la siguiente tabla se muestran los ejemplos implementados en C#. Cada ejemplo se describe en forma más general en un tema de grupo de ejemplo correspondiente que se centra en la solicitud HTTP y en mensajes de respuesta, como se detalla en el tema [Web API Samples](#).

Muestra	Grupo de ejemplo	Descripción

Muestra	Grupo de ejemplo	Descripción
Ejemplo de operaciones básicas de la API web (C#)	Ejemplo de operaciones básicas de la API web	Demuestra cómo crear, recuperar, actualizar, eliminar, asociar y anular la asociación de registros de entidad de Dataverse.
Ejemplo de datos de consulta de API web (C#)	Ejemplo de datos de consulta de la API web	Demuestra cómo usar sintaxis y funciones de consulta de OData v4 así como funciones de consulta de Dataverse. Incluye ejemplos de trabajo con consultas predefinidas y uso de FetchXML para realizar consultas.
Ejemplo de operaciones condicionales de la API web (C#)	Ejemplo de operaciones condicionales de la API web	Demuestra cómo realizar operaciones condicionales que usted especifica con criterios de ETag.
Ejemplo de funciones y acciones de la API web (C#)	Ejemplo de funciones y acciones de la API web	Demuestra cómo usar funciones y acciones enlazadas y sin enlazar, incluidas acciones personalizadas.
Ejemplo de operaciones de metadatos de API web (C#)	Ejemplo de operaciones de metadatos de API web	Demuestra cómo realizar operaciones seleccionadas que modifican el esquema o los metadatos de Dataverse.

Los siguientes ejemplos muestran métodos para agregar paralelismo y simultaneidad a las aplicaciones. Estas capacidades son una parte importante para maximizar el rendimiento al realizar operaciones que agregarán o actualizarán datos dentro de Dataverse.

- [Ejemplo de operaciones en paralelo de la API web Web ApiService \(C#\)](#)
- [Ejemplo de operaciones paralelas de API web con componentes de flujo de datos TPL \(C#\)](#)

Cómo descargar y ejecutar los ejemplos

El código fuente de cada muestra está disponible en GitHub en: [PowerApps-Samples/dataverse/webapi/C#-NETx/](#). Puede descargar el repositorio como un archivo zip que contiene los archivos de solución para los ejemplos. Para obtener más información, consulte la sección **Cómo ejecutar este ejemplo** en cada tema de ejemplo.

Consulte también

[Usar la API web de Dataverse](#)

[Ejemplos de la API web](#)

[Ejemplos de la API web \(JavaScript del lado del cliente\)](#)

[Biblioteca de clases WebAPIService \(C#\)](#)

 **Nota**

¿Puede indicarnos sus preferencias de idioma de documentación? [Realice una breve encuesta.](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad](#)).

Biblioteca de clases WebAPIService (C#)

Artículo • 23/02/2023

WebAPIService es un proyecto de biblioteca de clases de .NET 6.0 de muestra que demuestra varias capacidades importantes que debe incluir cuando usa la API web de Dataverse.

Esta biblioteca demuestra:

- Administrar los [límites de protección del servicio](#) de Dataverse con la resiliencia de .NET y la biblioteca de manejo de fallas transitorias [Polly](#).
- Administrar un [HttpClient](#) en .NET usando [IHttpClientFactory](#).
- Usar datos de configuración para gestionar el comportamiento del cliente.
- Gestionar los errores devueltos por la API web de Dataverse.
- Un patrón de reutilización de código mediante:
 - Creando clases que heredan de [HttpRequestMessage](#) y [HttpResponseMessage](#).
 - Métodos que usan esas clases.
 - Un patrón modular para agregar nuevas capacidades según sea necesario.

ⓘ Nota

Esta biblioteca de muestras es una ayuda que utilizan todos los ejemplos de API web de C# de Dataverse, pero no es un SDK. Se prueba solo para confirmar que las muestras que lo usan se ejecutan correctamente. Este código de muestra se proporciona 'tal cual' sin garantía de reutilización.

Esta biblioteca no:

- **Administra la autenticación.** Depende de una función pasada desde una aplicación que proporcionará el token de acceso para usar. Todos los ejemplos de Web API dependen de una [clase de aplicación](#) que gestiona la autenticación mediante la [Biblioteca de autenticación de Microsoft \(MSAL\)](#). MSAL admite varios tipos diferentes de flujos de autenticación. Estas muestras utilizan el flujo de [Nombre de usuario/contraseña \(ROPC\)](#) por simplicidad, pero no se recomienda. Para sus aplicaciones, debe usar uno de los otros flujos. Más información: [Compatibilidad con el flujo de autenticación en la biblioteca de autenticación de Microsoft](#).
- **Proporcionar cualquier capacidad de generación de código.** Todas las clases utilizadas en las muestras están escritas a mano. Todos los datos de entidades

comerciales utilizan la conocida [clase Json.NET JObject](#) en lugar de una clase que representa el tipo de entidad.

- **Proporcionar un modelo de objetos para componer consultas de OData.** Todas las consultas muestran la sintaxis de consulta de OData como parámetros de consulta.

Código

Puede encontrar el código fuente de la biblioteca de clases WebAPIService y la solución Visual Studio en [PowerApps-Samples/dataverse/webapi/C#-NETx/WebAPIService](#)

Lista de clases

Las siguientes son clases que se incluyen en WebAPIService.

Clase Service

La clase `Service` proporciona métodos para enviar solicitudes a Dataverse a través de un [HttpClient](#) administrado usando [IHttpClientFactory](#).

`Service` es el componente principal de todos los ejemplos y se puede usar solo para completar cualquier operación demostrada con el código de ejemplo. Todo lo demás que se incluye en el WebAPIService o cualquiera de los ejemplos que lo utilizan proporciona la reutilización del código y permite que las capacidades de la API web de Dataverse se prueben a un nivel superior.

El constructor `Service` acepta una instancia de clase `Config` que contiene dos propiedades requeridas: `GetAccessToken` y `Url`. Todas las demás propiedades representan opciones que tienen valores predeterminados.

El constructor utiliza la inyección de dependencia para crear un `IHttpClientFactory` que puede devolver un `HttpClient` con nombre con las propiedades especificadas en la función `ConfigureHttpClient`. Si este cliente utilizará o no cookies depende de si se establece el parámetro `Config.DisableCookies`. En el constructor la política definida por el método estático `GetRetryPolicy` que controla cómo se gestionarán los errores transitorios y los límites de protección del servicio de Dataverse.

Métodos de servicio

La clase `Service` tiene los métodos siguientes:

Método SendAsync

Este es el único método responsable en última instancia de todas las operaciones.

Este método:

- Tiene un parámetro [HttpRequestMessage](#).
- Devuelve [Task<HttpResponseMessage>](#)
- Expone la misma firma que el [método HttpClient.SendAsync\(HttpRequestMessage\)](#) y se puede usar de la misma manera.
- Llama a la función configurada en el método [Config.GetAccessToken](#) para establecer el valor de encabezado [Authorization](#) para la solicitud.
- Utiliza el [método IHttpClientFactory.CreateClient](#) para obtener el [HttpClient](#) con nombre para enviar la solicitud.
- Lanzará una [ServiceException](#) si la propiedad [HttpResponseMessage.IsSuccessStatusCode](#) es false, por lo que no necesita verificar esto cuando use este método.

Método SendAsync<T>

Este método facilita la devolución de una clase que incluye propiedades que se encuentran en los ComplexTypes devueltos por OData Actions and Functions en la API web de Dataverse.

- Tiene un parámetro [HttpRequestMessage](#). Cuando se usa este método se espera, pero no se requiere, que el parámetro de solicitud sea una de las [*clases de respuesta](#) que se derivan de [HttpResponseMessage](#).
- Devuelve [Task<T>](#) donde [T](#) es una clase derivada de [HttpResponseMessage](#). Para obtener más información, consulte [*Clases de respuesta](#).
- Llama al [método SendAsync](#).
- Utiliza el método de extensión [HttpResponseMessage As<T>](#) para devolver el tipo solicitado.

El siguiente ejemplo muestra el uso con [WhoAmI function](#):

C#

```
static async Task WhoAmI(Service service)
{
    var response = await service.SendAsync<WhoAmIResponse>(new
WhoAmIRequest());

    Console.WriteLine($"Your user ID is {response.UserId}");
}
```

Método ParseError

Este método analizará el contenido de un `HttpResponseMessage` para que un `HttpRequestMessage` con errores devuelva una `ServiceException`. Se utiliza dentro del método `SendAsync` cuando la propiedad `HttpResponseMessage.IsSuccessStatusCode` es `false`. También se puede usar para extraer información de error de las instancias devueltas de `HttpResponseMessage` por `BatchResponse.HttpResponseMessages` cuando la propiedad `BatchRequest.ContinueOnError` se establece en `true`. Más información: [Batch](#)

Propiedades del servicio

Service tiene una sola propiedad: `BaseAddress`.

Propiedad BaseAddress

Esta propiedad devuelve la URL base establecida en `Config.Url`. Esto es necesario cuando se instancia la clase `BatchRequest` para agregar a una URL relativa cada vez que se requiera una URL absoluta.

Clase Config

La clase `Config` contiene propiedades que controlan el comportamiento de la aplicación como se muestra en la siguiente tabla:

Property	Type	Description
<code>GetAccessToken</code>	<code>Func<Task<string>></code>	Una función proporcionada por la aplicación cliente para devolver un token de acceso.
<code>Url</code>	<code>string?</code>	URL base del entorno. ¿Qué es: https://org.api.crm.dynamics.com ?
<code>CallerObjectId</code>	<code>Guid</code>	Valor <code>SystemUser.ActiveDirectoryGuid</code> para aplicar para la suplantación de identidad. El valor predeterminado es <code>Guid.Empty</code> Más información: Suplantar a otro usuario utilizando la API web
<code>TimeoutInSeconds</code>	<code>ushort</code>	Cuánto tiempo esperar un tiempo de espera. El valor predeterminado es 120 segundos.
<code>MaxRetries</code>	<code>byte</code>	Número máximo de veces para volver a intentar cuando se llega a los límites de protección del servicio. El valor predeterminado es 3.

Property	Type	Description
Version	string	La versión del servicio que se va a utilizar. El valor predeterminado es v9.2
DisableCookies	bool	Si se deshabilitan las cookies para ganar rendimiento en escenarios de carga masiva de datos. Más información: Afinidad del servidor

Clase EntityReference

La clase EntityReference representa una referencia a un registro en una tabla de Dataverse. En OData, los recursos se identifican mediante una URL. EntityReference proporciona métodos para facilitar la creación y el acceso a las propiedades de las direcciones URL.

Constructores de EntityReference

Use los siguientes constructores para instanciar una EntityReference.

EntityReference(string entitySetName, Guid? ID)

Crea una referencia de entidad usando `EntitySetName` y `Guid`.

EntityReference(string uri)

Analiza una URL absoluta o relativa para crear una referencia de entidad, incluidas las URL que usan claves alternativas.

EntityReference(string setName, Dictionary<string, string>? keyAttributes)

Use este constructor para instanciar una referencia de entidad usando un clave alternativa.

ⓘ Nota

Los valores clave deben ser valores de cadena. Esto no convierte otros tipos en cadenas apropiadas.

Propiedades de EntityReference

EntityReference tiene las siguientes propiedades públicas:

Property	Type	Description
Id	Guid?	El valor de la clave principal del registro cuando no se usa un clave alternativa.
KeyAttributes	Dictionary<string, string>	Los valores de cadena que representan los valores clave alternativa utilizados en una URL.
SetName	string	EntitySetName del tipo de entidad.
Path	string	URL relativa del registro.

Métodos EntityReference

EntityReference tiene los siguientes métodos públicos: Ninguno de ellos requiere ningún parámetro.

Nombre del método	Tipo devuelto	Description
AsODataId	string	Devuelve una cadena formateada para usar como referencia de parámetro a un registro en la URL para una función OData.
As JObject	JObject	Devuelve un JObject que se puede usar como referencia de parámetro a un registro en una acción de OData.

Clases de errores

ODataError, Error y ODataException son clases utilizadas para deserializar errores devueltos por el servicio. No necesitarás trabajar con ellos directamente.

ServiceException

ServiceException es una [clase de excepción](#) que contiene propiedades del error devuelto por el servicio. Utilice el [método ParseError](#) para obtener una instancia de esta excepción.

Extensiones

WebAPIService tiene un método de extensión de un tipo .NET.

HttpResponseMessage As<T>

Esta extensión crea una instancia de `T` donde `T` se deriva de `HttpResponseMessage` y copia las propiedades de `HttpResponseMessage` en la clase derivada. Lo usa el método `Service SendAsync<T>`, pero también se puede usar por separado. Por ejemplo, al usar la clase `BatchRequest`, los elementos de `BatchResponse.HttpResponseMessages` serán tipos `HttpResponseMessage`. Puede usar esta extensión para convertirlos a la clase derivada apropiada para facilitar el acceso a cualquier propiedad.

Mensajes

La carpeta `Messages` incluye clases que heredan de `HttpRequestMessage` o `HttpResponseMessage`.

Estas clases proporcionan definiciones reutilizables de solicitudes y respuestas que corresponden a operaciones de OData que puede usar en cualquier entorno de Dataverse.

Estas clases también proporcionan ejemplos de operaciones específicas que se pueden aplicar usando `HttpRequestMessage` y `HttpResponseMessage` sin derivar de esos tipos.

Dentro de una aplicación, también puede crear mensajes personalizados, por ejemplo, representando una API personalizada en su entorno, utilizando el mismo patrón. Estas son clases modulares y no es necesario incluirlas en la carpeta `WebAPIService.Messages`.

Por ejemplo, el [ejemplo de acciones y funciones de API web \(C#\)](#) utiliza una API personalizada que no está incluida en Dataverse hasta que se instale una solución que contenga la API personalizada. La definición de las clases correspondientes para usar este mensaje se encuentra en la aplicación de muestra que lo usa:

- [FunctionsAndActions/Messages/IsSystemAdminRequest.cs ↗](#)
- [FunctionsAndActions/Messages/IsSystemAdminResponse.cs ↗](#)

Clases *Request

Estas clases generalmente tendrán un constructor con parámetros que instanciarán un `HttpRequestMessage` con los datos necesarios para realizar la operación. Pueden tener propiedades separadas según corresponda.

El ejemplo más simple de este patrón es la clase `WhoAmIRequest`.

C#

```
namespace PowerApps.Samples.Messages
{
    /// <summary>
    /// Contains the data to perform the WhoAmI function
    /// </summary>
    public sealed class WhoAmIRequest : HttpRequestMessage
    {
        /// <summary>
        /// Initializes the WhoAmIRequest
        /// </summary>
        public WhoAmIRequest()
        {
            Method = HttpMethod.Get;
            RequestUri = new Uri(
                uriString: "WhoAmI",
                uriKind: UriKind.Relative);
        }
    }
}
```

Los nombres de estas clases pueden alinearse con las clases del SDK de Dataverse [Microsoft.Xrm.Sdk.Messages Namespace](#) pero no se limitan a esas operaciones. La API web permite realizar algunas operaciones que no se pueden realizar con el SDK, por ejemplo `CreateRetrieveRequest` es un mensaje que creará un registro y lo recuperará. El SDK de Dataverse no proporciona esta capacidad en una sola solicitud.

Clases *Response

Cuando las clases *Request devuelvan un valor, habrá una clase *Response correspondiente para acceder a las propiedades devueltas. Si *Request devuelve `204 No Content`, la operación devolverá [HttpResponseMessage](#) pero no habrá clase derivada. Utilice el [método SendAsync](#) para enviar estas solicitudes.

Las clases *Response proporcionan propiedades con tipo que acceden a las propiedades `HttpResponseMessage Headers` o `Content` y las analiza para proporcionar acceso al tipo complejo devuelto por la operación.

Un ejemplo de esto es la clase WhoAmIResponse. Dentro de esta clase puede encontrar todo el código necesario para extraer las propiedades de [WhoAmIResponse ComplexType](#).

C#

```

using Newtonsoft.Json.Linq;

namespace PowerApps.Samples.Messages
{
    // This class must be instantiated by either:
    // - The Service.SendAsync<T> method
    // - The HttpResponseMessage.As<T> extension in Extensions.cs

    /// <summary>
    /// Contains the response from the WhoAmIRequest
    /// </summary>
    public sealed class WhoAmIResponse : HttpResponseMessage
    {

        // Cache the async content
        private string? _content;

        //Provides JObject for property getters
        private JObject _jObject
        {
            get
            {
                _content ??=
Content.ReadAsStringAsync().GetAwaiter().GetResult();

                return JObject.Parse(_content);
            }
        }

        /// <summary>
        /// Gets the ID of the business to which the logged on user belongs.
        /// </summary>
        public Guid BusinessUnitId =>
(Guid)_jObject.GetValue(nameof(BusinessUnitId));

        /// <summary>
        /// Gets ID of the user who is logged on.
        /// </summary>
        public Guid UserId => (Guid)_jObject.GetValue(nameof(UserId));

        /// <summary>
        /// Gets ID of the organization that the user belongs to.
        /// </summary>
        public Guid OrganizationId =>
(Guid)_jObject.GetValue(nameof(OrganizationId));
    }
}

```

Estas clases solo se pueden instanciar correctamente cuando las devuelve el [método SendAsync<T>](#) o usando la extensión [HttpResponseMessage As<T>](#) en un [HttpResponseMessage](#) devuelto por una propiedad [BatchResponse.HttpResponseMessages](#).

Lote

La carpeta `Batch` contiene tres clases para gestionar el envío de solicitudes OData `$batch`. Más información:[Ejecute las operaciones por lotes mediante API web](#)

BatchRequest

El constructor `BatchRequest` inicializa un `HttpRequestMessage` que se puede usar con el método `SendAsync<T>` para enviar solicitudes en lotes. El constructor requiere que se pase el valor `Service.BaseAddress` como parámetro.

`BatchRequest` tiene las siguientes propiedades.

Property	Type	Description
<code>ContinueOnError</code>	<code>Bool</code>	Controla si la operación por lotes debe continuar cuando se produce un error.
<code>ChangeSets</code>	<code>List<ChangeSet></code>	Uno o más conjuntos de cambios que se incluirán en el lote.
<code>Requests</code>	<code>List<HttpRequestMessage></code>	Uno o mas <code>HttpMessageRequest</code> para enviar fuera de cualquier <code>ChangeSet</code> .

Cuando `ChangeSets` o `Requests` están configurados, están encapsulados en `HttpMessageContent` y se añaden al `Content` de la solicitud. El método `ToMessageContent` privado aplica los cambios requeridos a los encabezados y devuelve el `HttpMessageContent` para las propiedades `ChangeSets` y `Requests`.

ChangeSet

Un conjunto de cambios representa un grupo de solicitudes que deben completarse dentro de una transacción.

Contiene una sola propiedad:

Property	Type	Description
<code>Requests</code>	<code>List<HttpRequestMessage></code>	Uno o mas <code>HttpMessageRequest</code> para realizar dentro de la transacción.

BatchResponse

`BatchResponse` tiene una sola propiedad:

Property	Type	Description
<code>HttpResponseMessages</code>	<code>List<HttpResponseMessage></code>	Las respuestas de la operación <code>\$batch</code> .

`BatchResponse` tiene un método privado `ParseMultipartContent` utilizado por el captador de la propiedad `HttpResponseMessages` para analizar el `MultipartContent` devuelto al `HttpResponseMessage` individual.

Para acceder a las propiedades de tipo de las instancias de `HttpResponseMessage` devueltas, puede usar el método de extensión [HttpResponseMessage As<T>](#).

Métodos

Para las operaciones que se realizan con frecuencia, la carpeta `Methods` contiene extensiones de la clase `Service`. Estos permiten utilizar las clases `*Request` correspondientes en una sola línea.

Se incluyen los siguientes métodos:

Method	Tipo devuelto	Description
<code>Create</code>	<code>Task<EntityReference></code>	Crea un nuevo registro.
<code>CreateRetrieve</code>	<code>Task<JObject></code>	Crea un nuevo registro y lo recupera.
<code>Delete</code>	<code>Task</code>	Elimina un registro.
<code>FetchXml</code>	<code>Task<FetchXmlResponse></code>	Recupera los resultados de una consulta FetchXml. La solicitud se envía con <code>POST</code> usando <code>\$batch</code> para mitigar los problemas en los que las URL largas se envían con <code>GET</code> puede exceder los límites.
<code>GetColumnValue<T></code>	<code>Task<T></code>	Recupera un único valor de columna de una fila de tabla.
<code>Retrieve</code>	<code>Task<JObject></code>	Recupera un registro.
<code>RetrieveMultiple</code>	<code>Task<RetrieveMultipleResponse></code>	Recupera múltiples registros.
<code>SetValue<T></code>	<code>Task</code>	Establece el valor de una columna para una fila de tabla.
<code>Update</code>	<code>Task</code>	Actualiza un registro.

Method	Tipo devuelto	Description
Upsert	Task<UpsertResponse>	Realiza un Upsert en un registro.

Dentro de una aplicación de muestra que usa WebAPIService cuando la operación no representa una API que se encuentra en Dataverse de forma predeterminada, el método se definirá en la aplicación en lugar de en WebAPIService.

Por ejemplo, el [ejemplo de acciones y funciones de API web \(C#\)](#) utiliza una API personalizada que no está incluida en Dataverse hasta que se instale una solución que contenga la API personalizada. La definición de este método se encuentra en la aplicación de muestra que lo utiliza: [FunctionsAndActions/Methods/IsSystemAdmin.cs](#)



Tipos

La carpeta `Types` contiene clases o enumeraciones que corresponden a [ComplexTypes](#) o [EnumTypes](#) necesarios como parámetros o propiedades de respuesta para los mensajes.

Metadata

La carpeta `Metadata` contiene `Messages` y `Types` específicos para operaciones que trabajan con definiciones de esquemas de Dataverse. Estas son frecuentemente clases con muchas propiedades que devuelven tipos complejos. Estos tipos se utilizan en el [ejemplo de operaciones de metadatos de API web \(C#\)](#).

Consulte también

[Ejemplo de operaciones básicas de la API web \(C#\)](#)

[Ejemplo de datos de consulta de API web \(C#\)](#)

[Ejemplo de operaciones condicionales de la API web \(C#\)](#)

[Ejemplo de funciones y acciones de la API web \(C#\)](#)

[Ejemplo de operaciones de metadatos de API web \(C#\)](#)

[Ejemplo de operaciones en paralelo de la API web WebAPIService \(C#\)](#)

[Ejemplo de operaciones paralelas de API web con componentes de flujo de datos TPL \(C#\)](#)

Nota

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta. [↗](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales (**declaración de privacidad** [↗](#)).

Ejemplo de operaciones básicas de la API web (C#)

Artículo • 23/02/2023

Este ejemplo .NET 6.0 demuestra cómo realizar operaciones de datos comunes utilizando la API web de Dataverse.

Esta muestra utiliza el código auxiliar común de la [biblioteca de clases WebAPIService \(C#\)](#).

ⓘ Nota

Este ejemplo implementa las operaciones de Dataverse y la salida de la consola detalladas en el [Ejemplo de operaciones básicas de API web](#) y utiliza las construcciones comunes de C# que se describen en [Ejemplos de API web \(C#\)](#).

Requisitos previos

Los siguientes elementos son necesarios para compilar y ejecutar este ejemplo:

- Microsoft Visual Studio 2022.
- Acceso a Dataverse con privilegios para realizar operaciones de datos.

Cómo ejecutar esta muestra

1. Clonar o descargar el repositorio de [ejemplos de PowerApps](#).
2. Localizar la carpeta `/dataverse/webapi/C#-NETx/BasicOperations/`.
3. Abrir el archivo `BasicOperations.sln` usando Visual Studio 2022
4. Editar el archivo `appsettings.json` para establecer los siguientes valores de propiedad:

Property	Instrucciones
<code>Url</code>	La URL para su entorno. Reemplazar el valor de marcador de posición <code>https://yourorg.api.crm.dynamics.com</code> con el valor para su entorno. Consulte Ver recursos para desarrolladores para encontrar esto.

Property	Instrucciones
UserPrincipalName	Reemplazar el valor de marcador de posición <code>you@yourorg.onmicrosoft.com</code> con el valor de UPN que usa para acceder al entorno.
Password	Reemplazar el valor de marcador de posición <code>yourPassword</code> con la contraseña que utiliza.

5. Guarde el archivo `appsettings.json`

6. Presione F5 para ejecutar el ejemplo.

Código

El código para este ejemplo está aquí: [PowerApps-Samples/dataverse/webapi/C#-NETx/BasicOperations/Program.cs](#)

Demostraciones

Esta muestra tiene 5 regiones:

Sección 1: Operaciones básicas para crear y actualizar

Operaciones:

- Crear un registro de contacto.
- Actualizar el registro de contacto.
- Recuperar el registro de contacto.
- Actualizar una sola propiedad del registro de contacto.
- Recuperar una sola propiedad del registro de contacto.

Sección 2: Crear registro asociado a otro

Operaciones: Asociar un nuevo registro a uno existente.

Sección 3: Crear entidades relacionadas

Operaciones: Crear las siguientes entradas en una operación: una cuenta, su contacto principal asociado y tareas abiertas para ese contacto. Estos tipos de entidad tienen el siguiente Relaciones:

```
Accounts
|---[Primary] Contact (N-to-1)
|---Tasks (1-to-N)
```

Sección 4: Asociar y desasociar entidades existentes

Operaciones:

- Agregar un contacto a la colección `contact_customer_accounts` de la cuenta.
- Quitar un contacto de la colección `contact_customer_accounts` de la cuenta.
- Asociar un rol de seguridad a un usuario usando la colección `systemuserroles_association`.
- Quitar un rol de seguridad para un usuario usando la colección `systemuserroles_association`.

Sección 5: Eliminar entidades de ejemplo

Operaciones: se agregó una referencia a cada registro creado en esta muestra a una lista a medida que se creaba. Esta sección recorre esa lista y elimina cada registro.

Limpiar

Por defecto esta muestra eliminará todos los registros creados en ella. Si desea ver los registros creados después de completar la muestra, cambie la variable `deleteCreatedRecords` a `false` y se le pedirá que decida si desea eliminar los registros.

Consulte también

[Usar la API web de Dataverse](#)

[Biblioteca de clases WebAPIService \(C#\)](#)

[Crear una fila de tabla usando la API web](#)

[Actualizar y eliminar filas de tablas usando la API web](#)

[Recuperar una fila de tabla usando la API web](#)

[Ejemplos de la API web](#)

[Ejemplo de operaciones básicas de la API web](#)

[Ejemplo de datos de consulta de API web \(C#\)](#)

[Ejemplo de operaciones condicionales de la API web \(C#\)](#)

[Ejemplo de funciones y acciones de la API web \(C#\)](#)

[Ejemplo de operaciones de metadatos de API web \(C#\)](#)

[Ejemplo de operaciones en paralelo de la API web WebApiService \(C#\)](#)

[Ejemplo de operaciones paralelas de API web con componentes de flujo de datos TPL \(C#\)](#)

 **Nota**

¿Puede indicarnos sus preferencias de idioma de documentación? [Realice una breve encuesta.](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad](#)).

Ejemplo de datos de consulta de API web (C#)

Artículo • 12/04/2023

Este ejemplo .NET 6.0 demuestra cómo realizar operaciones de datos comunes utilizando la API web de Dataverse.

Esta muestra utiliza el código auxiliar común de la [biblioteca de clases WebAPIService \(C#\)](#).

ⓘ Nota

Este ejemplo implementa las operaciones de Dataverse y la salida de la consola detalladas en el [Ejemplo de datos de consulta de la API web](#) y utiliza las construcciones comunes de C# que se describen en [Ejemplos de API web \(C#\)](#).

Requisitos previos

Los siguientes elementos son necesarios para compilar y ejecutar este ejemplo:

- Microsoft Visual Studio 2022.
- Acceso a Dataverse con privilegios para realizar operaciones de datos.

Cómo ejecutar esta muestra

1. Clonar o descargar el repositorio de [ejemplos de PowerApps](#).
2. Localizar la carpeta [/dataverse/webapi/C#-NETx/QueryData/](#).
3. Abrir el archivo `QueryData.sln` usando Visual Studio 2022
4. Editar el archivo `appsettings.json` para establecer los siguientes valores de propiedad:

Property	Instrucciones
<code>Url</code>	La URL para su entorno. Reemplazar el valor de marcador de posición <code>https://yourorg.api.crm.dynamics.com</code> con el valor para su entorno. Consulte Ver recursos para desarrolladores para encontrar esto.

Property	Instrucciones
UserPrincipalName	Reemplazar el valor de marcador de posición <code>you@yourorg.onmicrosoft.com</code> con el valor de UPN que usa para acceder al entorno.
Password	Reemplazar el valor de marcador de posición <code>yourPassword</code> con la contraseña que utiliza.

5. Guarde el archivo `appsettings.json`

6. Presione F5 para ejecutar el ejemplo.

Código

El código para este ejemplo está aquí: [PowerApps-Samples/dataverse/webapi/C#-NETx/QueryData/Program.cs](#)

Demostraciones

Esta muestra tiene 11 regiones:

Sección 0: Crear registros para consultar

Operaciones: Crear 1 registro `account` con 9 registros `contact` relacionados. Cada registro `contact` tiene 3 registros `task` relacionados.

Estos son los datos que se usarán en el ejemplo.

Sección 1 Seleccionar propiedades específicas

Operaciones:

- Usar `$select` en una entidad de contacto para obtener las propiedades que desea.
- Incluir anotaciones proporciona acceso a valores formateados con la anotación `@OData.Community.Display.V1.FormattedValue`

Sección 2 Uso de funciones de consulta

Operaciones:

- Uso de funciones de consulta estándar (`contains`, `endswith`, `startswith`) para filtrar los resultados.
- Uso de funciones de consulta de Dataverse (`LastXhours`, `Last7Days`, `Today`, `Between`, `In`)
- Uso de operadores de filtro y operadores lógicos (`eq`, `ne`, `gt`, `and`, `or`)
- Establecer prioridad usando paréntesis `((criteria1) and (criteria2)) or (criteria3)`

Sección 3 Pedidos y alias

Operaciones:

- Usar `$orderby`
- Uso de alias parametrizados (`?@p1=fullname`) con `$filter` y `$orderby`

Sección 4 Limitar y contar los resultados

Operaciones:

- Limitar los resultados usando `$top`.
- Obtenga un valor de conteo usando `$count`.

Sección 5 Paginación

Operaciones:

- Use el encabezado de solicitud `Prefer: odata.maxpagesize` para limitar el número de filas devueltas.
- Use la URL devuelta con la anotación `@odata.nextLink` para recuperar el siguiente conjunto de registros.

Sección 6 Expandir resultados

Operaciones:

- `$expand` con propiedades de navegación de un solo valor.
- `$expand` con la propiedad `partner`.
- `$expand` con propiedades de navegación con valores de colección.
- `$expand` con múltiples tipos de propiedades de navegación en una sola solicitud.
- `$expand` anidado.

- `$expand` anidado con propiedades de navegación de un solo valor y valoradas como colección.

Sección 7 Agregación de resultados

Operaciones: Uso de `$apply=aggregate` con `average`, `sum`, `min` y `max`.

Sección 8 Consultas FetchXML

Operaciones:

- Envío de solicitudes usando `fetchXml` con `?fetchXml=`
- Página usando los atributos `page` y `count`.

Sección 9 Utilizar consultas predefinidas

Operaciones:

- Uso de `{entitysetname}?savedQuery={savedqueryid}` para devolver los resultados de una consulta guardada (vista del sistema)
- Uso de `{entitysetname}?userQuery={userquery}` para devolver los resultados de una consulta de usuario (vista guardada)

Sección 10: Eliminar registros de muestra

Operaciones: se agregó una referencia a cada registro creado en esta muestra a una lista a medida que se creaba. En este ejemplo, todos los registros se eliminan mediante una operación `$batch`.

Consulte también

[Consultar datos utilizando la API web](#)

[Ejemplo de datos de consulta de la API web](#)

[Usar la API web de Dataverse Ejemplo de operaciones básicas de la API web \(C#\)](#)

[Ejemplo de operaciones condicionales de la API web \(C#\)](#)

[Ejemplo de funciones y acciones de la API web \(C#\)](#)

[Ejemplo de operaciones de metadatos de API web \(C#\)](#)

[Ejemplo de operaciones en paralelo de la API web Web ApiService \(C#\)](#)

[Ejemplo de operaciones paralelas de API web con componentes de flujo de datos TPL \(C#\)](#)

 **Nota**

¿Puede indicarnos sus preferencias de idioma de documentación? **Realice una breve encuesta.**  (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales (**declaración de privacidad** ).

Ejemplo de operaciones condicionales de la API web (C#)

Artículo • 23/02/2023

Este ejemplo .NET 6.0 demuestra cómo realizar operaciones de datos comunes utilizando la API web de Dataverse.

Esta muestra utiliza el código auxiliar común de la [biblioteca de clases WebAPIService \(C#\)](#).

ⓘ Nota

Este ejemplo implementa las operaciones de Dataverse y la salida de la consola detalladas en el [Ejemplo de operaciones condicionales de la API web](#) y utiliza las construcciones comunes de C# que se describen en [Ejemplos de API web \(C#\)](#).

Requisitos previos

Los siguientes elementos son necesarios para compilar y ejecutar este ejemplo:

- Microsoft Visual Studio 2022.
- Acceso a Dataverse con privilegios para realizar operaciones de datos.

Cómo ejecutar esta muestra

1. Clonar o descargar el repositorio de [ejemplos de PowerApps](#).
2. Localizar la carpeta [/dataverse/webapi/C#-NETx/ConditionalOperations/](#).
3. Abrir el archivo `ConditionalOperations.sln` usando Visual Studio 2022
4. Editar el archivo `appsettings.json` para establecer los siguientes valores de propiedad:

Property	Instrucciones
<code>Url</code>	La URL para su entorno. Reemplazar el valor de marcador de posición <code>https://yourorg.api.crm.dynamics.com</code> con el valor para su entorno. Consulte Ver recursos para desarrolladores para encontrar esto.

Property	Instrucciones
UserPrincipalName	Reemplazar el valor de marcador de posición <code>you@yourorg.onmicrosoft.com</code> con el valor de UPN que usa para acceder al entorno.
Password	Reemplazar el valor de marcador de posición <code>yourPassword</code> con la contraseña que utiliza.

5. Guarde el archivo `appsettings.json`

6. Presione F5 para ejecutar el ejemplo.

Código

El código para este ejemplo está aquí: [PowerApps-Samples/dataverse/webapi/C#-NETx/ConditionalOperations/Program.cs](#)

Demostraciones

Esta muestra tiene 4 regiones:

Sección 0: Crear registros de muestra

Operaciones:

- Crea un solo registro de cuenta.
- Recuperar el registro y almacena el valor de ETag como `initialAcctETagVal`.

Sección 1: GET condicional

Operaciones:

1. Intenta recuperar el registro configurando el encabezado `If-None-Match` con el valor `initialAcctETagVal`.

La solicitud falla con `304 NotModified` como se esperaba.

2. Actualizar el valor de la columna `telephone1` del registro.
3. Intente recuperar el registro de nuevo configurando el encabezado `If-None-Match` con el valor `initialAcctETagVal`.

La solicitud se realizó correctamente porque el valor inicial de ETag ya no coincide.

4. Guarde el nuevo valor de ETag como `updatedAcctETagVal`.

Sección 2: Simultaneidad optimista al eliminar y actualizar

Operaciones:

1. Intente eliminar la cuenta original configurando el encabezado `If-Match` con el valor `initialAcctETagVal`.

La solicitud falla con `412 PreconditionFailed` como se esperaba.

2. Intente actualizar la cuenta original configurando el encabezado `If-Match` con el valor `initialAcctETagVal`.

La solicitud falla con `412 PreconditionFailed` como se esperaba.

3. Intente actualizar la cuenta original configurando el encabezado `If-Match` con el valor `updatedAcctETagVal`.

La solicitud se realizó correctamente porque el valor de ETag coincide con el valor actual del registro.

Sección 3: Eliminar registros de muestra

Operaciones: se agregó una referencia a cada registro creado en esta muestra a una lista a medida que se creaba. Esta sección envía una solicitud `$batch` para la eliminación del registro.

Limpiar

Por defecto esta muestra eliminará todos los registros creados en ella. Si desea ver los registros creados después de completar la muestra, cambie la variable `deleteCreatedRecords` a `false` y se le pedirá que decida si desea eliminar los registros.

Consulte también

[Realizar operaciones condicionales mediante la API web](#)

[Usar la API web de Dataverse](#)

[Biblioteca de clases WebAPIService \(C#\)](#)

[Ejemplos de la API web](#)

[Ejemplo de operaciones básicas de la API web \(C#\)](#)

[Ejemplo de datos de consulta de API web \(C#\)](#)

[Ejemplo de funciones y acciones de la API web \(C#\)](#)

[Ejemplo de operaciones de metadatos de API web \(C#\)](#)

[Ejemplo de operaciones en paralelo de la API web WebAPIService \(C#\)](#)

[Ejemplo de operaciones paralelas de API web con componentes de flujo de datos TPL \(C#\)](#)

 **Nota**

¿Puede indicarnos sus preferencias de idioma de documentación? [Realice una breve encuesta.](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad](#)).

Ejemplo de funciones y acciones de la API web (C#)

Artículo • 23/02/2023

Este ejemplo .NET 6.0 demuestra cómo realizar operaciones de datos comunes utilizando la API web de Dataverse.

Esta muestra utiliza el código auxiliar común de la [biblioteca de clases WebAPIService \(C#\)](#).

ⓘ Nota

Este ejemplo implementa las operaciones de Dataverse y la salida de la consola detalladas en el [Ejemplo de funciones y acciones de la API web](#) y utiliza las construcciones comunes de C# que se describen en [Ejemplos de API web \(C#\)](#).

Requisitos previos

Los siguientes elementos son necesarios para compilar y ejecutar este ejemplo:

- Microsoft Visual Studio 2022.
- Acceso a Dataverse con privilegios para realizar operaciones de datos.

Cómo ejecutar esta muestra

1. Clonar o descargar el repositorio de [ejemplos de PowerApps](#).
2. Localizar la carpeta `/dataverse/webapi/C#-NETx/FunctionsAndActions/`.
3. Abrir el archivo `FunctionsAndActions.sln` usando Visual Studio 2022
4. Editar el archivo `appsettings.json` para establecer los siguientes valores de propiedad:

Property	Instrucciones
<code>Url</code>	La URL para su entorno. Reemplazar el valor de marcador de posición <code>https://yourorg.api.crm.dynamics.com</code> con el valor para su entorno. Consulte Ver recursos para desarrolladores para encontrar esto.

Property	Instrucciones
UserPrincipalName	Reemplazar el valor de marcador de posición <code>you@yourorg.onmicrosoft.com</code> con el valor de UPN que usa para acceder al entorno.
Password	Reemplazar el valor de marcador de posición <code>yourPassword</code> con la contraseña que utiliza.

5. Guarde el archivo `appsettings.json`

6. Presione F5 para ejecutar el ejemplo.

Código

El código para este ejemplo está aquí: [PowerApps-Samples/dataverse/webapi/C#-NETx/FunctionsAndActions/Program.cs ↗](#)

Demostraciones

Esta muestra tiene 9 regiones:

Sección 1: Funciones independientes WhoAmI

Operación: Enviar `WhoAmI Function` y recibir `WhoAmIResponse ComplexType`.

Sección 2: Funciones independientes FormatAddress

Operaciones:

1. Enviar `FormatAddress Function` con parámetros para una dirección en los Estados Unidos. Recibir `FormatAddressResponse ComplexType` con la dirección formateada.
2. Haga lo mismo con los parámetros para una dirección en Japón.

Sección 3: Funciones independientes InitializeFrom

Operaciones:

1. Cree un registro de cuenta para que sea el registro original.
2. Enviar `InitializeFrom Function` con parámetros que hacen referencia al registro de cuenta creado. Reciba una respuesta con datos para crear un nuevo registro de

cuenta con valores del registro original.

3. Cree un nuevo registro de cuenta utilizando los datos de `InitializeFromResponse` de modo que el nuevo registro se asocie con el registro original y posiblemente contenga datos copiados del registro original, según cómo se configuren las asignaciones de columnas para la organización.

Sección 4: Funciones independientes `RetrieveCurrentOrganization`

Operación: Enviar [RetrieveCurrentOrganization Function](#) y recibir [RetrieveCurrentOrganization ComplexType](#).

Sección 5: Funciones independientes `RetrieveTotalRecordCount`

Operaciones: Enviar [RetrieveTotalRecordCount Function](#) con parámetros para las tablas `account` y `contact`, y recibir [RetrieveTotalRecordCountResponse ComplexType](#) con el número de registros en cada tabla.

Sección 6: Funciones vinculadas: `IsSystemAdmin`

Operaciones:

1. Detectar si la organización tiene la API `sample_IsSystemAdmin` personalizada instalada.
2. Si no, instale la solución en `IsSystemAdminFunction_1_0_0_0_managed.zip` con la API personalizada.
3. Recuperar 10 registros de usuarios del sistema.
4. Recorra los registros usando la función `sample_IsSystemAdmin` para detectar cuales tienen el Sistema Administrador rol de seguridad.

Sección 7: Acciones sin consolidar `GrantAccess`

Operaciones:

1. Cree un registro de cuenta para compartir.
2. Recupere un usuario habilitado que no sea el usuario actual.
3. Utilice [RetrievePrincipalAccess Function](#) para determinar si el otro usuario tiene `AccessRights 'DeleteAccess'` en el registro de cuenta creado.

4. Si no tienen `DeleteAccess`, use `GrantAccess Action` para compartir este acceso con el otro usuario.
5. Pruebe el acceso del usuario nuevamente usando `RetrievePrincipalAccess` para verificar que ahora tienen `DeleteAccess`.

Sección 8: Acciones vinculadas: AddPrivilegesRole

Operaciones:

1. Cree un registro `role EntityType` asociado con la unidad de negocio de la persona que llama.
2. Recupere el registro de roles con registros expandidos `privilege EntityType` para mostrar los privilegios incluidos por defecto.
3. Recupere la información sobre los privilegios `prvCreateAccount` y `prvReadAccount`.
4. Use la información recuperada sobre esos privilegios para preparar una lista de instancias de `RolePrivilege ComplexType` para ser parámetros para `AddPrivilegesRole`.
5. Envíe `AddPrivilegesRole Action` con los parámetros `RolePrivilege`.
6. Recupere el registro de roles de nuevo con registros expandidos `privilege EntityType` para mostrar los privilegios ahora con los privilegios `prvCreateAccount` y `prvReadAccount`.

Sección 9: Eliminar registros de muestra

Operaciones: se agregó una referencia a cada registro creado en esta muestra a una lista a medida que se creaba. Esta sección envía una solicitud `$batch` para la eliminación del registro.

Limpiar

Por defecto esta muestra eliminará todos los registros creados en ella. Si desea ver los registros creados después de completar la muestra, cambie la variable `deleteCreatedRecords` a `false` y se le pedirá que decida si desea eliminar los registros.

Consulte también

[Realizar operaciones condicionales mediante la API web](#)
[Usar la API web de Dataverse](#)
[Biblioteca de clases WebAPIService \(C#\)](#)

Ejemplos de la API web

[Ejemplo de operaciones básicas de la API web \(C#\)](#)

[Ejemplo de datos de consulta de API web \(C#\)](#)

[Ejemplo de operaciones condicionales de la API web \(C#\)](#)

[Ejemplo de operaciones de metadatos de API web \(C#\)](#)

[Ejemplo de operaciones en paralelo de la API web WebApiService \(C#\)](#)

[Ejemplo de operaciones paralelas de API web con componentes de flujo de datos TPL \(C#\)](#)

Nota

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta. [↗](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad ↗](#)).

Ejemplo de operaciones de metadatos de API web (C#)

Artículo • 23/02/2023

Este ejemplo de .NET 6.0 muestra cómo realizar operaciones que crean y modifican definiciones de tablas, columnas y relaciones mediante la API web de Dataverse.

Esta muestra utiliza el código auxiliar común de la [biblioteca de clases WebAPIService \(C#\)](#).

ⓘ Nota

Este ejemplo implementa las operaciones de Dataverse y la salida de la consola detalladas en el [Ejemplo de operaciones de metadatos de API web](#) y utiliza las construcciones comunes de C# que se describen en [Ejemplos de API web \(C#\)](#).

Requisitos previos

Los siguientes elementos son necesarios para compilar y ejecutar este ejemplo:

- Microsoft Visual Studio 2022.
- Acceso a Dataverse con privilegios para realizar operaciones de datos.

Cómo ejecutar esta muestra

1. Clonar o descargar el repositorio de [ejemplos de PowerApps](#).
2. Localizar la carpeta [/dataverse/webapi/C#-NETx/MetadataOperations/](#).
3. Abrir el archivo `MetadataOperations.sln` usando Visual Studio 2022
4. Editar el archivo `appsettings.json` para establecer los siguientes valores de propiedad:

Property	Instrucciones
<code>Url</code>	La URL para su entorno. Reemplazar el valor de marcador de posición <code>https://yourorg.api.crm.dynamics.com</code> con el valor para su entorno. Consulte Ver recursos para desarrolladores para encontrar esto.

Property	Instrucciones
UserPrincipalName	Reemplazar el valor de marcador de posición <code>you@yourorg.onmicrosoft.com</code> con el valor de UPN que usa para acceder al entorno.
Password	Reemplazar el valor de marcador de posición <code>yourPassword</code> con la contraseña que utiliza.

5. Guarde el archivo `appsettings.json`

6. Presione F5 para ejecutar el ejemplo.

Código

El código para este ejemplo está aquí: [PowerApps-Samples/dataverse/webapi/C#-NETx/MetadataOperations/Program.cs](#) ↗

Demostraciones

Esta muestra tiene 11 regiones:

Sección 0: Crear un editor y una solución

Operaciones: cree un registro de solución y un registro de editor asociado.

ⓘ Nota

Todos los componentes de la solución creados en este ejemplo se asociarán a la solución para que puedan exportarse. Para operaciones sin un mensaje con nombre, esta asociación se crea usando el encabezado de solicitud `MSCRM.SolutionUniqueName` que establece el nombre único de la solución establecido como valor. Todos los nombres de los componentes de la solución tienen el prefijo de personalización del editor.

Sección 1: Crear, recuperar y actualizar una tabla

Operaciones:

1. Crear una nueva tabla propiedad del usuario `sample_BankAccount` enviando una solicitud `POST` a `/EntityDefinitions`.

2. Recuperar la tabla creada enviando una solicitud `GET` a
`/EntityDefinitions(LogicalName='sample_bankaccount')`.

3. Actualizar la tabla creada enviando una solicitud `PUT` a
`/EntityDefinitions(LogicalName='sample_bankaccount')`.

Sección 2: Crear, recuperar y actualizar columnas

Operaciones:

1. Crear una nueva columna booleana `sample_boolean` para la tabla
`sample_BankAccount` enviando una solicitud `POST` a
`/EntityDefinitions(LogicalName='sample_bankaccount')/Attributes`.

2. Recuperar la columna booleana `sample_boolean` enviando una solicitud `GET` a
`/EntityDefinitions(LogicalName='sample_bankaccount')/Attributes(LogicalName='sample_boolean')`.

3. Actualizar la columna booleana `sample_boolean` enviando una solicitud `PUT` a
`/EntityDefinitions(LogicalName='sample_bankaccount')/Attributes(LogicalName='sample_boolean')`.

4. Actualizar las etiquetas de opción para la columna booleana `sample_boolean` usando [UpdateOptionValue Action](#).

5. Crear y recuperar una nueva columna datetime `sample_datetime` para la tabla
`sample_BankAccount`.

6. Crear y recuperar una nueva columna decimal `sample_decimal` para la tabla
`sample_BankAccount`.

7. Crear y recuperar una nueva columna de enteros `sample_integer` para la tabla
`sample_BankAccount`.

8. Crear y recuperar una nueva columna de notas `sample_memo` para la tabla
`sample_BankAccount`.

9. Crear y recuperar una nueva columna de dinero `sample_money` para la tabla
`sample_BankAccount`.

10. Crear y recuperar una nueva columna de opción `sample_choice` para la tabla
`sample_BankAccount`.

11. Agregue una opción nueva a la columna `sample_choice` usando [InsertOptionValue Action](#).

12. Cambie el orden de las opciones de la columna `sample_choice` usando [OrderOption Action](#).

13. Elimine una de las opciones de la columna `sample_choice` usando [DeleteOptionValue Action](#).

14. Cree y recupere una nueva columna de selección múltiple `sample_multiselectchoice` para la tabla `sample_BankAccount`.
15. Cree una nueva opción de Estado para la tabla `sample_BankAccount` usando [InsertStatusValue Action](#).

Sección 3: Crear y usar OptionSet global

Operaciones:

1. Cree una nueva opción global denominada `sample_colors` enviando una solicitud `POST` a `/GlobalOptionSetDefinitions`.
2. Recupere la opción global `sample_colors` enviando una solicitud `GET` a `/GlobalOptionSetDefinitions(<id value>)`.
3. Cree una nueva columna de elección `sample_colors` para la tabla `sample_BankAccount` usando la elección global `sample_colors` enviando una solicitud `POST` a `/EntityDefinitions(LogicalName='sample_bankaccount')/Attributes` y asociándola a la elección global.

Sección 4: Crear relación con el cliente

Operaciones:

1. Cree una nueva columna de cliente `sample_customerid` para la tabla `sample_BankAccount` usando [CreateCustomerRelationships Action](#).
2. Recupere la columna de cliente `sample_customerid` enviando una solicitud `GET` a `/EntityDefinitions(LogicalName='sample_bankaccount')/Attributes(LogicalName='sample_customerid')`.
3. Recupere el Relaciones creado para la columna del cliente enviando solicitudes `GET` a `/RelationshipDefinitions(<id>)/Microsoft.Dynamics.CRM.OneToManyRelationshipMetadata``.

Sección 5: Crear y recuperar una relación de uno a muchos

Operaciones:

1. Verifique que la tabla `sample_BankAccount` es elegible para ser referenciada en una relación 1:N usando [CanBeReferenced Function](#).

2. Verifique que la tabla `contact` es elegible para hacer referencia a otras tablas en una relación 1:N usando [CanBeReferencing Function](#).
3. Identifique qué otras tablas pueden hacer referencia a la tabla `sample_BankAccount` en una relación 1:N usando [GetValidReferencingEntities Function](#).
4. Cree una relación 1:N entre las tablas `sample_BankAccount` y `contact` enviando una solicitud `POST` a `/RelationshipDefinitions`.
5. Recupere la relación 1:N enviando una solicitud `GET` a
`/RelationshipDefinitions(<id>)/Microsoft.Dynamics.CRM.OneToManyRelationshipMetadata`.

Sección 6: Crear y recuperar una relación de muchos a uno

Operaciones:

1. Cree una relación N:1 entre las tablas `sample_BankAccount` y `account` enviando una solicitud `POST` a `/RelationshipDefinitions`.
2. Recupere la relación N:1 enviando una solicitud `GET` a
`/RelationshipDefinitions(<id>)/Microsoft.Dynamics.CRM.OneToManyRelationshipMetadata`.

Sección 7: Crear y recuperar una relación de muchos a muchos

Operaciones:

1. Verifique que las tablas `sample_BankAccount` y `contact` son elegibles para participar en una relación N:N usando [CanManyToMany Function](#).
2. Verifique que las tablas `sample_BankAccount` y `contact` son elegibles para participar en una relación N:N usando [GetValidManyToMany Function](#).
3. Cree una relación N:N entre las tablas `sample_BankAccount` y `contact` enviando una solicitud `POST` a `/RelationshipDefinitions`.
4. Recupere la relación N:N enviando una solicitud `GET` a
`/RelationshipDefinitions(<id>)/Microsoft.Dynamics.CRM.ManyToManyRelationshipMetadata`.

Sección 8: Exportar una solución administrada

Operaciones: Exportar la solución creada en [Sección 0: Crear publicador y solución que contiene los elementos creados en esta muestra](#) usando el [ExportSolution Action](#).

Sección 9: Eliminar registros de muestra

Operaciones: se agregó una referencia a cada registro creado en esta muestra a una lista a medida que se creaba. En este ejemplo, todos los registros se eliminan mediante una operación `$batch`.

Sección 10: Importar y eliminar una solución administrada

Operaciones:

1. Importe la solución exportada en la [Sección 8](#) utilizando [ImportSolution Action](#).
2. Consulte la tabla de soluciones para obtener el `solutionid` de la solución importada.
3. Elimine la solución importada usando el `solutionid`.

Limpiar

Por defecto esta muestra eliminará todos los registros creados en ella. Si desea ver los registros creados después de completar la muestra, cambie la variable `deleteCreatedRecords` a `false` y se le pedirá que decida si desea eliminar los registros.

Consulte también

[Usar la API web de Dataverse](#)

[Biblioteca de clases WebAPIService \(C#\)](#)

[Utilizar la API web con definiciones de tabla](#)

[Ejemplos de la API web](#)

[Ejemplo de operaciones básicas de la API web \(C#\)](#)

[Ejemplo de datos de consulta de API web \(C#\)](#)

[Ejemplo de operaciones condicionales de la API web \(C#\)](#)

[Ejemplo de funciones y acciones de la API web \(C#\)](#)

[Ejemplo de operaciones en paralelo de la API web WebAPIService \(C#\)](#)

[Ejemplo de operaciones paralelas de API web con componentes de flujo de datos TPL \(C#\)](#)

 **Nota**

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta. [↗](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales (**declaración de privacidad** [↗](#)).

Ejemplo de operaciones en paralelo de la API web Web ApiService (C#)

Artículo • 23/02/2023

Este ejemplo .NET 6.0 demuestra cómo realizar operaciones de datos paralelos utilizando la API web de Dataverse.

Esta muestra utiliza el código auxiliar común de la [biblioteca de clases Web ApiService \(C#\)](#).

Requisitos previos

Los siguientes elementos son necesarios para compilar y ejecutar este ejemplo:

- Microsoft Visual Studio 2022.
- Acceso a Dataverse con privilegios para realizar operaciones de datos.

Cómo ejecutar esta muestra

1. Clonar o descargar el repositorio de [ejemplos de PowerApps](#).
2. Localice la carpeta `/dataverse/webapi/C#-NETx/ParallelOperations/`.
3. Abrir el archivo `ParallelOperations.sln` usando Visual Studio 2022
4. Editar el archivo `appsettings.json` para establecer los siguientes valores de propiedad:

Property	Instrucciones
<code>Url</code>	La URL para su entorno. Reemplazar el valor de marcador de posición <code>https://yourorg.api.crm.dynamics.com</code> con el valor para su entorno. Consulte Ver recursos para desarrolladores para encontrar esto.
<code>UserPrincipalName</code>	Reemplazar el valor de marcador de posición <code>you@yourorg.onmicrosoft.com</code> con el valor de UPN que usa para acceder al entorno.
<code>Password</code>	Reemplazar el valor de marcador de posición <code>yourPassword</code> con la contraseña que utiliza.

5. Guarde el archivo `appsettings.json`

6. Presione F5 para ejecutar el ejemplo.

Código

El código para este ejemplo está aquí: [PowerApps-Samples/dataverse/webapi/C#-NETx/ParallelOperations/Program.cs ↗](#)

Demostraciones

Esta muestra incluye configuraciones que puede aplicar para optimizar su conexión.

Esta muestra primero envía una solicitud simplemente para acceder al valor del encabezado de respuesta `x-ms-dop-hint` para determinar los grados recomendados de paralelismo para este entorno. Cuando el grado máximo de paralelismo se establece igual al valor del encabezado de respuesta `x-ms-dop-hint`, debe lograr un estado estable en el que el rendimiento se optimice con un mínimo de `429 TooManyRequests` se devolvieron errores de límite de protección de servicio.

Para encontrar límites de protección de servicio con esta muestra, debe aumentar la variable `numberOfRecords` a más de 10.000 o lo que sea necesario para que la muestra funcione durante más de 5 minutos. También debe cambiar el código para configurar `maxDegreeOfParallelism` ser significativamente mayor que el valor de `x-ms-dop-hint` del encabezado de respuesta. Luego, al usar Fiddler, debería poder observar cómo WebAPIService vuelve a intentar las solicitudes que devuelven este error.

Este ejemplo utiliza el [método Parallel.ForEachAsync](#) introducido con .NET 6.0.

Esta muestra procesa una lista de solicitudes para crear registros de cuentas, envía las solicitudes en paralelo y luego usa los datos devueltos para agregar solicitudes para eliminar las cuentas creadas en un [ConcurrentBag](#). Después de crear los registros, se muestra la cantidad de segundos para crear los registros.

Luego, las solicitudes de eliminación de `ConcurrentBag` se procesan y se muestra el tiempo empleado en eliminar los registros.

Es posible que desee comparar esta muestra con el [ejemplo de componentes de flujo de datos TPL en operaciones paralelas de la API web \(C#\)](#).

Limpiar

Por defecto esta muestra eliminará todos los registros creados en ella. Si desea ver los registros creados después de completar la muestra, cambie la variable `deleteCreatedRecords` a `false` y se le pedirá que decida si desea eliminar los registros.

Consulte también

[Usar la API web de Dataverse](#)

[Biblioteca de clases WebAPIService \(C#\)](#)

[Crear una fila de tabla usando la API web](#)

[Actualizar y eliminar filas de tablas usando la API web](#)

[Límites de la API de protección de servicio](#)

[Ejemplos de la API web](#)

[Ejemplo de operaciones básicas de la API web \(C#\)](#)

[Ejemplo de datos de consulta de API web \(C#\)](#)

[Ejemplo de operaciones condicionales de la API web \(C#\)](#)

[Ejemplo de funciones y acciones de la API web \(C#\)](#)

[Ejemplo de operaciones de metadatos de API web \(C#\)](#)

[Ejemplo de operaciones paralelas de API web con componentes de flujo de datos TPL \(C#\)](#)

Nota

¿Puede indicarnos sus preferencias de idioma de documentación? [Realice una breve encuesta](#).[↗] (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad](#)[↗]).

Ejemplo de operaciones paralelas de API web con componentes de flujo de datos TPL (C#)

Artículo • 23/02/2023

Este ejemplo de .NET 6.0 demuestra cómo realizar operaciones de datos paralelas utilizando la API web de Dataverse con componentes de flujo de datos de biblioteca paralela de tareas (TPL). Más información: [Flujo de datos \(biblioteca paralela de tareas\)](#).

Esta muestra utiliza el código auxiliar común de la [biblioteca de clases WebAPIService \(C#\)](#).

Requisitos previos

Los siguientes elementos son necesarios para compilar y ejecutar este ejemplo:

- Microsoft Visual Studio 2022.
- Acceso a Dataverse con privilegios para realizar operaciones de datos.

Cómo ejecutar esta muestra

1. Clonar o descargar el repositorio de [ejemplos de PowerApps](#).
2. Localizar la carpeta `/dataverse/webapi/C#-NETx/TPLDataFlowParallelOperations/`.
3. Abrir el archivo `TPLDataFlowParallelOperations.sln` usando Visual Studio 2022
4. Editar el archivo `appsettings.json` para establecer los siguientes valores de propiedad:

Property	Instrucciones
<code>Url</code>	La URL para su entorno. Reemplazar el valor de marcador de posición <code>https://yourorg.api.crm.dynamics.com</code> con el valor para su entorno. Consulte Ver recursos para desarrolladores para encontrar esto.
<code>UserPrincipalName</code>	Reemplazar el valor de marcador de posición <code>you@yourorg.onmicrosoft.com</code> con el valor de UPN que usa para acceder al entorno.

Property	Instrucciones
Password	Reemplazar el valor de marcador de posición <code>yourPassword</code> con la contraseña que utiliza.

5. Guarde el archivo `appsettings.json`

6. Presione F5 para ejecutar el ejemplo.

Código

El código para este ejemplo está aquí: [PowerApps-Samples/dataverse/webapi/C#-NETx/TPLDataFlowParallelOperations/Program.cs ↗](#)

Demostraciones

Esta muestra incluye configuraciones que puede aplicar para optimizar su conexión.

Esta muestra primero envía una solicitud simplemente para acceder al valor del encabezado de respuesta `x-ms-dop-hint` para determinar los grados recomendados de paralelismo para este entorno. Cuando el grado máximo de paralelismo se establece igual al valor del encabezado de respuesta `x-ms-dop-hint`, debe lograr un estado estable en el que el rendimiento se optimice con un mínimo de `429 TooManyRequests` se devolvieron errores de límite de protección de servicio.

Para encontrar límites de protección de servicio con esta muestra, debe aumentar la variable `numberOfRecords` a más de 10.000 o lo que sea necesario para que la muestra funcione durante más de 5 minutos. También debe cambiar el código para configurar `maxDegreeOfParallelism` ser significativamente mayor que el valor de `x-ms-dop-hint` del encabezado de respuesta. Luego, al usar Fiddler, debería poder observar cómo WebAPIService vuelve a intentar las solicitudes que devuelven este error.

Esta muestra simplemente crea un número configurable de registros de cuenta, que a su vez eliminará. Este ejemplo utiliza componentes de flujo de datos para procesar los registros y transformar los resultados de la operación de creación en la siguiente fase que elimina estos registros. Debido a la naturaleza de este flujo de datos, las operaciones de eliminación de registros creados anteriormente comenzarán antes de que finalicen todos los registros a crear.

Es posible que desee comparar esta muestra con el [ejemplo de operaciones paralelas de CDSWebService de la API web \(C#\)](#).

Limpiar

Por defecto esta muestra eliminará todos los registros creados en ella. Si desea ver los registros creados después de completar la muestra, cambie la variable `deleteCreatedRecords` a `false` y se le pedirá que decida si desea eliminar los registros.

Consulte también

[Usar la API web de Dataverse](#)

[Biblioteca de clases WebAPIService \(C#\)](#)

[Crear una fila de tabla usando la API web](#)

[Actualizar y eliminar filas de tablas usando la API web](#)

[Límites de la API de protección de servicio](#)

[Ejemplos de la API web](#)

[Ejemplo de operaciones básicas de la API web \(C#\)](#)

[Ejemplo de datos de consulta de API web \(C#\)](#)

[Ejemplo de operaciones condicionales de la API web \(C#\)](#)

[Ejemplo de funciones y acciones de la API web \(C#\)](#)

[Ejemplo de operaciones de metadatos de API web \(C#\)](#)

[Ejemplo de operaciones en paralelo de la API web WebApiService \(C#\)](#)

Nota

¿Puede indicarnos sus preferencias de idioma de documentación? [Realice una breve encuesta.](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad](#)).

Ejemplo: Operaciones de archivo usando la API web de Dataverse

Artículo • 19/04/2023

Este ejemplo .NET 6.0 demuestra cómo realizar operaciones con columnas de archivo utilizando la API web de Dataverse.

Esta muestra utiliza el código auxiliar común de la [biblioteca de clases WebAPIService \(C#\)](#).

Requisitos previos

- Microsoft Visual Studio 2022
- Accesa a Dataverse con Administrador del sistema o privilegios de personalizador del sistema.

Cómo ejecutar el ejemplo

1. Clonar o descargar el repositorio de [ejemplos de PowerApps](#).
2. Abra el archivo `PowerApps-Samples\dataverse\webapi\C#-NETx\FileOperations\FileOperations.sln` usando Visual Studio 2022.
3. Editar el archivo `appsettings.json` para establecer los siguientes valores de propiedad:

Property	Instrucciones
<code>Url</code>	La URL para su entorno. Reemplazar el valor de marcador de posición <code>https://yourorg.api.crm.dynamics.com</code> con el valor para su entorno. Consulte Ver recursos para desarrolladores para encontrar esto.
<code>UserPrincipalName</code>	Reemplazar el valor de marcador de posición <code>you@yourorg.onmicrosoft.com</code> con el valor de UPN que usa para acceder al entorno.
<code>Password</code>	Reemplazar el valor de marcador de posición <code>yourPassword</code> con la contraseña que utiliza.

4. Guarde el archivo `appsettings.json`

5. Seleccione el proyecto que quiere ejecutar en el explorador de soluciones. Haga clic derecho en el proyecto y elija **Establecer como proyecto de inicio**.

6. Presione F5 para ejecutar el ejemplo.

Salida de muestra

El resultado del ejemplo deberá ser ahora similar a esto:

```
Creating file column named 'sample_FileColumn' on the account table ...
Created file column named 'sample_FileColumn' in the account table.
Created account record with accountid:5fb4f993-7c55-ed11-bba3-000d3a9933c9
Uploading file Files\25mb.pdf ...
Uploaded file Files\25mb.pdf
Downloading file from accounts(5fb4f993-7c55-ed11-bba3-
000d3a9933c9)/sample_filecolumn ...
Downloaded the file to E:\GitHub\PowerApps-Samples\dataverse\webapi\C#-
NETx\FileOperations\FileOperationsWithActions\bin\Debug\net6.0//downloaded-
25mb.pdf.
Deleted the file using FileId.
Deleted the account record.
Deleting the file column named 'sample_filecolumn' on the account table ...
Deleted the file column named 'sample_filecolumn' in the account table.
```

Demostraciones

Este ejemplo es una solución con tres proyectos. Cada proyecto realiza las mismas operaciones de manera diferente. Consulte los archivos LÉAME respectivos para obtener detalles sobre cada proyecto.

- [LÉAME: Operaciones de archivo de la API web con ejemplo de acciones ↗](#)
- [LÉAME: Operaciones de archivo de la API web con ejemplo de intervalos ↗](#)
- [LÉAME: Operaciones de archivo de la API web con ejemplo de secuencias ↗](#)

El código de cada proyecto se encuentra en el archivo `Program.cs` correspondiente.

- [Program.cs: Operaciones de archivo de la API web con ejemplo de acciones ↗](#)
- [Program.cs: Operaciones de archivo de la API web con ejemplo de intervalos ↗](#)
- [Program.cs: Operaciones de archivo de la API web con ejemplo de secuencias ↗](#)

Cada proyecto utiliza una clase `Utility` para realizar operaciones comunes. Esta clase está en el archivo [Utility.cs ↗](#).

En general, cada proyecto realiza estas operaciones:

Crear una columna de archivo

La función `Utility.CreateFileColumn` crea una columna de archivo llamada `sample_FileColumn` en la tabla de cuentas con un valor `MaxSizeInKb` de 10 MB.

Actualizar una columna de archivo

La función `Utility.UpdateFileColumnMaxSizeInKB` actualiza el valor `MaxSizeInKb` de la columna de archivo `sample_FileColumn` a 100 MB.

Consejo: Si desea crear algunos escenarios de error porque el tamaño de la columna del archivo es demasiado pequeño, comente esta línea.

Recuperar el valor `MaxSizeInKb` de la columna del archivo

`Utility.GetFileColumnMaxSizeInKb` recupera el valor `MaxSizeInKb` valor de la columna de archivo `sample_FileColumn` y la almacena en una variable llamada `fileColumnMaxSizeInKb`.

Crear un registro contable

Antes de que se pueda cargar un archivo en la columna de archivo, debe existir un registro.

Cargar un archivo

La función para cargar el archivo acepta un parámetro llamado `fileColumnMaxSizeInKb` y usa ese valor para probar el tamaño del archivo. Si el archivo es más grande que el límite configurado de la columna del archivo, arrojará un error.

Descargar el archivo

Si la carga del archivo tuvo éxito, la función para descargar el archivo lo guardará en el directorio actual. Puede intentar abrir el archivo para confirmar que se cargó y descargó correctamente.

Eliminar el archivo

Si la carga del archivo se realizó correctamente, el archivo se elimina.

Limpiar

Para dejar el sistema en el estado anterior a la ejecución del ejemplo, hace lo siguiente:

- Eliminar el registro de cuenta
- Eliminar la columna de archivo

Consulte también

[Usar datos de columna de archivo](#)

[Ejemplo: Operaciones de archivo usando la SDK de Dataverse para .NET](#)

[Usar datos de columna de imagen](#)

[Ejemplo: Operaciones de imagen usando la API web de Dataverse](#)

Nota

¿Puede indicarnos sus preferencias de idioma de documentación? [Realice una breve encuesta.](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad](#)).

Ejemplo: Operaciones de imagen usando la API web de Dataverse

Artículo • 19/04/2023

Este ejemplo .NET 6.0 demuestra cómo realizar operaciones con columnas de imagen utilizando la API web de Dataverse.

Esta muestra utiliza el código auxiliar común de la [biblioteca de clases WebAPIService \(C#\)](#).

Requisitos previos

- Microsoft Visual Studio 2022
- Accesa a Dataverse con Administrador del sistema o privilegios de personalizador del sistema.

Cómo ejecutar el ejemplo

1. Clonar o descargar el repositorio de [ejemplos de PowerApps](#).
2. Abra el archivo `PowerApps-Samples\dataverse\webapi\C#-NETx\ImageOperations\ImageOperations.sln` usando Visual Studio 2022.
3. Editar el archivo `appsettings.json` para establecer los siguientes valores de propiedad:

Property	Instrucciones
<code>Url</code>	La URL para su entorno. Reemplazar el valor de marcador de posición <code>https://yourorg.api.crm.dynamics.com</code> con el valor para su entorno. Consulte Ver recursos para desarrolladores para encontrar esto.
<code>UserPrincipalName</code>	Reemplazar el valor de marcador de posición <code>you@yourorg.onmicrosoft.com</code> con el valor de UPN que usa para acceder al entorno.
<code>Password</code>	Reemplazar el valor de marcador de posición <code>yourPassword</code> con la contraseña que utiliza.

4. Guarde el archivo `appsettings.json`

5. Presione F5 para ejecutar el ejemplo.

Salida de muestra

El resultado del ejemplo deberá ser ahora similar a esto:

```
Creating image column named 'sample_ImageColumn' on the account table ...
Created image column named 'sample_ImageColumn' in the account table.
Create 5 records while CanStoreFullImage is false.
    Created account: 'CanStoreFullImage false 144x144.png'
    Created account: 'CanStoreFullImage false 144x400.png'
    Created account: 'CanStoreFullImage false 400x144.png'
    Created account: 'CanStoreFullImage false 400x500.png'
    Created account: 'CanStoreFullImage false 60x80.png'
Set the CanStoreFullImage property to True
Create 5 records while CanStoreFullImage is true.
    Created account: 'CanStoreFullImage true 144x144.png'
    Created account: 'CanStoreFullImage true 144x400.png'
    Created account: 'CanStoreFullImage true 400x144.png'
    Created account: 'CanStoreFullImage true 400x500.png'
    Created account: 'CanStoreFullImage true 60x80.png'
Retrieving records with thumbnail images:
    Thumbnail-sized file column data saved to
DownloadedImages\CanStoreFullImage false 144x144.png_retrieved.png
    Thumbnail-sized file column data saved to
DownloadedImages\CanStoreFullImage false 144x400.png_retrieved.png
    Thumbnail-sized file column data saved to
DownloadedImages\CanStoreFullImage false 400x144.png_retrieved.png
    Thumbnail-sized file column data saved to
DownloadedImages\CanStoreFullImage false 400x500.png_retrieved.png
    Thumbnail-sized file column data saved to
DownloadedImages\CanStoreFullImage false 60x80.png_retrieved.png
    Thumbnail-sized file column data saved to
DownloadedImages\CanStoreFullImage true 144x144.png_retrieved.png
    Thumbnail-sized file column data saved to
DownloadedImages\CanStoreFullImage true 144x400.png_retrieved.png
    Thumbnail-sized file column data saved to
DownloadedImages\CanStoreFullImage true 400x144.png_retrieved.png
    Thumbnail-sized file column data saved to
DownloadedImages\CanStoreFullImage true 400x500.png_retrieved.png
    Thumbnail-sized file column data saved to
DownloadedImages\CanStoreFullImage true 60x80.png_retrieved.png
Attempt to download full-size images for all 10 records using 3 different
methods:
Download full-sized files with actions: 5 should fail
    Download failed: No FileAttachment records found for
imagedescriptorId: e40bdccf1-598d-ed11-81ad-000d3a9933c9 for image attribute:
sample_imagecolumn of account record with id e30bdccf1-598d-ed11-81ad-
000d3a9933c9
    Download failed: No FileAttachment records found for
imagedescriptorId: 2258d4f7-598d-ed11-81ad-000d3a9933c9 for image attribute:
sample_imagecolumn of account record with id 2158d4f7-598d-ed11-81ad-
000d3a9933c9
```

```
        Download failed: No FileAttachment records found for
imagedescriptorId: 2658d4f7-598d-ed11-81ad-000d3a9933c9 for image attribute:
sample_imagecolumn of account record with id 2558d4f7-598d-ed11-81ad-
000d3a9933c9
        Download failed: No FileAttachment records found for
imagedescriptorId: 2a58d4f7-598d-ed11-81ad-000d3a9933c9 for image attribute:
sample_imagecolumn of account record with id 2958d4f7-598d-ed11-81ad-
000d3a9933c9
        Download failed: No FileAttachment records found for
imagedescriptorId: 2e58d4f7-598d-ed11-81ad-000d3a9933c9 for image attribute:
sample_imagecolumn of account record with id 2d58d4f7-598d-ed11-81ad-
000d3a9933c9
        Full-sized file downloaded to DownloadedImages\CanStoreFullImage
true 144x144.png_downloaded_with_actions.png
        Full-sized file downloaded to DownloadedImages\CanStoreFullImage
true 144x400.png_downloaded_with_actions.png
        Full-sized file downloaded to DownloadedImages\CanStoreFullImage
true 400x144.png_downloaded_with_actions.png
        Full-sized file downloaded to DownloadedImages\CanStoreFullImage
true 400x500.png_downloaded_with_actions.png
        Full-sized file downloaded to DownloadedImages\CanStoreFullImage
true 60x80.png_downloaded_with_actions.png
Download full-sized files with chunks: 5 should fail
        No full-sized image data returned because record was created while
CanStoreFullImage was false.
        No full-sized image data returned because record was created while
CanStoreFullImage was false.
        No full-sized image data returned because record was created while
CanStoreFullImage was false.
        No full-sized image data returned because record was created while
CanStoreFullImage was false.
        No full-sized image data returned because record was created while
CanStoreFullImage was false.
        Full-sized file downloaded to DownloadedImages\CanStoreFullImage
true 144x144.png_downloaded_with_chunks_full-sized.png
        Full-sized file downloaded to DownloadedImages\CanStoreFullImage
true 144x400.png_downloaded_with_chunks_full-sized.png
        Full-sized file downloaded to DownloadedImages\CanStoreFullImage
true 400x144.png_downloaded_with_chunks_full-sized.png
        Full-sized file downloaded to DownloadedImages\CanStoreFullImage
true 400x500.png_downloaded_with_chunks_full-sized.png
        Full-sized file downloaded to DownloadedImages\CanStoreFullImage
true 60x80.png_downloaded_with_chunks_full-sized.png
Download full-sized files in single requests: 5 should fail
        No full-sized image data returned because record was created while
CanStoreFullImage was false.
        No full-sized image data returned because record was created while
CanStoreFullImage was false.
        No full-sized image data returned because record was created while
CanStoreFullImage was false.
        No full-sized image data returned because record was created while
CanStoreFullImage was false.
        No full-sized image data returned because record was created while
CanStoreFullImage was false.
        Full-sized file downloaded to DownloadedImages\CanStoreFullImage
```

```
true 144x144.png_downloaded_with_stream_full-sized.png
    Full-sized file downloaded to DownloadedImages\CanStoreFullImage
true 144x400.png_downloaded_with_stream_full-sized.png
    Full-sized file downloaded to DownloadedImages\CanStoreFullImage
true 400x144.png_downloaded_with_stream_full-sized.png
    Full-sized file downloaded to DownloadedImages\CanStoreFullImage
true 400x500.png_downloaded_with_stream_full-sized.png
    Full-sized file downloaded to DownloadedImages\CanStoreFullImage
true 60x80.png_downloaded_with_stream_full-sized.png
Deleting the image data from the columns using 3 different methods:
    CanStoreFullImage false 144x144.png sample_imagecolumn deleted with
PATCH
    CanStoreFullImage false 144x400.png sample_imagecolumn deleted with
PATCH
    CanStoreFullImage false 400x144.png sample_imagecolumn deleted with
PATCH
    CanStoreFullImage false 400x500.png sample_imagecolumn deleted with
PUT
    CanStoreFullImage false 60x80.png sample_imagecolumn deleted with
PUT
    CanStoreFullImage true 144x144.png sample_imagecolumn deleted with
PUT
    CanStoreFullImage true 144x400.png sample_imagecolumn deleted with
DELETE
    CanStoreFullImage true 400x144.png sample_imagecolumn deleted with
DELETE
    CanStoreFullImage true 400x500.png sample_imagecolumn deleted with
DELETE
    CanStoreFullImage true 60x80.png sample_imagecolumn deleted with
DELETE
Deleted the account records created for this sample.
Deleting the image column named 'sample_ImageColumn' on the account table
...
Deleted the image column named 'sample_ImageColumn' in the account table.
Sample completed.
```

Demostraciones

El código para este ejemplo se incluye en el archivo [Program.cs](#).

El proyecto usa una clase `utility` para realizar operaciones que impliquen la creación o recuperación de datos de esquema. Esta clase está en el archivo [Utility.cs](#).

Este proyecto realiza estas operaciones:

Crear una columna de imagen automática

Esta muestra necesita crear una nueva columna de imagen que sea la imagen principal para la tabla de cuentas. También debe devolver el sistema al estado original cuando

haya terminado. Así que el programa hace estas cosas al principio:

1. Capture el nombre de la imagen principal original con el método

```
Utility.GetTablePrimaryImageName .
```

2. Utilice el método `Utility.CreateImageColumn` para crear una nueva columna de imagen llamada `sample_ImageColumn` en la tabla de cuentas si aún no existe.

⚠ Nota

El valor de esta columna de imagen `CanStoreFullImage` es falso.

3. Utilice el método `Utility.SetTablePrimaryImageName` para hacer `sample_ImageColumn` la columna de imagen primaria.

Crear registros de cuenta con datos de imagen

1. El programa recorre una lista de cinco nombres de archivo que coinciden con los nombres de los archivos en la carpeta `Images`.
2. Para cada imagen, el programa crea un registro de cuenta con el `name` ajustado a `CanStoreFullImage false {fileName}` y el archivo `byte[]` establecido como el valor `sample_ImageColumn`.
3. A continuación, el programa utiliza el método `Utility.UpdateCanStoreFullImage` para establecer la definición de columna `sample_ImageColumn` del valor `CanStoreFullImage` en verdadero.
4. Una vez más, el programa recorre los nombres de los archivos y crea cinco registros de cuenta con los mismos archivos de imagen establecidos en el valor `sample_ImageColumn`. Esta vez `name` es `CanStoreFullImage true {fileName}`

En el siguiente código, podemos ver cómo el valor de la propiedad `CanStoreFullImage` cambia qué datos están disponibles.

RECUPERAR LOS REGISTROS DE CUENTA

1. El código recupera los 10 registros de cuenta creados en el paso anterior, incluidos los datos de la imagen.
2. Para cada registro de cuenta, los datos de la imagen se descargan en la carpeta `DownloadedImages` con el nombre `{recordName}_retrieved.png`.

ⓘ Nota

Todos estos registros son imágenes en miniatura.

Descargar las imágenes del registro de cuenta

Este programa utiliza tres métodos diferentes para descargar archivos de imagen.

ⓘ Nota

En cada caso, 5 de las 10 operaciones fallen porque no se cargaron imágenes de tamaño completo mientras `CanStoreFullImage` era falso. Esos registros se crearon mientras `CanStoreFullImage` era verdadero y fueron correctos.

Descargar con Acciones

El código usa el método estático `DownloadImageWithActions`, que encapsula el uso de las acciones `InitializeFileBlocksDownload` y `DownloadBlock` para descargar las imágenes como se describe en [Usar Dataverse mensajes para descargar un archivo](#).

ⓘ Nota

Estas operaciones fallan cuando no hay una imagen de tamaño completo para descargar. El mensaje de error es: `No FileAttachment records found for imagedescriptorId: <guid> for image attribute: sample_imagecolumn of account record with id <guid>`.

Descargar con fragmentos

El código usa el método estático `DownloadImageWithChunks`, que demuestra cómo descargar imágenes como se describe en [Descargar el archivo en fragmentos usando la API web](#).

ⓘ Nota

Estas operaciones no fallan cuando no hay una imagen de tamaño completo para descargar, simplemente devuelven `204 No Content`.

Descargar con Stream

El código usa el método estático `DownloadImageWithStream`, que demuestra cómo descargar imágenes como se describe en [Descargar un archivo en una única solicitud usando la API web](#)

ⓘ Nota

Estas operaciones no fallan cuando no hay una imagen de tamaño completo para descargar, simplemente devuelven `204 No Content`.

Eliminar los datos de imagen

1. El programa utiliza tres métodos diferentes para demostrar la eliminación de valores de imagen, utilizando `PATCH`, `PUT` y `DELETE`.
2. El programa verifica la eliminación de los registros recuperando los registros nuevamente utilizando los mismos criterios que antes. El valor de la columna de la imagen es nulo.

Limpiar

Para dejar el sistema en el estado anterior a la ejecución del ejemplo, el programa hace lo siguiente:

- Eliminar los registros de cuenta.
- Establecer la imagen principal de la tabla de cuenta de nuevo en el valor original.
- Eliminar la columna de imagen.

Consulte también

[Usar datos de columna de imagen](#)

[Columnas imagen](#)

[Usar datos de columna de archivo](#)

[Ejemplo: Operaciones de imagen usando la SDK de Dataverse para .NET](#)

ⓘ Nota

¿Puede indicarnos sus preferencias de idioma de documentación? [Realice una breve encuesta.](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales
[\(declaración de privacidad ↗\)](#).

Ejemplo: operaciones con archivos adjuntos y anotaciones utilizando la API web de Dataverse

Artículo • 19/04/2023

Este ejemplo de .NET 6.0 demuestra cómo realizar operaciones con datos de archivo con los tipos de entidad `activitymimeattachment` y `annotation` utilizando la API web de Dataverse.

Esta muestra utiliza el código auxiliar común del proyecto de ejemplo de la [biblioteca de clases WebAPIService \(C#\)](#).

Requisitos previos

- Microsoft Visual Studio 2022
- Accesa a Dataverse con Administrador del sistema o privilegios de personalizador del sistema.

Cómo ejecutar el ejemplo

1. Clonar o descargar el repositorio de [ejemplos de PowerApps](#).

2. Abra el archivo `PowerApps-Samples\dataverse\webapi\C#\-`

`NETx\AttachmentAndAnnotationOperations\AttachmentAndAnnotationOperations.sln`
usando Visual Studio 2022.

Esta solución contiene dos proyectos que incluyen ejemplos:

- `ActivityMimeAttachmentOperations`: demuestra el uso de archivos adjuntos.
- `AnnotationOperations`: demuestra el uso de Anotaciones.

Nota: El proyecto `WebAPIService` se incluye para que cada uno de los otros proyectos pueda depender del código auxiliar común proporcionado por el servicio. Los ejemplos usan varias clases en la carpeta `WebAPIService/Messages`.

En **Explorador de la solución**, haga clic con el botón derecho en el proyecto que desea ejecutar y seleccione **Establecer como proyecto de inicio**.

3. En cualquier proyecto, edite el archivo `appsettings.json` para establecer los siguientes valores de propiedad:

Property	Instrucciones
Url	La URL para su entorno. Reemplazar el valor de marcador de posición <code>https://yourorg.api.crm.dynamics.com</code> con el valor para su entorno. Consulte Ver recursos para desarrolladores para encontrar esto.
UserPrincipalName	Reemplazar el valor de marcador de posición <code>you@yourorg.onmicrosoft.com</code> con el valor de UPN que usa para acceder al entorno.
Password	Reemplazar el valor de marcador de posición <code>yourPassword</code> con la contraseña que utiliza.

4. Guarde el archivo `appsettings.json`.

Nota: Ambos proyectos hacen referencia al mismo archivo `appsettings.json`, por lo que solo necesita hacer esto una vez para ejecutar cualquiera de los proyectos.

5. Presione **F5** para ejecutar el ejemplo.

Demostraciones

Este ejemplo es una solución con dos proyectos. Consulte los archivos LÉAME respectivos para obtener detalles sobre cada proyecto.

- LÉAME de ejemplo de operaciones de Adjuntos de la API web (ActivityMimeAttachment) ↗
- LÉAME de ejemplo de operaciones de Anotaciones (Notas) de la API web ↗

Clase de utilidad

Ambos proyectos utilizan una clase `Utility` para realizar operaciones comunes. Esta clase contiene tres métodos estáticos:

GetMimeType

Basado en el parámetro `FileInfo`, esta función utiliza `Microsoft.AspNetCore.StaticFiles.FileExtensionContentTypeProvider` para tratar de obtener el tipo MIME del archivo. Si esto no se puede determinar, devuelve `application/octet-stream`

GetMaxUploadFileSize

Utilizando el parámetro **WebAPIService** `Service service`, esta función devuelve el valor entero de `maxuploadfilesize` de la tabla `organization`.

SetMaxUploadFileSize

Utilizando el parámetro **WebAPIService** `Service service`, esta función establece el valor entero de `maxuploadfilesize` de la tabla `organization` al valor el parámetro de entero `maxUploadFileSizeInBytes`.

Consulte también

[Usar datos de archivo con registros de Adjuntos y Notas](#)

[Ejemplo: operaciones de archivo con archivos adjuntos y notas mediante el Dataverse SDK para .NET](#)

Nota

¿Puede indicarnos sus preferencias de idioma de documentación? [Realice una breve encuesta.](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad](#)).

API web, consultar definiciones de esquema y detectar ejemplo de cambios (C#)

Artículo • 23/02/2023

Este ejemplo muestra cómo recuperar y detectar cambios en las definiciones de tabla usando [RetrieveMetadataChanges Action](#).

Puede ver la muestra en [PowerApps-Samples/dataverse/webapi/C#-NETCore/Schema/RetrieveMetadataChanges/ ↗](#)

Consulte estos temas para obtener una explicación de la funcionalidad:

- [Consultar definiciones de esquema](#)
- [Almacenar en caché datos de esquema](#)

Esta muestra utiliza el código auxiliar común de la [biblioteca de clases WebAPIService \(C#\)](#).

Requisitos previos

Los siguientes elementos son necesarios para compilar y ejecutar este ejemplo:

- Microsoft Visual Studio 2022.
- Acceso a Dataverse con privilegios para realizar operaciones de datos.

Cómo ejecutar esta muestra

1. Clonar o descargar el repositorio de [ejemplos de PowerApps ↗](#).
2. Localizar la carpeta [/dataverse/webapi/C#-NETx/RetrieveMetadataChanges/ ↗](#).
3. Abrir el archivo `RetrieveMetadataChanges.sln` usando Visual Studio 2022
4. Editar el archivo `appsettings.json` para establecer los siguientes valores de propiedad:

Property	Instrucciones
----------	---------------

Property	Instrucciones
Url	La URL para su entorno. Reemplazar el valor de marcador de posición <code>https://yourorg.api.crm.dynamics.com</code> con el valor para su entorno. Consulte Ver recursos para desarrolladores para encontrar esto.
UserPrincipalName	Reemplazar el valor de marcador de posición <code>you@yourorg.onmicrosoft.com</code> con el valor de UPN que usa para acceder al entorno.
Password	Reemplazar el valor de marcador de posición <code>yourPassword</code> con la contraseña que utiliza.

5. Guarde el archivo `appsettings.json`

6. Presione F5 para ejecutar el ejemplo.

Código

El código para este ejemplo está aquí: [PowerApps-Samples/dataverse/webapi/C#-NETx/RetrieveMetadataChanges/Program.cs ↗](#)

Demostraciones

Este ejemplo muestra cómo recuperar definiciones de esquema para un conjunto específico de definiciones de columna y guardarlas (en la memoria) para representar una caché.

Luego crea una nueva columna, recupera los datos solo para esa nueva columna, que agrega a la memoria caché.

Luego, elimina la columna, recupera datos sobre los elementos eliminados y los usa para eliminar la definición de la columna eliminada de la caché.

Esta muestra tiene 6 secciones:

Definir consulta

La definición de una consulta usando `EntityQueryExpression` que devolverá todas las columnas de opción de la lista de selección de la tabla de contactos.

Inicializar caché

1. Cree una instancia de [RetrieveMetadataChanges](#) con el conjunto de parámetros `Query` establecido para la consulta.
2. Envíe la solicitud y obtenga un [RetrieveMetadataChangesResponse](#).
3. Almacene en caché el valor de `RetrieveMetadataChangesResponse.EntityMetadata`.
4. Guarde el valor de `RetrieveMetadataChangesResponse.ServerVersionStamp` para usarlo en la siguiente solicitud.
5. Escriba una lista de todas las columnas actuales en la caché.

Agregar columna de opción

Cree una nueva columna de opción creando una nueva instancia de [PicklistAttributeMetadata](#) en la tabla de contactos.

Detectar columna agregada

1. Cree una nueva instancia de [RetrieveMetadataChanges](#) con el parámetro `Query` establecido para la consulta original.
2. Establezca `RetrieveMetadataChangesRequest.ClientVersionStamp` con el valor previamente devuelto desde la primera solicitud.
3. Envíe la solicitud y obtenga un [RetrieveMetadataChangesResponse](#).
4. Compruebe que solo se haya devuelto una nueva definición de columna para representar la columna de elección que se creó.
5. Guarde el valor de `RetrieveMetadataChangesResponse.ServerVersionStamp` para usarlo en la siguiente solicitud.
6. Agregue esos datos de columna de opción a la memoria caché.

Eliminar columna de opción

Elimine la columna de opción creada anteriormente.

Detectar columna eliminada

1. Cree una nueva instancia de [RetrieveMetadataChanges](#) con el parámetro `Query` establecido para la consulta original.
2. Establezca `RetrieveMetadataChangesRequest.ClientVersionStamp` con el valor previamente devuelto desde la segunda solicitud.
3. Establezca `RetrieveMetadataChangesRequest.DeletedMetadataFilters` en `DeletedMetadataFilters.Attribute` porque estamos buscando definiciones de columnas eliminadas.

4. Envíe la solicitud y obtenga un [RetrieveMetadataChangesResponse](#).
5. Encuentre el id. de la columna de opción eliminada en `RetrieveMetadataChangesResponse.DeletedMetadata`, usando `DeletedMetadataFilters.Attribute` como un valor de índice para la colección.
6. Quite la definición de columna de la memoria caché.
7. Escriba una lista de todas las columnas actuales en la caché.

Limpiar

No se requiere limpieza porque se eliminaron todos los datos creados por esta muestra.

Consulte también

[Consultar definiciones de esquema](#)

[Almacenar en caché datos de esquema](#)

[Usar la API web de Dataverse](#)

[Biblioteca de clases WebAPIService \(C#\)](#)

[Ejemplo de operaciones de metadatos de API web \(C#\)](#)

Nota

¿Puede indicarnos sus preferencias de idioma de documentación? [Realice una breve encuesta.](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad](#)).

Muestra: API web, Usar operaciones masivas

Artículo • 07/12/2023

Estas aplicaciones de muestra en GitHub muestran cómo realizar operaciones masivas de creación y actualización mediante las acciones Web API CreateMultiple y UpdateMultiple. Los mensajes para estas acciones están optimizados para proporcionar la forma más eficiente de crear o actualizar registros con Dataverse.

Este ejemplo es una solución Visual Studio .NET 6.0 que contiene dos proyectos que realizan las mismas operaciones de diferentes maneras para que pueda comparar el rendimiento de cada método.

Consulte el archivo README.md en cada muestra para obtener instrucciones detalladas sobre cómo ejecutar la muestra y lo que hace.

Ejemplo de operaciones masivas de la API web

Consulte también

[Mensajes de operación en masa](#)

[Muestra: SDK para .NET, Usar operaciones masivas](#)

[Escribir complementos para CreateMultiple y UpdateMultiple](#)

[Ejemplo: complementos CreateMultiple y UpdateMultiple](#)

ⓘ Nota

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta. [↗](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad ↗](#)).

Ejemplo de operaciones de datos de la API web (JavaScript del lado del cliente)

Artículo • 23/02/2023

Este artículo proporciona una comprensión común sobre los ejemplos de API web que utilizan JavaScript del lado del cliente. Si bien cada ejemplo se centra en un aspecto distinto de la API web de Microsoft Dataverse, todos ellos siguen un proceso y una estructura similares descritos en este tema.

Ejemplos de la API web mediante JavaScript del lado del cliente

Los siguientes ejemplos utilizan los patrones descritos aquí:

Muestra	Grupo de ejemplo	Descripción
Ejemplo de operaciones básicas de la API web (JavaScript del lado del cliente)	Ejemplo de operaciones básicas de la API web	Muestra cómo crear, recuperar, actualizar, eliminar, asociar y anular la asociación de registros de tabla de Dataverse (registros de entidades).
Ejemplo de datos de consulta de la API web (JavaScript del lado del cliente)	Ejemplo de datos de consulta de la API web	Demuestra cómo usar sintaxis y funciones de consulta de OData v4 así como funciones de consulta de Dataverse. Incluye demostración de trabajo con consultas predefinidas y uso de FetchXML para realizar consultas.
Ejemplo de operaciones condicionales de la API web (JavaScript del lado del cliente)	Ejemplo de operaciones condicionales de la API web	Demuestra cómo realizar operaciones condicionales. El comportamiento de estas operaciones depende de los criterios que especifique.
Ejemplo de funciones y acciones de la API web (JavaScript del lado del cliente)	Ejemplo de funciones y acciones de la API web	Demuestra cómo usar funciones y acciones enlazadas y sin enlazar, incluidas acciones personalizadas.

Cómo descargar el código de origen del ejemplo.

El código de origen para cada ejemplo está disponible en [GitHub](#). El vínculo para descargar cada ejemplo se incluye en la página individual de ese ejemplo.

Después de descargar el ejemplo, extraiga el archivo comprimido. Encuentre la solución para cada muestra dentro de la carpeta C# porque el proyecto está un proyecto de aplicación web ASP.NET vacío. Una solución Dataverse también se proporciona en la descarga que puede importar y ejecutar.

ⓘ Nota

Ninguno de los dos, Visual Studio o ASP.NET es necesario para desarrollar JavaScript del lado del cliente para Dataverse. Sin embargo, Visual Studio proporciona una buena experiencia para escribir JavaScript.

Cómo importar la solución de Dataverse que contiene el ejemplo.

Dentro de cada proyecto encontrará un archivo de solución administrada de Dataverse. El nombre de este archivo dependerá del nombre del proyecto de ejemplo, pero el nombre de archivo terminará con `_managed.zip`.

Para importar la solución de Dataverse en el servidor de Dataverse, siga estos pasos:

1. Extraiga el contenido del archivo zip descargado y busque el archivo de la solución de Dataverse, que también será un archivo zip. Por ejemplo, si ha descargado el ejemplo `Basic Operations`, busque el archivo zip de la solución de Dataverse con el nombre `WebAPIBasicOperations\WebAPIBasicOperations_1_0_0_0_managed.zip`.
2. En la UI de Dataverse, vaya a **Configuración > Soluciones**. Esta página enumera todas las soluciones del servidor de Dataverse. Después de acabar de importar esta solución, el nombre de la solución para ese ejemplo aparecerá en esta lista (por ejemplo, `Web API Basics Operations`).
3. Seleccione **Importar** y siga las instrucciones del cuadro de diálogo de importación para completar esta acción.

Cómo ejecutar el ejemplo para ver el script en acción

El programa de ejemplo se ejecuta como recurso web en Dataverse. La solución importada proporciona una página de configuración que le da una opción de mantener o de eliminar datos de ejemplo y un botón para iniciar el programa de ejemplo.

Para ejecutar el ejemplo, lleva a cabo lo siguiente:

1. Desde la página **Todas las soluciones** en Dataverse, seleccione el nombre de la solución (por ejemplo: **Operaciones básicas de la API web Enlace**). Se abrirán las propiedades de la solución en una nueva ventana.
2. Desde el menú de navegación izquierdo, seleccione **Configuración**.
3. Seleccione **Iniciar muestra** para ejecutar el código de muestra.

Elementos comunes que se encuentran en cada ejemplo

En la siguiente lista se resaltan algunos elementos comunes que se encuentran en cada uno de estos ejemplos.

- La función `Sdk.startSample` se llama cuando un usuario selecciona el botón **Iniciar muestra** de la página HTML. La función `Sdk.startSample` inicializa variables globales y pone en marcha la primera operación de la cadena.
- La salida del programa y mensajes de error se envían a la consola de depurador del explorador. Para ver esta salida, abra la ventana de la consola primero antes de ejecutar el ejemplo. Pulse F12 para acceder a las herramientas de desarrollador, incluida la ventana de la consola, en el navegador Microsoft Edge.
- Estos ejemplos usan la implementación nativa [ES6-Promise](#) del explorador para exploradores modernos que la admiten.

No se requieren promesas. Pueden realizarse interacciones similares con funciones de devolución de llamada.

- La función `Sdk.request` controla la solicitud en función de la información pasada en como parámetros. En función de la necesidad de cada ejemplo, los parámetros pasados pueden ser diferentes. Consulte el código de origen del ejemplo para obtener más detalles.

JavaScript

```
/**  
 * @function request
```

```

* @description Generic helper function to handle basic XMLHttpRequest
calls.
* @param {string} action - The request action. String is case-
sensitive.
* @param {string} uri - An absolute or relative URI. Relative URI
starts with a "/".
* @param {object} data - An object representing an entity. Required for
create and update actions.
* @returns {Promise} - A Promise that returns either the request object
or an error object.
*/
Sdk.request = function (action, uri, data) {
    if (!RegExp(action, "g").test("POST PATCH PUT GET DELETE")) {
        // Expected action verbs.
        throw new Error(
            "Sdk.request: action parameter must be one of the following: " +
            "POST, PATCH, PUT, GET, or DELETE."
        );
    }
    if (!typeof uri === "string") {
        throw new Error("Sdk.request: uri parameter must be a string.");
    }
    if (
        RegExp(action, "g").test("POST PATCH PUT") &&
        (data === null || data === undefined)
    ) {
        throw new Error(
            "Sdk.request: data parameter must not be null for operations that
create or modify data."
        );
    }

    // Construct a fully qualified URI if a relative URI is passed in.
    if (uri.charAt(0) === "/") {
        uri = clientUrl + webAPIPath + uri;
    }

    return new Promise(function (resolve, reject) {
        var request = new XMLHttpRequest();
        request.open(action, encodeURI(uri), true);
        request.setRequestHeader("OData-MaxVersion", "4.0");
        request.setRequestHeader("OData-Version", "4.0");
        request.setRequestHeader("Accept", "application/json");
        request.setRequestHeader(
            "Content-Type",
            "application/json; charset=utf-8"
        );
        request.onreadystatechange = function () {
            if (this.readyState === 4) {
                request.onreadystatechange = null;
                switch (this.status) {
                    case 200: // Success with content returned in response body.
                    case 204: // Success with no content returned in response
body.
                }
                resolve(this);
            }
        };
    });
}

```

```
        break;
    default: // All other statuses are unexpected so are treated
like errors.
        var error;
        try {
            error = JSON.parse(request.response).error;
        } catch (e) {
            error = new Error("Unexpected Error");
        }
        reject(error);
        break;
    }
}
};

request.send(JSON.stringify(data));
});

};
```

La función `Sdk.request` devuelve una promesa. Cuando la solicitud envuelta por la promesa está completa, se resuelve o se rechaza la promesa. Si se resuelve, se llamará a la función en el método `then` siguiente. Si se rechaza, se llamará a la función en el método `catch` siguiente. Si la función en el método `then` propiamente dicho devuelve una promesa, la cadena de operaciones en los métodos `then` consecutivos puede continuar. Devolver una promesa nos permite encadenar estas operaciones de ejemplo juntas de forma que sea preferida por muchos programadores a las funciones de devolución de llamada tradicionales. Para obtener más información sobre promesa, consulte [Promesa de JavaScript ↗](#).

Consultar también

[Usar la API web de Dataverse](#)

[Ejemplos de la API web](#)

[Ejemplos de la API web \(C#\)](#)

ⓘ Nota

¿Puede indicarnos sus preferencias de idioma de documentación? [Realice una breve encuesta. ↗](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad ↗](#)).

Ejemplo de operaciones básicas de la API web (JavaScript del lado del cliente)

Artículo • 23/02/2023

Este ejemplo muestra cómo realizar operaciones básicas CRUD (Crear, Recuperar, Actualizar y Eliminar) y de asociación y disociación en filas de tablas (registros de entidades) usando JavaScript de lado de cliente.

ⓘ Nota

En este ejemplo se implementan las operaciones detalladas en el [Ejemplo de operaciones básicas de la API web](#) y se utilizan las construcciones comunes de JavaScript que se describen en [Ejemplos de API web \(JavaScript del lado cliente\)](#)

Requisitos previos

Para ejecutar este ejemplo, se requiere lo siguiente:

- Acceda al entorno de Microsoft Dataverse.
- Una cuenta de usuario con privilegios para importar soluciones y realizar operaciones CRUD, normalmente un rol de seguridad de administrador o personalizador del sistema.

Ejecute este ejemplo

Para ejecutar esta muestra, descargue el paquete de solución de [aquí](#) y extraiga el contenido. Busque la solución `WebAPIBasicOperations_1_0_0_1_managed.zip` e impórtela en su entorno Dataverse y ejecute la muestra. Para obtener instrucciones sobre cómo importar la solución de ejemplo, consulte [ejemplos de API Web \(JavaScript del lado cliente\)](#).

Ejemplo de código

Este ejemplo incluye dos recursos web:

- [WebAPIBasicOperations.html](#)
- [WebAPIBasicOperations.js](#)

WebAPIBasicOperations.html

El recurso web WebAPIBasicOperations.html proporciona el contexto en el que el código JavaScript se ejecutará.

HTML

```
<html>
<head>
    <title>Microsoft CRM Web API Basic Operations Example</title>
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=Edge" />
    <script
        src="..../ClientGlobalContext.js.aspx"
        type="text/javascript"
    ></script>
    <script src="scripts/es6promise.js" type="text/javascript"></script>
    <script
        src="scripts/WebAPIBasicOperations.js"
        type="text/javascript"
    ></script>

    <style type="text/css">
        body {
            font-family: "Segoe UI", Tahoma, Geneva, Verdana, sans-serif;
        }

        #preferences {
            border: inset;
            padding: 10px 10px;
        }

        #output_area {
            border: inset;
            background-color: gainsboro;
            padding: 10px 10px;
        }
    </style>
</head>
<body>
    <h1>Microsoft CRM Web API Basic Operations Example</h1>
    <p>
        This page demonstrates the CRM Web API's basic operations using
        JavaScript.
    </p>

    <h2>Instructions</h2>
    <p>
        Choose your preferences and run the JavaScript code. Use your
        browser's
        developer tools to view the output written to the console (e.g.: in IE
        or Microsoft Edge, press F12 to load the Developer Tools).
    </p>

```

```

</p>
<p>
    Remove sample data (Choose whether you want to delete sample data
created
    during this execution):
<br />
<input name="removesampled" type="radio" value="yes" checked />
Yes
<input name="removesampled" type="radio" value="no" />
No
</p>
<input
    type="button"
    name="start_sample"
    value="Start Sample"
    onclick="Sdk.startSample()"
/>
</body>
</html>

```

WebAPIBasicOperations.js

El recurso web WebAPIBasicOperations.js es la biblioteca JavaScript que define las operaciones que este ejemplo realiza.

JavaScript

```

"use strict";
var Sdk = window.Sdk || {};

/**
 * @function getClientUrl
 * @description Get the client URL.
 * @returns {string} The client URL.
 */
Sdk.getClientUrl = function () {
    var context;
    // GetGlobalContext defined by including reference to
    // ClientGlobalContext.js.aspx in the HTML page.
    if (typeof GetGlobalContext != "undefined") {
        context = GetGlobalContext();
    } else {
        if (typeof Xrm != "undefined") {
            // Xrm.Page.context defined within the Xrm.Page object model for form
            // scripts.
            context = Xrm.Page.context;
        } else {
            throw new Error("Context is not available.");
        }
    }
    return context.getClientUrl();
};

```

```

/**
 * An object instantiated to manage detecting the
 * Web API version in conjunction with the
 * Sdk.retrieveVersion function
 */
Sdk.versionManager = new (function () {
    //Start with base version
    var _webAPIMajorVersion = 8;
    var _webAPIMinorVersion = 0;
    //Use properties to increment version and provide WebAPIPath string used
    by Sdk.request;
    Object.defineProperties(this, {
        WebAPIMajorVersion: {
            get: function () {
                return _webAPIMajorVersion;
            },
            set: function (value) {
                if (typeof value != "number") {
                    throw new Error(
                        "Sdk.versionManager.WebAPIMajorVersion property must be a
number."
                );
            }
            _webAPIMajorVersion = parseInt(value, 10);
        },
        WebAPIMinorVersion: {
            get: function () {
                return _webAPIMinorVersion;
            },
            set: function (value) {
                if (isNaN(value)) {
                    throw new Error(
                        "Sdk.versionManager._webAPIMinorVersion property must be a
number."
                );
            }
            _webAPIMinorVersion = parseInt(value, 10);
        },
        WebAPIPath: {
            get: function () {
                return "/api/data/v" + _webAPIMajorVersion + "." +
_webAPIMinorVersion;
            },
        },
    });
})();

//Setting variables specific to this sample within a container so they won't
be
// overwritten by another scripts code
Sdk.SampleVariables = {
    entitiesToDelete: [], // Entity URIs to be deleted later (if user so

```

```

choosees)

    deleteData: true, // Controls whether sample data are deleted at the end
of sample run
    contact1Uri: null, // e.g.: Peter Cambel
    contactAltUri: null, // e.g.: Peter_Alt Cambel
    account1Uri: null, // e.g.: Contoso, Ltd
    account2Uri: null, // e.g.: Fourth Coffee
    contact2Uri: null, // e.g.: Susie Curtis
    opportunity1Uri: null, // e.g.: Adventure Works
    competitor1Uri: null,
};

/***
 * @function request
 * @description Generic helper function to handle basic XMLHttpRequest
calls.
 * @param {string} action - The request action. String is case-sensitive.
 * @param {string} uri - An absolute or relative URI. Relative URI starts
with a "/".
 * @param {object} data - An object representing an entity. Required for
create and update actions.
 * @param {object} addHeader - An object with header and value properties to
add to the request
 * @returns {Promise} - A Promise that returns either the request object or
an error object.
*/
Sdk.request = function (action, uri, data, addHeader) {
    if (!RegExp(action, "g").test("POST PATCH PUT GET DELETE")) {
        // Expected action verbs.
        throw new Error(
            "Sdk.request: action parameter must be one of the following: " +
            "POST, PATCH, PUT, GET, or DELETE."
        );
    }
    if (!typeof uri === "string") {
        throw new Error("Sdk.request: uri parameter must be a string.");
    }
    if (RegExp(action, "g").test("POST PATCH PUT") && !data) {
        throw new Error(
            "Sdk.request: data parameter must not be null for operations that
create or modify data."
        );
    }
    if (addHeader) {
        if (
            typeof addHeader.header != "string" ||
            typeof addHeader.value != "string"
        ) {
            throw new Error(
                "Sdk.request: addHeader parameter must have header and value
properties that are strings."
            );
        }
    }
}

```

```

// Construct a fully qualified URI if a relative URI is passed in.
if (uri.charAt(0) === "/") {
    //This sample will try to use the latest version of the web API as
detected by the
    // Sdk.retrieveVersion function.
    uri = Sdk.getClientUrl() + Sdk.versionManager.WebAPIPath + uri;
}

return new Promise(function (resolve, reject) {
    var request = new XMLHttpRequest();
    request.open(action, encodeURI(uri), true);
    request.setRequestHeader("OData-MaxVersion", "4.0");
    request.setRequestHeader("OData-Version", "4.0");
    request.setRequestHeader("Accept", "application/json");
    request.setRequestHeader("Content-Type", "application/json; charset=utf-
8");
    if (addHeader) {
        request.setRequestHeader(addHeader.header, addHeader.value);
    }
    request.onreadystatechange = function () {
        if (this.readyState === 4) {
            request.onreadystatechange = null;
            switch (this.status) {
                case 200: // Operation success with content returned in response
body.
                case 201: // Create success.
                case 204: // Operation success with no content returned in
response body.
                    resolve(this);
                    break;
                default: // All other statuses are unexpected so are treated like
errors.
                    var error;
                    try {
                        error = JSON.parse(request.response).error;
                    } catch (e) {
                        error = new Error("Unexpected Error");
                    }
                    reject(error);
                    break;
            }
        }
    };
    request.send(JSON.stringify(data));
});

};

/***
 * @function startSample
 * @description Runs the sample.
 * This sample demonstrates basic CRUD+ operations.
 * Results are sent to the debugger's console window.
 */
Sdk.startSample = function () {
    // Initializing.

```

```

Sdk.SampleVariables.deleteData =
  document.getElementsByName("removesampleddata")[0].checked;
Sdk.SampleVariables.entitiesToDelete = [];// Reset the array.
Sdk.SampleVariables.contact1Uri = "";
Sdk.SampleVariables.account1Uri = "";
Sdk.SampleVariables.account2Uri = "";
Sdk.SampleVariables.contact2Uri = "";
Sdk.SampleVariables.opportunity1Uri = "";
Sdk.SampleVariables.competitor1Uri = "";

</*
 * Behavior of this sample varies by version
 * So starting by retrieving the version;
 */

Sdk.retrieveVersion()
  .then(function () {
    return Sdk.basicCreateAndUpdatesAsync();
  })
  .then(function () {
    return Sdk.createWithAssociationAsync();
  })
  .then(function () {
    return Sdk.createRelatedAsync();
  })
  .then(function () {
    return Sdk.associateExistingAsync();
  })
  .then(function () {
    return Sdk.deleteSampleData();
  })
  .catch(function (err) {
    console.log("ERROR: " + err.message);
  });
};

Sdk.retrieveVersion = function () {
  return new Promise(function (resolve, reject) {
    Sdk.request("GET", "/RetrieveVersion")
      .then(function (request) {
        try {
          var RetrieveVersionResponse = JSON.parse(request.response);
          var fullVersion = RetrieveVersionResponse.Version;
          var versionData = fullVersion.split(".");
          Sdk.versionManager.WebAPIMajorVersion = parseInt(versionData[0],
10);
          Sdk.versionManager.WebAPIMinorVersion = parseInt(versionData[1],
10);
          resolve();
        } catch (err) {
          reject(new Error("Error processing version: " + err.message));
        }
      })
      .catch(function (err) {
        reject(new Error("Error retrieving version: " + err.message));
      });
  });
};

```

```
        });
    });
};

Sdk.basicCreateAndUpdatesAsync = function () {
    return new Promise(function (resolve, reject) {
        // Section 1.
        //
        // Create the contact using POST request.
        // A new entry will be added regardless if a contact with this info
        already exists in the system or not.
        console.log("--Section 1 started--");
        var contact = {};
        contact.firstname = "Peter";
        contact.lastname = "Cambel";

        var entitySetName = "/contacts";

        Sdk.request("POST", entitySetName, contact)
            .then(function (request) {
                // Process response from previous request.
                Sdk.SampleVariables.contact1Uri =
                    request.getResponseHeader("OData-EntityId");
                Sdk.SampleVariables.entitiesToDelete.push(
                    Sdk.SampleVariables.contact1Uri
                ); // To delete later
                console.log(
                    "Contact 'Peter Cambel' created with URI: %s",
                    Sdk.SampleVariables.contact1Uri
                );

                // Setup for next request.
                //
                // Update contact.
                // Add property values to a specific contact using PATCH request.
                var contact = {};
                contact.annualincome = 80000.0;
                contact.jobtitle = "Junior Developer";
                return Sdk.request("PATCH", Sdk.SampleVariables.contact1Uri,
contact);
            })
            .then(function () {
                // Process response from previous request.
                console.log(
                    "Contact 'Peter Cambel' updated with job title and annual income."
                );

                // Setup for next request.
                //
                // Retrieve selected properties of a Contact entity using GET
                request.
                // NOTE: It is performance best practice to select only the
                properties you need.

                // Retrieved contact properties.
```

```

var properties = [
    "fullname",
    "annualincome",
    "jobtitle",
    "description",
].join();

// NOTE: For performance best practices, use $select to limit the
properties you want to return
// See also:
https://msdn.microsoft.com/library/gg334767.aspx#bkmk\_requestProperties
var query = "?$select=" + properties;
return Sdk.request(
    "GET",
    Sdk.SampleVariables.contact1Uri + query,
    null
);
.then(function (request) {
    // Process response from previous request.
    var contact1 = JSON.parse(request.response);
    var successMsg =
        "Contact '%s' retrieved:\n" +
        "\tAnnual income: %s \n" +
        "\tJob title: %s \n" +
        "\tDescription: %s";
    console.log(
        successMsg,
        contact1.fullname, // This property is read-only. Calculated from
firstname and lastname.
        contact1.annualincome,
        contact1.jobtitle,
        contact1.description
    ); // Description will be "null" because it has not been set yet.

    // Setup for next request.
    //
    // Update properties.
    // Set new values for some of the properties and apply the values to
the server via PATCH request.
    // Notice that we are updating the jobtitle and annualincome
properties and adding value to the
    // description property in the same request.
    var contact = {};
    contact.jobtitle = "Senior Developer";
    contact.annualincome = 95000.0;
    contact.description = "Assignment to-be-determined. ";
    return Sdk.request("PATCH", Sdk.SampleVariables.contact1Uri,
contact);
})
.then(function () {
    // Process response from previous request.
    console.log(
        "Contact 'Peter Cambel' updated:\n" +
        "\tJob title: Senior Developer, \n" +

```

```

        "\tAnnual income: 95000, \n" +
        "\tDescription: Assignment to-be-determined."
    );

    // Setup for next request.
    //
    // Set value for a single property using PUT request.
    // In this case, we are setting the telephone1 property to "555-
0105".
    var value = { value: "555-0105" };
    return Sdk.request(
        "PUT",
        Sdk.SampleVariables.contact1Uri + "/telephone1",
        value
    );
}
.then(function () {
    // Process response from previous request.
    console.log("Contact 'Peter Cambel' phone number updated.");

    // Setup for next request.
    //
    // Retrieve single value property.
    // Get a value of a single property using GET request.
    // In this case, telephone1 is retrieved. We should get back "555-
0105".
    return Sdk.request(
        "GET",
        Sdk.SampleVariables.contact1Uri + "/telephone1",
        null
    );
})
.then(function (request) {
    // Process response from previous request.
    var phoneNumber = JSON.parse(request.response);
    console.log("Contact's phone number is: %s", phoneNumber.value);
})
.then(function () {
    // Setup for next request.
    //The following operations require version 8.2 or higher
    if (
        Sdk.versionManager.WebAPIMajorVersion > 8 ||
        (Sdk.versionManager.WebAPIMajorVersion == 8 &&
         Sdk.versionManager.WebAPIMinorVersion >= 2)
    ) {
        // Starting with December 2016 update (v8.2), a contact instance
can be
        // created and its properties returned in one operation by using a
        //'Prefer: return=representation' header.
        var contactAlt = {};
        contactAlt.firstname = "Peter_Alt";
        contactAlt.lastname = "Cambel";
        contactAlt.jobtitle = "Junior Developer";
        contactAlt.annualincome = 80000;
        contactAlt.telephone1 = "555-0110";
    }
})

```

```

var properties = ["fullname", "annualincome", "jobtitle"].join();
var query = "?$select=" + properties;
// Create contact and return its state (in the body).
var retRepHeader = {
  header: "Prefer",
  value: "return=representation",
};
Sdk.request("POST", entitySetName + query, contactAlt,
retRepHeader)
.then(function (request) {
  var contactA = JSON.parse(request.response);
  //Because 'OData-EntityId' header not returned in a 201
response, you must instead
  // construct the URI.
  Sdk.SampleVariables.contactAltUri =
    Sdk.getClientUrl() +
    Sdk.versionManager.WebAPIPath +
    "/contacts(" +
    contactA.contactid +
    ")";
  Sdk.SampleVariables.entitiesToDelete.push(
    Sdk.SampleVariables.contactAltUri
  );
  var successMsg =
    "Contact '%s' created:\n" +
    "\tAnnual income: %s \n" +
    "\tJob title: %s \n";
  console.log(
    successMsg,
    contactA.fullname,
    contactA.annualincome,
    contactA.jobtitle
  );
  console.log("Contact URI: %s",
  Sdk.SampleVariables.contactAltUri);
})
.then(function () {
  // Setup for next request.
  //Similarly, the December 2016 update (v8.2) also enables
returning selected properties
  //after an update operation (PATCH), with the 'Prefer:
return=representation' header.
  var contactAlt = {};
  contactAlt.jobtitle = "Senior Developer";
  contactAlt.annualincome = 95000;
  contactAlt.description = "MS Azure and Dataverse Specialist";
  var properties = [
    "fullname",
    "annualincome",
    "jobtitle",
    "description",
  ].join();
  var query = "?$select=" + properties;
  // Update contact and return its state (in the body).
  var retRepHeader = {

```

```

        header: "Prefer",
        value: "return=representation",
    );
    return Sdk.request(
        "PATCH",
        Sdk.SampleVariables.contactAltUri + query,
        contactAlt,
        retRepHeader
    );
})
.then(function (request) {
    // Process response from previous request.
    var contactA = JSON.parse(request.response);
    var successMsg =
        "Contact '%s' updated:\n" +
        "\tAnnual income: %s \n" +
        "\tJob title: %s \n";
    console.log(
        successMsg,
        contactA.fullname,
        contactA.annualincome,
        contactA.jobtitle
    );
    //End this series of operations:
    resolve();
})
.catch(function (err) {
    reject(err);
});
} else {
    resolve();
}
})
.catch(function (err) {
    reject(err);
});
});
});
};

Sdk.createWithAssociationAsync = function () {
    return new Promise(function (resolve, reject) {
        // Section 2.
        //
        // Create a new account entity and associate it with an existing contact
        using POST request.
        console.log("\n--Section 2 started--");
        var account = {};
        account.name = "Contoso, Ltd.";
        account.telephone1 = "555-5555";
        account["primarycontactid@odata.bind"] =
        Sdk.SampleVariables.contact1Uri; //relative URI ok. E.g.: "/contacts(###)".

        var entitySetName = "/accounts";

        Sdk.request("POST", entitySetName, account)
    })
}
});
```

```

        .then(function (request) {
            // Process response from previous request.
            Sdk.SampleVariables.account1Uri =
                request.getResponseHeader("OData-EntityId");
            Sdk.SampleVariables.entitiesToDelete.push(
                Sdk.SampleVariables.account1Uri
            );
            console.log("Account 'Contoso, Ltd.' created.");

            // Setup for next request.
            //
            // Retrieve account's primary contact with selected properties using
            GET request and 'expand' query.
            var contactProperties = ["fullname", "jobtitle",
            "annualincome"].join();
            var query =
                "?$select=name,telephone1&$expand=primarycontactid($select=" +
                contactProperties +
                ")";
            return Sdk.request(
                "GET",
                Sdk.SampleVariables.account1Uri + query,
                null
            );
        })
        .then(function (request) {
            // Process response from previous request.
            var account1 = JSON.parse(request.response);
            var successMsg =
                "Account '%s' has primary contact '%s': \n" +
                "\tJob title: %s \n" +
                "\tAnnual income: %s ";
            console.log(
                successMsg,
                account1.name,
                account1.primarycontactid.fullname,
                account1.primarycontactid.jobtitle,
                account1.primarycontactid.annualincome
            );
            //End this series of operations:
            resolve();
        })
        .catch(function (err) {
            reject(err);
        });
    });
};

Sdk.createRelatedAsync = function () {
    return new Promise(function (resolve, reject) {
        // Section 3.
        //
        // Create related entities (deep insert).
        // Create the following entities in one operation using deep insert
        technique:
    });
};

```

```

// account
// |--- contact
//     |--- tasks
// Then retrieve properties of these entities
//
// Constructing the entity relationship.
console.log("\n--Section 3 started--");
var account = {};
account.name = "Fourth Coffee";
account.primarycontactid = {
    firstname: "Susie",
    lastname: "Curtis",
    jobtitle: "Coffee Master",
    annualincome: 48000.0,
    Contact_Tasks: [
        {
            subject: "Sign invoice",
            description: "Invoice #12321",
            scheduledend: new Date("April 19th, 2016"),
        },
        {
            subject: "Setup new display",
            description: "Theme is - Spring is in the air",
            scheduledstart: new Date("4/20/2016"),
        },
        {
            subject: "Conduct training",
            description: "Train team on making our new blended coffee",
            scheduledstart: new Date("6/1/2016"),
        },
    ],
};

var entitySetName = "/accounts";
Sdk.request("POST", entitySetName, account)
.then(function (request) {
    // Process response from previous request.
    Sdk.SampleVariables.account2Uri =
        request.getResponseHeader("OData-EntityId");
    Sdk.SampleVariables.entitiesToDelete.push(
        Sdk.SampleVariables.account2Uri
    );
    console.log("Account 'Fourth Coffee' created.");

    // Setup for next request.
    //
    // Retrieve account entity info using GET request and 'expand'
query.
    var contactProperties = ["fullname", "jobtitle",
    "annualincome"].join();

    // Expand on primarycontactid to select some of contact's
properties.
    // NOTE: With $expand, the CRM server will return values for the
selected properties.

```

```

        // The CRM Web API only supports expansions one level deep.
        // See also:
https://msdn.microsoft.com/library/mt607871.aspx#bkmk\_expandRelated
    var query =
        "?$select=name&$expand=primarycontactid($select=" +
        contactProperties +
        ")";
    return Sdk.request(
        "GET",
        Sdk.SampleVariables.account2Uri + query,
        null
    );
})
.then(function (request) {
    // Process response from previous request.
    var account2 = JSON.parse(request.response);
    var successMsg =
        "Account '%s' has primary contact '%s':\n" +
        "\tJob title: %s\n" +
        "\tAnnual income: %s";
    console.log(
        successMsg,
        account2.name,
        account2.primarycontactid.fullname,
        account2.primarycontactid.jobtitle,
        account2.primarycontactid.annualincome
    );

    // Setup for next request.
    //
    // Retrieve contact entity and expanding on its tasks using GET
request.
    Sdk.SampleVariables.contact2Uri =
        Sdk.getClientUrl() +
        Sdk.versionManager.WebAPIPath +
        "/contacts(" +
        account2.primarycontactid.contactid +
        ")"; //Full URI.
    Sdk.SampleVariables.entitiesToDelete.push(
        Sdk.SampleVariables.contact2Uri
    ); // For Susie Curtis
    var contactProperties = ["fullname", "jobtitle"].join();
    var contactTaskProperties = [
        "subject",
        "description",
        "scheduledstart",
        "scheduledend",
    ].join();

    // Expand on contact_tasks to select some of its properties for each
task.
    var query =
        "?$select=" +
        contactProperties +
        "&$expand=Contact_Tasks($select=" +

```

```

        contactTaskProperties +
    ")");
    return Sdk.request(
        "GET",
        Sdk.SampleVariables.contact2Uri + query,
        null
    );
})
.then(function (request) {
    // Process response from previous request.
    var contact2 = JSON.parse(request.response);
    console.log(
        "Contact '%s' has the following assigned tasks:",
        contact2.fullname
    );

    // construct the output string.
    var successMsg =
        "Subject: %s \n" +
        "\tDescription: %s \n" +
        "\tStart: %s \n" +
        "\tEnd: %s \n";

    for (var i = 0; i < contact2.Contact_Tasks.length; i++) {
        console.log(
            successMsg,
            contact2.Contact_Tasks[i].subject,
            contact2.Contact_Tasks[i].description,
            contact2.Contact_Tasks[i].scheduledstart,
            contact2.Contact_Tasks[i].scheduledend
        );
    }
}

//End this series of operations:
.resolve();
})
.catch(function (err) {
    reject(err);
});
});

Sdk.associateExistingAsync = function () {
    return new Promise(function (resolve, reject) {
        // Section 4
        //
        // Entity associations:
        // Associate to existing entities via the different relationship types:
        // 1) 1:N relationship - Associate an existing contact to an existing
        account
        //      (e.g.: contact - Peter Cambel to account - Fourth Coffee).
        // 2) N:N relationship - Associate an competitor to opportunity.

        console.log("\n--Section 4 started--");
        var contact = {};
    });
}

```

```

contact["@odata.id"] = Sdk.SampleVariables.contact1Uri;

Sdk.request(
  "POST",
  Sdk.SampleVariables.account2Uri + "/contact_customer_accounts/$ref",
  contact
)
.then(function () {
  // Process response from previous request.
  console.log(
    "Contact 'Peter Cambel' associated to account 'Fourth Coffee'."
  );

  // Setup for next request.
  //
  // Verify that the reference was made as expected.
  var contactProperties = ["fullname", "jobtitle"].join();

  // This returns a collection of all associated contacts...in a
"value" array.
  var query = "/contact_customer_accounts?$select=" +
contactProperties;
  return Sdk.request(
    "GET",
    Sdk.SampleVariables.account2Uri + query,
    null
  );
})
.then(function (request) {
  // Process response from previous request.
  var relatedContacts = JSON.parse(request.response).value;
//collection is in the "value" array.
  var successMsg = "\tName: %s, " + "Job title: %s ";

  console.log("Contact list for account 'Fourth Coffee': ");

  for (var i = 0; i < relatedContacts.length; i++) {
    console.log(
      successMsg,
      relatedContacts[i].fullname,
      relatedContacts[i].jobtitle
    );
  }

  // Setup for next request.
  //
  // Disassociate a contact from an account.
  return Sdk.request(
    "DELETE",
    Sdk.SampleVariables.account2Uri +
      "/contact_customer_accounts/$ref?id=" +
      Sdk.SampleVariables.contact1Uri,
    null
  );
})

```

```

.then(function () {
    // Process response from previous request.
    console.log(
        "Contact 'Peter Cambel' disassociated from account 'Fourth
Coffee'."
    );
}

// Setup for next request.
//
// N:N relationship:
// Associate a competitor to an opportunity.
var competitor = {};
competitor.name = "Adventure Works";
competitor.strengths =
    "Strong promoter of private tours for multi-day outdoor
adventures./";

var entitySetName = "/competitors";
return Sdk.request("POST", entitySetName, competitor);
})
.then(function (request) {
    // Process response from previous request.
Sdk.SampleVariables.competitor1Uri =
    request.getResponseHeader("OData-EntityId");
Sdk.SampleVariables.entitiesToDelete.push(
    Sdk.SampleVariables.competitor1Uri
);
console.log("Competitor 'Adventure Works' created.");

// Setup for next request.
//
// Create a new opportunity...
var opportunity = {};
opportunity.name = "River rafting adventure";
opportunity.description =
    "Sales team on a river-rafting offsite and team building.";
var entitySetName = "/opportunities";
return Sdk.request("POST", entitySetName, opportunity);
})
.then(function (request) {
    // Process response from previous request.
Sdk.SampleVariables.opportunity1Uri =
    request.getResponseHeader("OData-EntityId");
Sdk.SampleVariables.entitiesToDelete.push(
    Sdk.SampleVariables.opportunity1Uri
);
console.log("Opportunity 'River rafting adventure' created.");

// Setup for next request.
//
// Associate competitor to opportunity.
var competitor = {};
competitor["@odata.id"] = Sdk.SampleVariables.competitor1Uri;
return Sdk.request(
    "POST",

```

```

        Sdk.SampleVariables.opportunity1Uri +
        "/opportunitycompetitors_association/$ref",
        competitor
    );
})
.then(function () {
    // Process response from previous request.
    console.log(
        "Opportunity 'River rafting adventure' associated with competitor
'Adventure Works'."
    );

    // Setup for next request.
    //
    // Retrieve competitor entity and expanding on its
opportunitycompetitors_association
    // for all opportunities, using GET request.
    var opportunityProperties = ["name", "description"].join();
    var competitorProperties = ["name"].join();
    var query =
        "?$select=" +
        competitorProperties +
        "&$expand=opportunitycompetitors_association($select=" +
        opportunityProperties +
        ")";
    return Sdk.request(
        "GET",
        Sdk.SampleVariables.competitor1Uri + query,
        null
    );
})
.then(function (request) {
    // Process response from previous request.
    var competitor1 = JSON.parse(request.response);
    console.log(
        "Competitor '%s' has the following opportunities:",
        competitor1.name
    );
    var successMsg = "\tName: %s, \n" + "\tDescription: %s";
    for (
        var i = 0;
        i < competitor1.opportunitycompetitors_association.length;
        i++
    ) {
        console.log(
            successMsg,
            competitor1.opportunitycompetitors_association[i].name,
            competitor1.opportunitycompetitors_association[i].description
        );
    }
}

// Setup for next request.
//
// Disassociate competitor from opportunity.
return Sdk.request(

```

```

        "DELETE",
        Sdk.SampleVariables.opportunity1Uri +
            "/opportunitycompetitors_association/$ref?$id=" +
            Sdk.SampleVariables.competitor1Uri,
            null
        );
    })
    .then(function () {
        // Process response from previous request.
        console.log(
            "Opportunity 'River rafting adventure' disassociated with
competitor 'Adventure Works'"
        );
        //End this series of operations:
        resolve();
    })
    .catch(function (err) {
        reject(err);
    });
});
};

Sdk.deleteSampleData = function () {
    return new Promise(function (resolve, reject) {
        // House cleaning - deleting sample data
        // NOTE: If instances have a parent-child relationship, then deleting
        the parent will,
        // by default, automatically cascade delete child instances. In this
        program,
        // tasks related using the Contact_Tasks relationship have contact as
        their parent.
        // Other relationships may behave differently.
        // See also:
https://msdn.microsoft.com/library/gg309412.aspx#BKMK\_CascadingBehavior
        console.log("\n--Section 5 started--");
        if (Sdk.SampleVariables.deleteData) {
            for (var i = 0; i < Sdk.SampleVariables.entitiesToDelete.length; i++) {
                console.log(
                    "Deleting entity: " + Sdk.SampleVariables.entitiesToDelete[i]
                );
                Sdk.request(
                    "DELETE",
                    Sdk.SampleVariables.entitiesToDelete[i],
                    null
                ).catch(function (err) {
                    reject(
                        new Error("ERROR: Delete failed --Reason: \n\t" + err.message)
                    );
                });
            }
            resolve();
        } else {
            console.log("Sample data not deleted.");
            resolve();
        }
    })
}

```

```
    }  
});  
};
```

Consultar también

[Usar la API web de Dataverse](#)

[Crear una fila de tabla usando la API web](#)

[Recuperar una fila de tabla usando la API web](#)

[Actualizar y eliminar filas de tablas usando la API web](#)

[Ejemplos de la API web](#)

[Ejemplo de operaciones básicas de la API web](#)

[Ejemplo de operaciones básicas de la API web \(C#\)](#)

[Ejemplos de la API web \(JavaScript del lado del cliente\)](#)

[Ejemplo de datos de consulta de la API web \(JavaScript del lado del cliente\)](#)

[Ejemplo de operaciones condicionales de la API web \(JavaScript del lado del cliente\)](#)

[Ejemplo de funciones y acciones de la API web \(JavaScript del lado del cliente\)](#)

Nota

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta. [↗](#) (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales ([declaración de privacidad ↗](#)).

Ejemplo de operaciones condicionales de la API web (JavaScript del lado del cliente)

Artículo • 23/02/2023

Este ejemplo muestra cómo realizar operaciones condicionales mediante la API web de Microsoft Dataverse usando JavaScript del lado del cliente.

ⓘ Nota

Este ejemplo implementa las operaciones detalladas en el [ejemplo de operaciones condicionales de la API web](#) y utiliza las construcciones comunes del lado cliente de JavaScript que se describen en [ejemplos de API web \(JavaScript del lado cliente\)](#).

Requisitos previos

Para ejecutar este ejemplo, se requiere lo siguiente:

- Acceder a un entorno de Dataverse.
- Una cuenta de usuario con privilegios para importar soluciones y realizar operaciones CRUD, normalmente un rol de seguridad de administrador o personalizador del sistema.

Ejecute este ejemplo

Para ejecutar esta muestra, descargue el paquete de solución de [aquí](#), extraiga el contenido y ubique la solución administrada

`WebAPIConditionalOperations_1_0_0_0_managed.zip`. Importe la solución administrada en su entorno de Dataverse y vea la página de configuración de la solución para ejecutar el ejemplo. Para obtener instrucciones sobre cómo importar la solución de ejemplo, consulte [ejemplos de API Web \(JavaScript del lado cliente\)](#).

Ejemplo de código

Este ejemplo incluye dos recursos web:

- [WebAPIConditionalOperations.html](#)
- [WebAPIConditionalOperations.js](#)

WebAPIConditionalOperations.html

El recurso web WebAPIConditionalOperations.html proporciona el contexto en el que el código JavaScript se ejecutará.

HTML

```
<!DOCTYPE html>
<html>
<head>
    <title>Microsoft CRM Web API Conditional Operations Example</title>
    <meta charset="utf-8" />
    <script
        src="../ClientGlobalContext.js.aspx"
        type="text/javascript"
    ></script>
    <script src="scripts/es6promise.js"></script>
    <script src="scripts/WebAPIConditionalOperations.js"></script>

    <style type="text/css">
        body {
            font-family: "Segoe UI", Tahoma, Geneva, Verdana, sans-serif;
        }
    </style>
</head>
<body>
    <h1>Microsoft CRM Web API Conditional Operations Example</h1>
    <p>
        This page demonstrates the CRM Web API's Conditional Operations using
        JavaScript.
    </p>

    <h2>Instructions</h2>
    <p>
        Choose your preferences and run the JavaScript code. Use your
        browser's
        developer tools to view the output written to the console (e.g.: in
        IE11
        or Microsoft Edge, press F12 to load the Developer Tools).
    </p>
    <form id="preferences">
        <p>This sample deletes the single record it creates.</p>
        <input
            type="button"
            name="start_samples"
            value="Start Sample"
            onclick="Sdk.startSample()"
        />
    </form>
</body>
</html>
```

```
</form>
</body>
</html>
```

WebAPIConditionalOperations.js

El recurso web WebAPIConditionalOperations.js es la biblioteca JavaScript que define las operaciones que este ejemplo realiza.

JavaScript

```
"use strict";
var Sdk = window.Sdk || {};
/**
 * @function getClientUrl
 * @description Get the client URL.
 * @return {string} The client URL.
 */
Sdk.getClientUrl = function () {
    var context;
    // GetGlobalContext defined by including reference to
    // ClientGlobalContext.js.aspx in the HTML page.
    if (typeof GetGlobalContext != "undefined") {
        context = GetGlobalContext();
    } else {
        if (typeof Xrm != "undefined") {
            // Xrm.Page.context defined within the Xrm.Page object model for form
            // scripts.
            context = Xrm.Page.context;
        } else {
            throw new Error("Context is not available.");
        }
    }
    return context.getClientUrl();
};

// Global variables.
var clientUrl = Sdk.getClientUrl(); // e.g.: https://org.crm.dynamics.com
var webAPIPath = "/api/data/v8.1"; // Path to the web API.
var account1Uri; // e.g.: Contoso Ltd (sample)
var initialAcctETagVal; // The initial ETag value of the account created
var updatedAcctETagVal; // The ETag value of the account after it is updated

// Entity properties to select in a request.
var contactProperties = ["fullname", "jobtitle", "annualincome"];
var accountProperties = ["name"];
var taskProperties = ["subject", "description"];

/**
 * @function request
 * @description Generic helper function to handle basic XMLHttpRequest
 * calls.
```

```

    * @param {string} action - The request action. String is case-sensitive.
    * @param {string} uri - An absolute or relative URI. Relative URI starts
      with a "/".
    * @param {object} data - An object representing an entity. Required for
      create and update actions.
    * @param {object} addHeader - An object with header and value properties to
      add to the request
    * @returns {Promise} - A Promise that returns either the request object or
      an error object.
  */
Sdk.request = function (action, uri, data, addHeader) {
  if (!RegExp(action, "g").test("POST PATCH PUT GET DELETE")) {
    // Expected action verbs.
    throw new Error(
      "Sdk.request: action parameter must be one of the following: " +
      "POST, PATCH, PUT, GET, or DELETE."
    );
  }
  if (!typeof uri === "string") {
    throw new Error("Sdk.request: uri parameter must be a string.");
  }
  if (RegExp(action, "g").test("POST PATCH PUT") && !data) {
    throw new Error(
      "Sdk.request: data parameter must not be null for operations that
      create or modify data."
    );
  }
  if (addHeader) {
    if (
      typeof addHeader.header != "string" ||
      typeof addHeader.value != "string"
    ) {
      throw new Error(
        "Sdk.request: addHeader parameter must have header and value
        properties that are strings."
      );
    }
  }
}

// Construct a fully qualified URI if a relative URI is passed in.
if (uri.charAt(0) === "/") {
  uri = clientUrl + webAPIPath + uri;
}

return new Promise(function (resolve, reject) {
  var request = new XMLHttpRequest();
  request.open(action, encodeURI(uri), true);
  request.setRequestHeader("OData-MaxVersion", "4.0");
  request.setRequestHeader("OData-Version", "4.0");
  request.setRequestHeader("Accept", "application/json");
  request.setRequestHeader("Content-Type", "application/json; charset=utf-
8");
  if (addHeader) {
    request.setRequestHeader(addHeader.header, addHeader.value);
  }
})

```

```
request.onreadystatechange = function () {
    if (this.readyState === 4) {
        request.onreadystatechange = null;
        switch (this.status) {
            case 200: // Success with content returned in response body.
            case 204: // Success with no content returned in response body.
            case 304: // Success with Not Modified.
                resolve(this);
                break;
            default: // All other statuses are error cases.
                var error;
                try {
                    error = JSON.parse(request.response).error;
                } catch (e) {
                    error = new Error("Unexpected Error");
                }
                reject(error);
                break;
        }
    }
};

request.send(JSON.stringify(data));
});

};

/***
 * @function startSample
 * @description Runs the sample.
 * This sample demonstrates conditional operations using CRM Web API.
 * Results are sent to the debugger's console window.
 */
Sdk.startSample = function () {
    // Initializing...
    console.log("-- Sample started --");

    // Create the CRM account instance.
    var account = {
        name: "Contoso, Ltd",
        telephone1: "555-0000", // Phone number value will increment with each
        update attempt.
        revenue: 5000000,
        description: "Parent company of Contoso Pharmaceuticals, etc.",
    };

    var uri = "/accounts"; // A relative URI to the account entity.
    Sdk.request("POST", uri, account)
        .then(function (request) {
            console.log("Account entity created.");
            // Assign the Uri to the created account to a global variable.
            account1Uri = request.getResponseHeader("OData-EntityId");

            // Retrieve the created account entity.
            return Sdk.request(
                "GET",
                account1Uri + "?$select=name,revenue,telephone1,description"
            );
        })
        .catch(function (error) {
            console.error("Error creating account: " + error.message);
        });
};
```

```

    );
  })
  .then(function (request) {
    // Show the current entity properties.
    var account = JSON.parse(request.response);
    console.log(JSON.stringify(account, null, 2));

    initialAcctETagVal = account["@odata.etag"]; // Save the current ETag
    value.

    // Conditional Get START.
    // Attempt to retrieve using conditional GET with current ETag value.
    // Expecting nothing in the response because entity was not modified.
    console.log("-- Conditional GET section started --");
    var ifNoneMatchETag = {
      header: "If-None-Match",
      value: initialAcctETagVal,
    };
    return Sdk.request(
      "GET",
      account1Uri + "?$select=name,revenue,telephone1,description",
      null,
      ifNoneMatchETag
    );
  })
  .then(function (request) {
    console.log("Instance retrieved using ETag: %s", initialAcctETagVal);
    if (request.status == 304) {
      //Expected:
      console.log("\tEntity was not modified so nothing was returned.");
      console.log(request.response); //Nothing
    } else {
      //Not Expected:
      console.log(JSON.stringify(JSON.parse(request.response), null, 2));
    }

    // Modify the account instance by updating telephone1.
    // This request operation will also update the ETag value.
    return Sdk.request("PUT", account1Uri + "/telephone1", {
      value: "555-0001",
    });
  })
  .then(function (request) {
    console.log("Account telephone number updated.");

    // Re-attempt conditional GET with original ETag value.
    var ifNoneMatchETag = {
      header: "If-None-Match",
      value: initialAcctETagVal,
    };
    return Sdk.request(
      "GET",
      account1Uri + "?$select=name,revenue,telephone1,description",
      null,
      ifNoneMatchETag
    )
  })
})
```

```

    );
  })
  .then(function (request) {
    if (request.status == 200) {
      // Expected.
      console.log("Instance retrieved using ETag: %s",
initialAcctETagVal);
      var account = JSON.parse(request.response);
      updatedAcctETagVal = account["@odata.etag"]; //Capture updated ETag.
      console.log(JSON.stringify(account, null, 2));
    } else {
      // Not Expected.
      console.log("Unexpected status: %s", request.status);
    }
    // Conditional Get END.

    // Optimistic concurrency on delete and update START.
    console.log("-- Optimistic concurrency section started --");
    // Attempt to delete original account (only if matches original ETag
value).
    var ifMatchETag = { header: "If-Match", value: initialAcctETagVal };
    return Sdk.request("DELETE", account1Uri, null, ifMatchETag);
  })
  .then(
    function (request) {
      // Success not expected.
      console.log("Unexpected status: %s", request.status);
    },
    // Catch error.
    function (error) {
      // DELETE: Precondition failed error expected.
      console.log("Expected Error: %s", error.message);
      console.log(
        "\tAccount not deleted using ETag '%s', status code: '%s'.",
        initialAcctETagVal,
        412
      );
    }

    // Attempt to update account (if matches original ETag value).
    var accountUpdate = {
      telephone1: "555-0002",
      revenue: 6000000,
    };
    var ifMatchETag = { header: "If-Match", value: initialAcctETagVal };
    return Sdk.request("PATCH", account1Uri, accountUpdate,
ifMatchETag);
  }
)
.then(
  function (request) {
    // Success not expected.
    console.log("Unexpected status: %s", request.status);
  },
  // Catch error.
  function (error) {

```

```

        // UPDATE: Precondition failed error expected.
        console.log("Expected Error: %s", error.message);
        console.log(
            "\tAccount not updated using ETag '%s', status code: '%s'.",
            initialAcctETagVal,
            412
        );

        // Re-attempt update if matches current ETag value.
        var accountUpdate = {
            telephone1: "555-0003",
            revenue: 6000000,
        };
        var ifMatchETag = { header: "If-Match", value: updatedAcctETagVal };
        return Sdk.request("PATCH", account1Uri, accountUpdate,
ifMatchETag);
    }
)
.then(function (request) {
    if (request.status == 204) {
        //No Content
        // Expected.
        console.log(
            "Account successfully updated using ETag '%s', status code:
'%s'.",
            updatedAcctETagVal,
            request.status
        );
    } else {
        // Not Expected.
        console.log("Unexpected status: %s", request.status);
    }
    // Retrieve and output current account state.
    return Sdk.request(
        "GET",
        account1Uri + "?$select=name,revenue,telephone1,description"
    );
})
.then(function (request) {
    var account = JSON.parse(request.response);
    updatedAcctETagVal = account["@odata.etag"]; // Capture updated ETag.
    console.log(JSON.stringify(account, null, 2));
    // Optimistic concurrency on delete and update END.

    // Controlling upsert operations START.
    console.log("-- Controlling upsert operations section started --");

    // Attempt to insert (without update) some properties for this
account.
    var accountUpsert = {
        telephone1: "555-0004",
        revenue: 7500000,
    };
    var ifNoneMatchResource = { header: "If-None-Match", value: "*" };
    return Sdk.request(

```

```

        "PATCH",
        account1Uri,
        accountUpsert,
        ifNoneMatchResource
    );
})
.then(
    function (request) {
        // Success not expected.
        console.log("Unexpected status: %s", request.status);
    },
    // Catch error.
    function (error) {
        // Precondition failed error expected.
        console.log("Expected Error: %s", error.message);
        console.log(
            "\tAccount not updated using ETag '%s', status code: '%s'.",
            initialAcctETagVal,
            412
        );
    }

    // Attempt to perform same update without creation.
    var accountUpsert = {
        telephone1: "555-0005",
        revenue: 7500000,
    };
    // Perform operation only if matching resource exists.
    var ifMatchResource = { header: "If-Match", value: "*" };
    return Sdk.request(
        "PATCH",
        account1Uri,
        accountUpsert,
        ifMatchResource
    );
}
)
.then(function (request) {
    if (request.status == 204) {
        // No Content.
        // Expected.
        console.log("Account updated using If-Match '*'");
    } else {
        // Not Expected.
        console.log("Unexpected status: %s", request.status);
    }

    // Retrieve and output current account state.
    return Sdk.request(
        "GET",
        account1Uri + "?$select=name,revenue,telephone1,description"
    );
})
.then(function (request) {
    var account = JSON.parse(request.response);
    updatedAcctETagVal = account["@odata.etag"]; // Capture updated ETag.
})

```

```

        console.log(JSON.stringify(account, null, 2));

        // Controlling upsert operations END.

        // Prevent update of deleted entity START.
        // Delete the account.
        return Sdk.request("DELETE", account1Uri);
    })
    .then(function (request) {
        if (request.status == 204) {
            console.log("Account was deleted");

            // Attempt to update it.
            var accountUpsert = {
                telephone1: "555-0005",
                revenue: 7500000,
            };
            // Perform operation only if matching resource exists.
            var ifMatchResource = { header: "If-Match", value: "*" };
            return Sdk.request(
                "PATCH",
                account1Uri,
                accountUpsert,
                ifMatchResource
            );
        }
    })
    .then(
        function (request) {
            // Success not expected.
            // Without the If-Match header while using PATCH a new entity would
            have been created with the
            // same ID as the deleted entity.
            console.log("Unexpected status: %s", request.status);
        },
        // Catch error.
        function (error) {
            // Not found error expected.
            console.log("Expected Error: %s", error.message);
            console.log("\tAccount not updated because it doesn't exist.");
        }
    )
    .catch(function (error) {
        console.log(error.message);
    });
});

```

Vea también

[Usar la API web de Dataverse](#)

[Realizar operaciones condicionales mediante la API web](#)

[Ejemplos de la API web](#)

[Ejemplo de operaciones condicionales de la API web](#)

[Ejemplo de operaciones condicionales de la API web \(C#\)](#)

[Ejemplos de la API web \(JavaScript del lado del cliente\)](#)

[Ejemplo de operaciones básicas de la API web \(JavaScript del lado del cliente\)](#)

[Ejemplo de datos de consulta de la API web \(JavaScript del lado del cliente\)](#)

[Ejemplo de funciones y acciones de la API web \(JavaScript del lado del cliente\)](#)

 **Nota**

¿Puede indicarnos sus preferencias de idioma de documentación? **Realice una breve encuesta.**  (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales (**declaración de privacidad** ).

Ejemplo de funciones y acciones de la API web (JavaScript del lado del cliente)

Artículo • 23/02/2023

Este ejemplo demuestra cómo realizar funciones y acciones enlazadas y sin enlazar, incluidas acciones personalizadas, mediante la API web de Microsoft Dataverse utilizando JavaScript del lado del cliente.

ⓘ Nota

Este ejemplo implementa las operaciones detalladas en el [Ejemplo de funciones y acciones de la API web](#) y utiliza las construcciones comunes de JavaScript del lado cliente que se describen en [Ejemplos de API web \(JavaScript del lado cliente\)](#)

En esta sección

- [Requisitos previos](#)
- [Ejecute este ejemplo](#)
- [Ejemplo de código](#)

Requisitos previos

Para ejecutar este ejemplo, se requiere lo siguiente:

- Acceder a un entorno de Dataverse.
- Una cuenta de usuario con privilegios para importar soluciones y realizar operaciones CRUD, normalmente un rol de seguridad de administrador o personalizador del sistema.

Ejecute este ejemplo

Para ejecutar esta muestra, descargue el paquete de solución de [aquí](#), extraiga el contenido y ubique el archivo solución administrada

`WebAPIFunctionsandActions_1_0_0_0_managed.zip`. Importe la solución administrada en la organización de Dataverse y vea la página de configuración de la solución para ejecutar el ejemplo. Para obtener instrucciones sobre cómo importar la solución de ejemplo, consulte [ejemplos de API Web \(JavaScript del lado cliente\)](#).

Ejemplo de código

Este ejemplo incluye dos recursos web:

- [WebAPIFunctionsAndActions.html](#)
- [WebAPIFunctionsAndActions.js](#)

WebAPIFunctionsAndActions.html

El recurso web WebAPIFunctionsAndActions.html proporciona el contexto en el que el código JavaScript se ejecutará.

HTML

```
<!DOCTYPE html>
<html>
  <head>
    <title>Microsoft CRM Web API Functions and Actions Example</title>
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=Edge" />
    <script
      src="../ClientGlobalContext.js.aspx"
      type="text/javascript"
    ></script>
    <script src="scripts/es6promise.js"></script>
    <script src="scripts/WebAPIFunctionsAndActions.js"></script>

    <style type="text/css">
      body {
        font-family: "Segoe UI", Tahoma, Geneva, Verdana, sans-serif;
      }

      #preferences {
        border: inset;
        padding: 10px 10px;
      }

      #output_area {
        border: inset;
        background-color: gainsboro;
        padding: 10px 10px;
      }
    </style>
  </head>
  <body>
    <h1>Microsoft CRM Web API Functions and Actions Example</h1>
    <p>
      This page demonstrates the CRM Web API's Functions and Actions using
      JavaScript.
    </p>
  </body>
</html>
```

```

<h2>Instructions</h2>
<p>
    Choose your preferences and run the JavaScript code. Use your
    browser's
    developer tools to view the output written to the console (e.g.: in
    IE11
    or Microsoft Edge, press F12 to load the Developer Tools).
</p>
<form id="preferences">
    <p>
        Remove sample data (Choose whether you want to delete sample data
        created for this sample):<br />
        <input name="removesampleddata" type="radio" value="yes" checked />
Yes
        <input name="removesampleddata" type="radio" value="no" /> No
    </p>
    <input
        type="button"
        name="start_samples"
        value="Start Samples"
        onclick="Sdk.startSample()"
    />
</form>
</body>
</html>

```

WebAPIFunctionsAndActions.js

El recurso web WebAPIFunctionsAndActions.js es la biblioteca JavaScript que define las operaciones que este ejemplo realiza.

JavaScript

```

"use strict";
var Sdk = window.Sdk || {};

/**
 * @function getClientUrl
 * @description Get the client URL.
 * @return {string} The client URL.
 */
Sdk.getClientUrl = function () {
    var context;
    // GetGlobalContext defined by including reference to
    // ClientGlobalContext.js.aspx in the HTML page.
    if (typeof GetGlobalContext != "undefined") {
        context = GetGlobalContext();
    } else {
        if (typeof Xrm != "undefined") {
            // Xrm.Page.context defined within the Xrm.Page object model for form
            scripts.
            context = Xrm.Page.context;
        }
    }
    return context;
}

```

```

        } else {
            throw new Error("Context is not available.");
        }
    }
    return context.getClientUrl();
};

// Global variables
var entitiesToDelete = [] // Entity URIs to be deleted later
// (if user chooses to delete sample data).
var deleteData = true; // Controls whether sample data are deleted at the
end of this sample run.
var clientUrl = Sdk.getClientUrl(); // ie.: https://org.crm.dynamics.com
var webAPIPath = "/api/data/v8.1"; // Path to the web API.
var incidentUri; // Incident created with three closed tasks.
var opportunityUri; // Closed opportunity to re-open before deleting.
var letterUri; // Letter to add to contact's queue.
var myQueueUri; // The contact's queue uri.
var contactUri; // Add a note to this contact.
var CUSTOMERACCOUNTNAME =
    "Account Customer Created in WebAPIFunctionsAndActions sample"; // For
custom action.

/**
 * @function getWebAPIPath
 * @description Get the full path to the Web API.
 * @return {string} The full URL of the Web API.
 */
Sdk.getWebAPIPath = function () {
    return Sdk.getClientUrl() + webAPIPath;
};

/**
 * @function request
 * @description Generic helper function to handle basic XMLHttpRequest
calls.
 * @param {string} action - The request action. String is case-sensitive.
 * @param {string} uri - An absolute or relative URI. Relative URI starts
with a "/".
 * @param {object} data - An object representing an entity. Required for
create and update actions.
 * @param {object} addHeader - An object with header and value properties to
add to the request
 * @returns {Promise} - A Promise that returns either the request object or
an error object.
 */
Sdk.request = function (action, uri, data, addHeader) {
    if (!RegExp(action, "g").test("POST PATCH PUT GET DELETE")) {
        // Expected action verbs.
        throw new Error(
            "Sdk.request: action parameter must be one of the following: " +
            "POST, PATCH, PUT, GET, or DELETE."
        );
    }
    if (!typeof uri === "string") {

```

```

        throw new Error("Sdk.request: uri parameter must be a string.");
    }
    if (RegExp(action, "g").test("POST PATCH PUT") && !data) {
        throw new Error(
            "Sdk.request: data parameter must not be null for operations that
create or modify data."
        );
    }
    if (addHeader) {
        if (
            typeof addHeader.header != "string" ||
            typeof addHeader.value != "string"
        ) {
            throw new Error(
                "Sdk.request: addHeader parameter must have header and value
properties that are strings."
            );
        }
    }
}

// Construct a fully qualified URI if a relative URI is passed in.
if (uri.charAt(0) === "/") {
    uri = clientUrl + webAPIPath + uri;
}

return new Promise(function (resolve, reject) {
    var request = new XMLHttpRequest();
    request.open(action, encodeURI(uri), true);
    request.setRequestHeader("OData-MaxVersion", "4.0");
    request.setRequestHeader("OData-Version", "4.0");
    request.setRequestHeader("Accept", "application/json");
    request.setRequestHeader("Content-Type", "application/json; charset=utf-
8");
    if (addHeader) {
        request.setRequestHeader(addHeader.header, addHeader.value);
    }
    request.onreadystatechange = function () {
        if (this.readyState === 4) {
            request.onreadystatechange = null;
            switch (this.status) {
                case 200: // Success with content returned in response body.
                case 204: // Success with no content returned in response body.
                case 304: // Success with Not Modified
                    resolve(this);
                    break;
                default: // All other statuses are error cases.
                    var error;
                    try {
                        error = JSON.parse(request.response).error;
                    } catch (e) {
                        error = new Error("Unexpected Error");
                    }
                    reject(error);
                    break;
            }
        }
    }
})

```

```

        }
    );
    request.send(JSON.stringify(data));
});
};

/***
 * @function Sdk.startSample
 * @description Initiates a chain of promises to show use of Functions and
Actions with the Web API.
 * Functions and actions represent re-usable operations you can perform
using the Web API.
 * For more info, see
https://msdn.microsoft.com/library/mt607990.aspx#bkmk\_actions
 * The following standard CRM Web API functions and actions are invoked:
 * - WhoAmI, a basic unbound function
 * - GetTimeZoneCodeByLocalizedname, an unbound function that requires
parameters
 * - CalculateTotalTimeIncident, a bound function
 * - WinOpportunity, an unbound action that takes parameters
 * - AddToQueue, a bound action that takes parameters
 * - In addition, a custom bound and an unbound action contained within the
solution are invoked.
*/
Sdk.startSample = function () {
// Initializing.
deleteData = document.getElementsByName("removesampleddata")[0].checked;
entitiesToDelete = []; // Reset the array.

console.log("-- Sample started --");

// Create the CRM entry instances used by this sample program.
Sdk.createRequiredRecords()
.then(function () {
    console.log("-- Working with functions --");
    // Bound and Unbound functions
    // See
https://msdn.microsoft.com/library/gg309638.aspx#bkmk\_boundAndUnboundFunctions

    console.log("Using functions to look up your full name.");
    // Calling a basic unbound function without parameters.
    // Retrieves the user's full name using a series of function requests.
    // - Call WhoAmI via the Sdk.getUsersFullName function.
    // For more info on the WhoAmI function, see
https://msdn.microsoft.com/library/mt607925.aspx
    return Sdk.getUsersFullName();
})
.then(function (fullName) {
    console.log("\tYour full name is: %s\n", fullName);

    console.log("Unbound function: GetTimeZoneCodeByLocalizedname");
    // Calling a basic unbound function with no parameters.
    // Retrieves the time zone code for the specified time zone.
    // Pass parameters to an unbound function by calling the

```

```

GetTimeZoneCodeByLocalizedFunction.
// For more info, see https://msdn.microsoft.com/library/mt607644.aspx
var localizedStandardName = "Pacific Standard Time";
var localeId = 1033;
// Demonstrates best practice of passing parameters.
var uri = [
    "/GetTimeZoneCodeByLocalizedFunction",
    "(LocalizedStandardName=@p1,LocaleId=@p2)",
    "?@p1='" + localizedStandardName + "'&@p2=" + localeId,
];
/* This would also work:
var uri = ["/GetTimeZoneCodeByLocalizedFunction",
    "(LocalizedStandardName=' " + localizedStandardName + "',LocaleId=" +
localeId + ")"];
*/
return Sdk.request("GET", uri.join("")) // Send request.
})
.then(function (request) {
    // Returns GetTimeZoneCodeByLocalizedFunctionResponse ComplexType.
    // For more info, see https://msdn.microsoft.com/library/mt607889.aspx
    var localizedStandardName = "Pacific Standard Time";
    var timeZoneCode = JSON.parse(request.response).TimeZoneCode;
    console.log(
        "\tFunction returned time zone %s, with code '%s'.",
        localizedStandardName,
        timeZoneCode
    );
    console.log("Bound function: CalculateTotalTimeIncident");
    // Calling a basic bound function that requires parameters.
    // Retrieve the total time, in minutes, spent on all tasks associated
    with this incident.
    // - Use CalculateTotalTimeIncident to get the total duration of all
    closed activities.
    // For more info, see https://msdn.microsoft.com/library/mt593054.aspx
    // Note that in a bound function the full function name includes the
    // namespace Microsoft.Dynamics.CRM. Functions that aren't bound must
    not use the full name.
    return Sdk.request(
        "GET",
        incidentUri + "/Microsoft.Dynamics.CRM.CalculateTotalTimeIncident()"
    );
})
.then(function (request) {
    // Returns CalculateTotalTimeIncidentResponse ComplexType.
    // For more info, see https://msdn.microsoft.com/library/mt607924.aspx
    var totalTime = JSON.parse(request.response).TotalTime; //returns 90
    console.log(
        "\tFunction returned %s minutes - total duration of tasks associated
        with the incident.\n",
        totalTime
    );
}

```

```

        console.log("-- Working with Actions --");
        // For more info about Action, see
https://msdn.microsoft.com/library/mt607600.aspx

        console.log("Unbound Action: WinOpportunity");
        // Calling an unbound action that requires parameters.
        // Closes an opportunity and markt it as won.
        // - Update the WinOpportunity (created by
Sdk.createRequiredRecords()) by closing it as won.
        // Use WinOpportunity Action
(https://msdn.microsoft.com/library/mt607971.aspx)
        // This action does not return a value
        var parameters = {
            Status: 3,
            OpportunityClose: {
                subject: "Won Opportunity",
                "opportunityid@odata.bind": opportunityUri,
            },
        };
        return Sdk.request("POST", "/WinOpportunity", parameters);
    )
    .then(function () {
        console.log("\tOpportunity won.");

        console.log("Bound Action: AddToQueue");
        // Calling a bound action that requires parameters.
        // Adds a new letter tracking activity to the current user's queue.
        // The letter was created as part of the Sdk.createRequiredRecords().
        // - Get a reference to the current user.
        // - Get a reference to the letter activity.
        // - Add letter to current user's queue via the bound action
AddToQueue.
        // For more info on AddToQueue, see
https://msdn.microsoft.com/library/mt607880.aspx

        return Sdk.request("GET", "/WhoAmI");
    )
    .then(function (request) {
        var whoAmIResponse = JSON.parse(request.response);
        var myId = whoAmIResponse.UserId;

        // Get a reference to the current user.
        return Sdk.request(
            "GET",
            Sdk.getWebAPIPath() + "/systemusers(" + myId + ")/queueid/$ref"
        );
    )
    .then(function (request) {
        myQueueUri = JSON.parse(request.response)[ "@odata.id" ];

        // Get a reference to the letter activity.
        return Sdk.request("GET", letterUri + "?$select=activityid");
    )
    .then(function (request) {

```

```

        var letterActivityId = JSON.parse(request.response).activityid;

        var parameters = {
            Target: {
                activityid: letterActivityId,
                "@odata.type": "Microsoft.Dynamics.CRM.letter",
            },
        };
        //Adding the letter to the user's default queue.
        return Sdk.request(
            "POST",
            myQueueUri + "/Microsoft.Dynamics.CRM.AddToQueue",
            parameters
        );
    }
    .then(function (request) {
        var queueItemId = JSON.parse(request.response).QueueItemId;
        console.log(
            "\tQueueItemId returned from AddToQueue Action: %s\n",
            queueItemId
        );

        console.log("");
        console.log("Custom action: sample_AddNoteToContact");
        // Add a note to an existing contact.
        // This operation calls a custom action named sample_AddNoteToContact.
        // This custom action is installed when you install this sample's
        solution to your CRM server.
        // - Add a note to an existing contact (e.g.: contactUri)
        // - Get the note info and the contact's full name.
        // For more info, see
https://msdn.microsoft.com/library/mt607600.aspx#bkmk\_customActions
//sample_AddNoteToContact custom action parameters
        var parameters = {
            NoteTitle: "The Title of the Note",
            NoteText: "The text content of the note.",
        };
        return Sdk.request(
            "POST",
            contactUri + "/Microsoft.Dynamics.CRM.sample_AddNoteToContact",
            parameters
        );
    })
    .then(function (request) {
        var annotationid = JSON.parse(request.response).annotationid;
        var annotationUri =
            Sdk.getWebAPIPath() + "/annotations(" + annotationid + ")";
        // The annotation will be deleted with the contact when it is deleted.

        return Sdk.request(
            "GET",
            annotationUri +
            "?$select=subject,notetext&$expand=objectid_contact($select=fullname)"
        );
    })

```

```

})
.then(function (request) {
  var annotation = JSON.parse(request.response);
  console.log(
    "\tA note with the title '%s' and the content '%s' was created and
associated with the contact %s.\n",
    annotation.subject,
    annotation.notetext,
    annotation.objectid_contact.fullname
);

  console.log("Custom action: sample_CreateCustomer");
  // Create a customer of a specified type using the custom action
sample_CreateCustomer.
  // - Shows how create a valid customer of type "account".
  // - Shows how to handle exception from a custom action.

  var parameters = {
    CustomerType: "account",
    AccountName: CUSTOMERACCOUNTNAME,
  };

  // Create the account. This is a valid request
  return Sdk.request("POST", "/sample_CreateCustomer", parameters);
})
.then(function (request) {
  // Retrieve the account we just created
  return Sdk.request(
    "GET",
    "/accounts?$select=name&$filter=name eq '" + CUSTOMERACCOUNTNAME +
    ...
  );
})
.then(function (request) {
  var customerAccount = JSON.parse(request.response).value[0];
  var customerAccountId = customerAccount.accountid;
  var customerAccountIdUri =
    Sdk.getWebAPIPath() + "/accounts(" + customerAccountId + ")";
  entitiesToDelete.push(customerAccountIdUri);
  console.log(
    "\tAccount customer created with the name '%s'",
    customerAccount.name
);

  // Create a contact but uses invalid parameters
  // - Throws an error intentionally
  return new Promise(function (resolve, reject) {
    var parameters = {
      CustomerType: "contact",
      AccountName: CUSTOMERACCOUNTNAME, //not valid for contact
      // e.g.: ContactFirstName and ContactLastName are required when
CustomerType is "contact".
    };
    Sdk.request("POST", "/sample_CreateCustomer", parameters) // This
request is expected to fail.
  });
})

```

```

        .then(function () {
            console.log("Not expected.");
            reject(
                new Error(
                    "Call to sample_CreateCustomer not expected to succeed."
                )
            );
        })
        .catch(function (err) {
            //Expected error
            console.log("\tExpected custom error: " + err.message); //
Custom action can return custom error messages.
            resolve(); // Show the error but resolve the thread so sample
can continue.
        });
    });
})
.then(function () {
    // House cleaning.
    console.log("\n-- Deleting sample data --");
    if (deleteData) {
        return Sdk.deleteEntities();
    } else {
        console.log("Sample data not deleted.");
    }
})
.catch(function (err) {
    console.log("ERROR: " + err.message);
});
};

/***
 * @function Sdk.deleteEntities
 * @description Deletes the entities created by this sample
 */
Sdk.deleteEntities = function () {
    return new Promise(function (resolve, reject) {
        entitiesToDelete.unshift(opportunityUri); // Adding to the beginning so
it will get deleted before the parent account.
        // Re-open the created opportunity so it can be deleted.
        Sdk.request("PATCH", opportunityUri, { statecode: 0, statuscode: 2 })
            .then(function () {
                // Get the opportunityclose URI so it can be deleted
                return Sdk.request(
                    "GET",
                    opportunityUri + "/Opportunity_OpportunityClose/$ref"
                );
            })
            .then(function (request) {
                var opportunityCloseUri = JSON.parse(request.response).value[0][
                    "@odata.id"
                ];
                // Adding to the opportunityclose URI it will get deleted before the
opportunity.
            });
    });
}

```

```

        entitiesToDelete.unshift(opportunityCloseUri);

        /*
        These deletions have to be done consecutively in a specific order to avoid
        a Generic SQL error
        which can occur because of relationship behavior actions for the delete
        event.
        */

        return Sdk.request("DELETE", entitiesToDelete[0]);
//opportunityclose
})
.then(function () {
    console.log(entitiesToDelete[0] + " Deleted");
    return Sdk.request("DELETE", entitiesToDelete[1]); //opportunity
})
.then(function () {
    console.log(entitiesToDelete[1] + " Deleted");
    return Sdk.request("DELETE", entitiesToDelete[2]); //account
})
.then(function () {
    console.log(entitiesToDelete[2] + " Deleted");
    return Sdk.request("DELETE", entitiesToDelete[3]); //Fourth Coffee
account
})
.then(function () {
    console.log(entitiesToDelete[3] + " Deleted");
    return Sdk.request("DELETE", entitiesToDelete[4]); //Letter
})
.then(function () {
    console.log(entitiesToDelete[4] + " Deleted");
    return Sdk.request("DELETE", entitiesToDelete[5]); //Contact
})
.then(function () {
    console.log(entitiesToDelete[5] + " Deleted");
    return Sdk.request("DELETE", entitiesToDelete[6]); //AccountCustomer
})
.then(function () {
    console.log(entitiesToDelete[6] + " Deleted");
    resolve();
})
.catch(function (err) {
    reject(new Error("Error from Sdk.deleteEntities: " + err.message));
});
});

/**
 * @function Sdk.getUsersFullName
 * @description Retrieves the current user's full name.
 * @returns {Promise} - A Promise that returns the full name of the user
 */
Sdk.getUsersFullName = function () {
    return new Promise(function (resolve, reject) {
        //Use WhoAmI Function (https://msdn.microsoft.com/library/mt607925.aspx)
    });
}

```

```

Sdk.request("GET", "/WhoAmI")
    .then(function (request) {
        //Returns WhoAmIResponse ComplexType
(https://msdn.microsoft.com/library/mt607982.aspx)
        var myId = JSON.parse(request.response).UserId;
        //Retrieve the systemuser Entity fullname property
(https://msdn.microsoft.com/library/mt608065.aspx)
        return Sdk.request(
            "GET",
            "/systemusers(" + myId + ")?$select=fullname"
        );
    })
    .then(function (request) {
        //Return the users full name
        resolve(JSON.parse(request.response).fullname);
    })
    .catch(function (err) {
        reject("Error in Sdk.getUsersFullName function: " + err.message);
    });
});

/***
 * @function Sdk.createRequiredRecords
 * @description Creates data required by this sample program.
 * - Create an account with three 30 minute tasks.
 * - Create another account associated with an opportunity.
 * - Create a letter.
 * - Create a contact.
 * @returns {Promise} - resolve the promise if all goes well; reject otherwise.
 */
Sdk.createRequiredRecords = function () {
    console.log("-- Creating sample data --");
    // Create a parent account, an associated incident with three
    // associated tasks(required for CalculateTotalTimeIncident).
    return new Promise(function (resolve, reject) {
        Sdk.createAccountWithIncidentAndThree30MinuteClosedTasks()
            .then(function (iUri) {
                incidentUri = iUri;

                //Create another account and associated opportunity (required for
                //CloseOpportunityAsWon).
                return Sdk.createAccountWithOpportunityToWin();
            })
            .then(function (oUri) {
                opportunityUri = oUri;

                // Create a letter to use with AddToQueue action.
                var letter = {
                    description: "Example letter",
                };
                return Sdk.request("POST", "/letters", letter);
            })
            .then(function (request) {

```

```

        letterUri = request.getResponseHeader("OData-EntityId");
        entitiesToDelete.push(letterUri);

        // Create a contact to use with custom action
sample_AddNoteToContact
    var contact = {
        firstname: "Jon",
        lastname: "Fogg",
    };
    return Sdk.request("POST", "/contacts", contact);
})
.then(function (request) {
    contactUri = request.getResponseHeader("OData-EntityId");
    entitiesToDelete.push(contactUri);

    resolve();
})
.catch(function (err) {
    reject("Error in Sdk.createRequiredRecords function: " +
err.message);
});
});

/***
 * @function Sdk.createAccountwithIncidentAndThree30MinuteClosedTasks
 * @description Create an account and associate three 30 minute tasks. Close
the tasks.
 * @returns {Promise} - A Promise that returns the uri of an incident
created.
 */
Sdk.createAccountWithIncidentAndThree30MinuteClosedTasks = function () {
    return new Promise(function (resolve, reject) {
        var iUri; // incidentUri
        // Create a parent account for the incident.
        Sdk.request("POST", "/accounts", { name: "Fourth Coffee" })
            .then(function (request) {
                // Capture the URI of the created account so it can be deleted
later.
                var accountUri = request.getResponseHeader("OData-EntityId");
                entitiesToDelete.push(accountUri);
                // Define an incident associated with the account with three related
tasks.
                // Each task has a 30 minute duration.
                var incident = {
                    title: "Sample Case",
                    "customerid_account@odata.bind": accountUri,
                    Incident_Tasks: [
                        {
                            subject: "Task 1",
                            actualdurationminutes: 30,
                        },
                        {
                            subject: "Task 2",
                            actualdurationminutes: 30,
                        }
                    ]
                };
                return Sdk.request("POST", "/incidents", incident);
            })
            .then(function (request) {
                var incidentUri = request.getResponseHeader("OData-EntityId");
                entitiesToDelete.push(incidentUri);
                // Create three tasks associated with the incident.
                var tasks = [
                    {
                        subject: "Task 1A",
                        actualdurationminutes: 30,
                        "customerid_incident@odata.bind": incidentUri
                    },
                    {
                        subject: "Task 1B",
                        actualdurationminutes: 30,
                        "customerid_incident@odata.bind": incidentUri
                    },
                    {
                        subject: "Task 1C",
                        actualdurationminutes: 30,
                        "customerid_incident@odata.bind": incidentUri
                    }
                ];
                return Sdk.request("POST", "/tasks", tasks);
            })
            .then(function (request) {
                var taskUris = request.response;
                entitiesToDelete.push(taskUris);
                resolve();
            })
            .catch(function (err) {
                reject("Error in Sdk.createAccountwithIncidentAndThree30MinuteClosedTasks function: " +
err.message);
            });
    });
}

```

```

        },
        {
            subject: "Task 3",
            actualdurationminutes: 30,
        },
    ],
});
// Create the incident and related tasks.
return Sdk.request("POST", "/incidents", incident);
})
.then(function (request) {
    iUri = request.getResponseHeader("OData-EntityId");

    // Retrieve references to the tasks created.
    return Sdk.request("GET", iUri + "/Incident_Tasks/$ref");
})
.then(function (request) {
    // Capture the URL for the three tasks in this array.
    var taskReferences = [];
    JSON.parse(request.response).value.forEach(function (tr) {
        taskReferences.push(tr["@odata.id"]);
    });
    // An array to hold a set of promises.
    var promises = [];
    // The data to use to update the tasks so that they are closed.
    var update = {
        statecode: 1, //Completed
        statuscode: 5, //Completed
    };
    // Fill the array with promises
    taskReferences.forEach(function (tr) {
        promises.push(Sdk.request("PATCH", tr, update));
    });
    // When all the promises resolve, return a promise.
    return Promise.all(promises);
})
.then(function () {
    // Return the incident URI to the calling code.
    resolve(iUri);
})
.catch(function (err) {
    // Differentiate the message for any error returned by this
    // function.
    reject(
        new Error(
            "ERROR in
Sdk.createAccountwithIncidentAndThree30MinuteClosedTasks function: " +
            err.message
        )
    );
});
});
});
};

/**

```

```

* @function Sdk.createAccountwithOpportunityToWin
* @description Create an account and an associated opportunity.
* @returns {Promise} - A Promise that returns the uri of an opportunity.
*/
Sdk.createAccountWithOpportunityToWin = function () {
    return new Promise(function (resolve, reject) {
        var accountUri;
        var account = {
            name: "Sample Account for WebAPIFunctionsAndActions sample",
            opportunity_customer_accounts: [
                {
                    name: "Opportunity to win",
                },
            ],
        };
        Sdk.request("POST", "/accounts", account) // Create the account.
            .then(function (request) {
                accountUri = request.getResponseHeader("OData-EntityId");
                entitiesToDelete.push(accountUri);

                // Retrieve the opportunity's reference.
                return Sdk.request(
                    "GET",
                    accountUri + "/opportunity_customer_accounts/$ref"
                );
            })
            .then(function (request) {
                var oUri = JSON.parse(request.response).value[0][ "@odata.id" ];
                resolve(oUri); // Return the opportunity's uri.
            })
            .catch(function (err) {
                reject(
                    new Error(
                        "Error in Sdk.createAccountwithOpportunityToWin: " + err.message
                    )
                );
            });
    });
};

```

Vea también

[Usar la API web de Dataverse](#)

[Usar funciones de la API web](#)

[Usar acciones de la API web](#)

[Ejemplos de la API web](#)

[Ejemplo de funciones y acciones de la API web](#)

[Ejemplo de funciones y acciones de la API web \(C#\)](#)

[Ejemplos de la API web \(JavaScript del lado del cliente\)](#)

[Ejemplo de operaciones básicas de la API web \(JavaScript del lado del cliente\)](#)

[Ejemplo de datos de consulta de la API web \(JavaScript del lado del cliente\)](#)

[Ejemplo de operaciones condicionales de la API web \(JavaScript del lado del cliente\)](#)

 **Nota**

¿Puede indicarnos sus preferencias de idioma de documentación? **Realice una breve encuesta.**  (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales (**declaración de privacidad** ).

Ejemplo de datos de consulta de la API web (JavaScript del lado del cliente)

Artículo • 23/02/2023

Este ejemplo muestra cómo realizar solicitudes de consultas básicas mediante la API web de Microsoft Dataverse usando JavaScript del lado del cliente.

ⓘ Nota

Este ejemplo implementa las operaciones detalladas en el [Ejemplo de datos de consulta de la API web](#) y utiliza las construcciones comunes del lado cliente de JavaScript que se describen en [Ejemplos de API web \(JavaScript del lado cliente\)](#)

Requisitos previos

Para ejecutar este ejemplo, se requiere lo siguiente:

- Acceder a un entorno de Dataverse.
- Una cuenta de usuario con privilegios para importar soluciones y realizar operaciones CRUD, normalmente un rol de seguridad de administrador o personalizador del sistema.

Ejecute este ejemplo

Para ejecutar esta muestra, descargue el paquete de solución [aquí](#). Extraiga el contenido de la muestra y localice el archivo de solución administrada `WebAPIQueryData_1_0_0_0_managed.zip`. Importe la solución administrada en la organización de Dataverse y ejecute el ejemplo. Para obtener instrucciones sobre cómo importar la solución de ejemplo, consulte [ejemplos de API Web \(JavaScript del lado cliente\)](#).

Ejemplo de código

Este ejemplo incluye dos recursos web:

- [WebAPIQuery.html](#)
- [WebAPIQuery.js](#)

WebAPIQuery.html

El recurso web WebAPIQuery.html proporciona el contexto en el que el código JavaScript se ejecutará.

HTML

```
<!DOCTYPE html>
<html>
  <head>
    <title>Microsoft CRM Web API Query Example</title>
    <meta charset="utf-8" />
    <script
      src="../ClientGlobalContext.js.aspx"
      type="text/javascript"
    ></script>
    <script src="scripts/es6promise.js"></script>
    <script src="scripts/WebAPIQuery.js"></script>

    <style type="text/css">
      body {
        font-family: "Segoe UI", Tahoma, Geneva, Verdana, sans-serif;
      }

      #preferences {
        border: inset;
        padding: 10px 10px;
      }

      #output_area {
        border: inset;
        background-color: gainsboro;
        padding: 10px 10px;
      }
    </style>
  </head>
  <body>
    <h1>Microsoft CRM Web API Query Example</h1>
    <p>
      This page demonstrates the CRM Web API's Query operations using
      JavaScript.
    </p>

    <h2>Instructions</h2>
    <p>
      Choose your preferences and run the JavaScript code. Use your
      browser's
      developer tools to view the output written to the console (e.g.: in IE
      11
      or Microsoft Edge, press F12 to load the Developer Tools).
    </p>
    <form id="preferences">
      <p>
```

```

        Remove sample data (Choose whether you want to delete sample data
        created during this execution):
        <br />
        <input name="removesampleddata" type="radio" value="yes" checked />
Yes
        <input name="removesampleddata" type="radio" value="no" /> No
</p>
<input
    type="button"
    name="start_samples"
    value="Start Sample"
    onclick="Sdk.startSample()"
/>
</form>
</body>
</html>

```

WebAPIQuery.js

El recurso web WebAPIQuery.js es la biblioteca JavaScript que define las operaciones que este ejemplo realiza.

JavaScript

```

"use strict";
var Sdk = window.Sdk || {};
/**
 * @function getClientUrl
 * @description Get the client URL.
 * @returns {string} The client URL.
 */
Sdk.getClientUrl = function () {
    var context;
    // GetGlobalContext defined by including reference to
    // ClientGlobalContext.js.aspx in the HTML page.
    if (typeof GetGlobalContext != "undefined")
    { context = GetGlobalContext(); }
    else
    {
        if (typeof Xrm != "undefined") {
            // Xrm.Page.context defined within the Xrm.Page object model for
            form scripts.
            context = Xrm.Page.context;
        }
        else { throw new Error("Context is not available."); }
    }
    return context.getClientUrl();
}

// Global variables.
var entitiesToDelete = [];                                // Entity URIs to be deleted (if
user chooses to delete sample data)

```

```

var deleteData = true; // Delete data by default unless user chooses not to delete.
var clientUrl = Sdk.getClientUrl(); // e.g.:
https://org.crm.dynamics.com
var webAPIPath = "/api/data/v8.1"; // Path to the web API.
var account1Uri; // e.g.: Contoso Inc (sample)
var contact1Uri; // e.g.: Yvonne McKey (sample)
var page2Uri; // URI of next page in pagination sample.

// Entity properties to select in a request.
var contactProperties = ["fullname", "jobtitle", "annualincome"];
var accountProperties = ["name"];
var taskProperties = ["subject", "description"];

/**
 * @function request
 * @description Generic helper function to handle basic XMLHttpRequest calls.
 * @param {string} action - The request action. String is case-sensitive.
 * @param {string} uri - An absolute or relative URI. Relative URI starts with a "/".
 * @param {object} data - An object representing an entity. Required for create and update action.
 * @param {boolean} formattedValue - If "true" then include formatted value; "false" otherwise.
 *      For more info on formatted value, see:
 *
https://msdn.microsoft.com/library/gg334767.aspx#bkmk\_includeFormattedValues
 * @param {number} maxPageSize - Indicate the page size. Default is 10 if not defined.
 * @returns {Promise} - A Promise that returns either the request object or an error object.
 */
Sdk.request = function (action, uri, data, formattedValue, maxPageSize) {
    if (!RegExp(action, "g").test("POST PATCH PUT GET DELETE")) { // Expected action verbs.
        throw new Error("Sdk.request: action parameter must be one of the following: " +
            "POST, PATCH, PUT, GET, or DELETE.");
    }
    if (!typeof uri === "string") {
        throw new Error("Sdk.request: uri parameter must be a string.");
    }
    if ((RegExp(action, "g").test("POST PATCH PUT")) && (data === null || data === undefined)) {
        throw new Error("Sdk.request: data parameter must not be null for operations that create or modify data.");
    }
    if (maxPageSize === null || maxPageSize === undefined) {
        maxPageSize = 10; // Default limit is 10 entities per page.
    }

    // Construct a fully qualified URI if a relative URI is passed in.
    if (uri.charAt(0) === "/") {

```

```

        uri = clientUrl + webAPIPath + uri;
    }

    return new Promise(function (resolve, reject) {
        var request = new XMLHttpRequest();
        request.open(action, encodeURI(uri), true);
        request.setRequestHeader("OData-MaxVersion", "4.0");
        request.setRequestHeader("OData-Version", "4.0");
        request.setRequestHeader("Accept", "application/json");
        request.setRequestHeader("Content-Type", "application/json;
charset=utf-8");
        request.setRequestHeader("Prefer", "odata.maxpagesize=" +
maxPageSize);
        if (formattedValue) {
            request.setRequestHeader("Prefer",
            "odata.include-
annotations=OData.Community.Display.V1.FormattedValue");
        }
        request.onreadystatechange = function () {
            if (this.readyState === 4) {
                request.onreadystatechange = null;
                switch (this.status) {
                    case 200: // Success with content returned in response
body.
                    case 204: // Success with no content returned in
response body.
                        resolve(this);
                        break;
                    default: // All other statuses are unexpected so are
treated like errors.
                        var error;
                        try {
                            error = JSON.parse(request.response).error;
                        } catch (e) {
                            error = new Error("Unexpected Error");
                        }
                        reject(error);
                        break;
                }
            }
        };
        request.send(JSON.stringify(data));
    });
};

/***
 * @function output
 * @description Generic helper function to output data to console.
 * @param {array} collection - Array of entities.
 * @param {string} label - Text label for what the collection contains.
 * @param {array} properties - Array of properties appropriate for the
collection.
 */
Sdk.output = function (collection, label, properties) {
    console.log(label);
}

```

```

collection.forEach(function (row, i) {
    var prop = [];
    properties.forEach(function (p) {
        var f = p + "@odata.Community.Display.V1.FormattedValue";
        prop.push((row[f] ? row[f] : row[p])); // Get formatted value if
one exists for this property.
    })
    console.log("\t%s) %s", i + 1, prop.join(", "));
});
}

/***
 * @function startSample
 * @description Runs the sample.
 * This sample demonstrates basic query operations.
 * Results are sent to the debugger's console window.
 */
Sdk.startSample = function () {
    // Initializing...
    deleteData = document.getElementsByName("removesampleddata")[0].checked;
    entitiesToDelete = []; //Reset the array.
    account1Uri = "";
    contact1Uri = "";
    page2Uri = "";

    console.log("-- Sample started --");
    console.log("Create sample data:");
    // Add some data to the CRM server so we can query against it.
    // Using Deep Insert, we create all the sample data in one request.
    // Data structure:
    //   Accounts
    //     |--- primarycontactid
    //           |--- Contact_Tasks (3 tasks)
    //           |--- Account_Tasks (3 tasks)
    //           |--- contact_customer_accounts (9 child contacts, each with 3
tasks)
    //           |--- Contacts
    //               |--- Contact_Tasks
    //
    var sampleData = {
        "name": "Contoso, Ltd. (sample)",
        "primarycontactid": {
            "firstname": "Yvonne", "lastname": "McKay (sample)", "jobtitle": "Coffee Master",
            "annualincome": 45000, "Contact_Tasks": [
                { "subject": "Task 1", "description": "Task 1 description" },
                { "subject": "Task 2", "description": "Task 2 description" },
                { "subject": "Task 3", "description": "Task 3 description" }
            ]
        }, "Account_Tasks": [
            { "subject": "Task 1", "description": "Task 1 description" },
            { "subject": "Task 2", "description": "Task 2 description" },
            { "subject": "Task 3", "description": "Task 3 description" }
        ],
        "contact_customer_accounts": [

```

```
{
    "firstname": "Susanna", "lastname": "Stubberod (sample)",
"jobtitle": "Senior Purchaser",
    "annualincome": 52000, "Contact_Tasks": [
{ "subject": "Task 1", "description": "Task 1 description" },
{ "subject": "Task 2", "description": "Task 2 description" },
{ "subject": "Task 3", "description": "Task 3 description" }
]
},
{
    "firstname": "Nancy", "lastname": "Anderson (sample)",
"jobtitle": "Activities Manager",
    "annualincome": 55500, "Contact_Tasks": [
{ "subject": "Task 1", "description": "Task 1 description" },
{ "subject": "Task 2", "description": "Task 2 description" },
{ "subject": "Task 3", "description": "Task 3 description" }
]
},
{
    "firstname": "Maria", "lastname": "Cambell (sample)",
"jobtitle": "Accounts Manager",
    "annualincome": 31000, "Contact_Tasks": [
{ "subject": "Task 1", "description": "Task 1 description" },
{ "subject": "Task 2", "description": "Task 2 description" },
{ "subject": "Task 3", "description": "Task 3 description" }
]
},
{
    "firstname": "Nancy", "lastname": "Anderson (sample)",
"jobtitle": "Logistics Specialist",
    "annualincome": 63500, "Contact_Tasks": [
{ "subject": "Task 1", "description": "Task 1 description" },
{ "subject": "Task 2", "description": "Task 2 description" },
{ "subject": "Task 3", "description": "Task 3 description" }
]
},
{
    "firstname": "Scott", "lastname": "Konersmann (sample)",
"jobtitle": "Accounts Manager",
    "annualincome": 38000, "Contact_Tasks": [
{ "subject": "Task 1", "description": "Task 1 description" },
{ "subject": "Task 2", "description": "Task 2 description" },
{ "subject": "Task 3", "description": "Task 3 description" }
]
},
{
    "firstname": "Robert", "lastname": "Lyon (sample)",
"jobtitle": "Logistics Specialist",
    "annualincome": 63500, "Contact_Tasks": [
{ "subject": "Task 1", "description": "Task 1 description" },
{ "subject": "Task 2", "description": "Task 2 description" },
{ "subject": "Task 3", "description": "Task 3 description" }
]
}
]
```

```

        "jobtitle": "Senior Technician",
            "annualincome": 78000, "Contact_Tasks": [
                { "subject": "Task 1", "description": "Task 1 description"
},
                { "subject": "Task 2", "description": "Task 2 description"
},
                { "subject": "Task 3", "description": "Task 3 description" }
            ]
        },
        {
            "firstname": "Paul", "lastname": "Cannon (sample)",
"jobtitle": "Ski Instructor",
            "annualincome": 68500, "Contact_Tasks": [
                { "subject": "Task 1", "description": "Task 1 description"
},
                { "subject": "Task 2", "description": "Task 2 description"
},
                { "subject": "Task 3", "description": "Task 3 description" }
            ]
        },
        {
            "firstname": "Rene", "lastname": "Valdes (sample)",
"jobtitle": "Data Analyst III",
            "annualincome": 86000, "Contact_Tasks": [
                { "subject": "Task 1", "description": "Task 1 description"
},
                { "subject": "Task 2", "description": "Task 2 description"
},
                { "subject": "Task 3", "description": "Task 3 description" }
            ]
        },
        {
            "firstname": "Jim", "lastname": "Glynn (sample)",
"jobtitle": "Senior International Sales Manager",
            "annualincome": 81400, "Contact_Tasks": [
                { "subject": "Task 1", "description": "Task 1 description"
},
                { "subject": "Task 2", "description": "Task 2 description"
},
                { "subject": "Task 3", "description": "Task 3 description" }
            ]
        }
    ];
};

var uri = "/accounts"; // A relative URI to the account entity.
Sdk.request("POST", uri, sampleData) // Adding sample data so we can
query against it.
.then(function (request) {
    // Process request.
    account1Uri = request.getResponseHeader("OData-EntityId");
    entitiesToDelete.push(account1Uri); // To delete later.
    console.log("Account 'Contoso, Ltd. (sample)' created with 1 primary
contact and 9 associated contacts.");
}

```

```

        // Get primary contact info.
        // Most queries are done using this contact.
        var uri = account1Uri + "/primarycontactid/$ref"; // Request for the
URI only.
        return Sdk.request("GET", uri);
    })
    .then(function (request) {
        contact1Uri = JSON.parse(request.response)[ "@odata.id" ];
        entitiesToDelete.push(contact1Uri); // To delete later.
        console.log("Has primary contact 'Yvonne McKay (sample)' with URI:
%s\n", contact1Uri);

        // Basic query:
        // Query using $select option against a contact entity to get the
properties you want.
        // For performance best practice, always use $select otherwise all
properties are returned.
        console.log("-- Basic Query --");
        var query = "?$select=" + contactProperties.join(); // Array defined
in the global scope.
        return Sdk.request("GET", contact1Uri + query, null, true);
    })
    .then(function (request) {
        var contact1 = JSON.parse(request.response);
        console.log("Contact basic info:\n\tfullname: '%s'\n\tjobtitle:
'%s'\n\tAnnualincome: '%s' (unformatted)",
            contact1.fullname, contact1.jobtitle, contact1.annualincome);
        console.log("\tAnnualincome: %s (formatted)\n",
            contact1["annualincome@OData.Community.Display.V1.FormattedValue"]);

        // Filter criteria:
        // Applying filters to get targeted data.
        // 1) Using standard query functions (e.g.: contains, endswith,
startswith)
        // 2) Using CRM query functions (e.g.: LastXhours, Last7Days, Today,
Between, In, ...)
        // 3) Using filter operators and logical operators (e.g.: eq, ne,
gt, and, or, etc...)
        // 4) Set precedence using parenthesis (e.g.: ((criteria1) and
(criteria2)) or (criteria3))
        // For more info, see:
https://msdn.microsoft.com/library/gg334767.aspx#bkmk\_filter
        console.log("-- Filter Criteria --");

        // Filter 1: Using standard query functions to filter results.
        // In this operation, we will query for all contacts with fullname
containing the string "(sample)".
        var filter = "&$filter=contains(fullname,'(sample)')";
        var query = "?$select=" + contactProperties.join() + filter;
        return Sdk.request("GET", "/contacts" + query, null, true);
    })
    .then(function (request) {
        var collection = JSON.parse(request.response).value;
        Sdk.output(collection, "Contacts filtered by fullname containing

```

```

'(sample)':", contactProperties);

        // Filter 2: Using CRM query functions to filter results.
        // In this operation, we will query for all contacts that was
        created in the last hour.
        // For complete list of CRM query functions, see:
        // https://msdn.microsoft.com/library/mt607843.aspx
        var filter =
"&$filter=Microsoft.Dynamics.CRM.LastXHours(PropertyName='createdon',Propert
yValue='1')";
        var query = "?$select=" + contactProperties.join() + filter;
        return Sdk.request("GET", "/contacts" + query, null, true); //
Remember page size limit is set to 10.
    })
    .then(function(request){
        var collection = JSON.parse(request.response).value;
        Sdk.output(collection, "Contacts that were created within the last
1hr:", contactProperties);

        // Filter 3: Using operators
        // Building on the previous operation, we will further limit the
        results by the contact's income.
        // For more info on standard filter operators, see:
        // https://msdn.microsoft.com/library/gg334767.aspx#bkmk_filter
        var filter = "&$filter=contains(fullname,'(sample)') and
annualincome gt 55000";
        var query = "?$select=" + contactProperties.join() + filter;
        return Sdk.request("GET", "/contacts" + query, null, true);
    })
    .then(function (request) {
        var collection = JSON.parse(request.response).value;
        Sdk.output(collection, "Contacts filtered by fullname and
annualincome (<$55,000):", contactProperties);

        // Filter 4: Set precedence using parenthesis.
        // Continue building on the previous operation, we will further
        limit results by job title.
        // Parenthesis and the order of filter statements can impact results
        returned.
        var filter = "&$filter=contains(fullname,'(sample)') " +
            "and (contains(jobtitle,'senior') or
contains(jobtitle,'specialist')) and annualincome gt 55000";
        var query = "?$select=" + contactProperties.join() + filter;
        return Sdk.request("GET", "/contacts" + query, null, true);
    })
    .then(function (request) {
        var collection = JSON.parse(request.response).value;
        Sdk.output(collection, "Contacts filtered by fullname, annualincome
and jobtitle (Senior or Specialist):",
        contactProperties);

        // Order results:
        // Filtered results can be order in descending or ascending order.
        console.log("\n-- Order Results --");
        var filter = "&$filter=contains(fullname,'(sample)') " +

```

```

        "&$orderby=jobtitle asc, annualincome desc";
    var query = "?$select=" + contactProperties.join() + filter;
    return Sdk.request("GET", "/contacts" + query, null, true);
})
.then(function (request) {
    var collection = JSON.parse(request.response).value;
    Sdk.output(collection, "Contacts ordered by jobtitle (Ascending) and
annualincome (descending):",
        contactProperties);

    // Parameterized Aliases.
    // Aliases can be used as parameters in a query. These parameters
    can be used in $filter and $orderby options.
    // Using the previous operation as basis, parameterizing the query
    will give us the same results.
    // For more info, see:
https://msdn.microsoft.com/library/gg309638.aspx#bkmk\_passParametersToFunctions
    console.log("\n-- Parameterized Aliases --");
    var filter = "&$filter=contains(@p1,'(sample)') " +
        "&$orderby=@p2 asc, @p3
desc&@p1=fullname&@p2=jobtitle&@p3=annualincome";
    var query = "?$select=" + contactProperties.join() + filter;
    return Sdk.request("GET", "/contacts" + query, null, true);
})
.then(function (request) {
    var collection = JSON.parse(request.response).value;
    Sdk.output(collection, "Contacts list using parameterized aliases:",

contactProperties);

    // Limit records returned.
    // To further limit the records returned, use the $top query option.
    // Specifying a limit number for $top will return at most that
    number of results per request.
    // Extra results are ignored.
    console.log("\n-- Top Results --");
    var filter = "&$filter=contains(fullname,'(sample)')&$top=5";
    var query = "?$select=" + contactProperties.join() + filter;
    return Sdk.request("GET", "/contacts" + query, null, true);
})
.then(function (request) {
    var collection = JSON.parse(request.response).value;
    Sdk.output(collection, "Contacts top 5 results:",

contactProperties);

    // Result count.
    // Count the number of results matching the filter criteria.
    // 1) Get a count of a collection without the data.
    // 2) Get a count along with the data.
    // HINT: Use count together with the "odata.maxpagesize" to
    calculate the number of pages in the query.
    // NOTE: CRM has a max record limit of 5000 records per response.
    console.log("\n-- Result Count --");
    return Sdk.request("GET", "/contacts/$count"); // Count is returned
in response body.

```

```

        })
        .then(function (request) {
            console.log("The contacts collection has %s contacts.",
request.response); // Count maximum is 5000.

            // 2) Get filtered result with a count
            var filter = "&$filter=contains(jobtitle,'senior') or
contains(jobtitle, 'manager')&&$count=true";
            var query = "?$select=" + contactProperties.join() + filter;
            return Sdk.request("GET", "/contacts" + query, null, true);
        })
        .then(function (request) {
            var count = JSON.parse(request.response)[ "@odata.count" ];
            console.log("%s contacts have either 'Manager' or 'Senior'
designation in their jobtitle.", count);
            var collection = JSON.parse(request.response).value;
            Sdk.output(collection, "Manager or Senior:", contactProperties);

            // Pagination:
            // For large data sets, you can limit the number of records returned
            // per page.
            // Then offer a "next page" and "previous page" links for users to
            // browse through all the data.
            // NOTE: This is why you should not use $top with maxpagesize. $top
            // will limit results returned
            // preventing you from accessing all possible results in the
            // query.
            // For example: If your query has 10 entities in the result
            // and you limit your result to $top=5
            // then, you can't get to the remaining 5 results; but with
            // "maxpagesize" (without $top), you can.
            // HINT: Save the URI of the current page so users can go "next" and
            // "previous".
            console.log("\n-- Pagination --");
            var filter = "&$filter=contains(fullname,'(sample)')&&$count=true";
            var query = "?$select=" + contactProperties.join() + filter;
            return Sdk.request("GET", "/contacts" + query, null, true, 4); // 4
records per page.
        })
        .then(function (request) {
            var count = JSON.parse(request.response)[ "@odata.count" ];
            var maxpages = Math.ceil(count / 4);
            console.log("Contacts total: %s \tContacts per page: %s.\tOutputting
first 2 pages.", count, 4);
            var collection = JSON.parse(request.response).value;
            Sdk.output(collection, "Page 1 of " + maxpages + ":", contactProperties);

            // Getting the next page.
            page2Uri = JSON.parse(request.response)[ "@odata.nextLink" ]; // This
URI is already encoded.
            return Sdk.request("GET", decodeURI(page2Uri), null, true, 4); // URI
            // re-encoded in the request function.
        })
        .then(function (request) {

```

```

        var count = JSON.parse(request.response)[ "@odata.count" ];
        var maxpages = Math.ceil(count / 4);
        var collection = JSON.parse(request.response).value;
        Sdk.output(collection, "Page 2 of " + maxpages + ":" ,
contactProperties);

        // Using expand option to retrieve additional information.
        // It is common for entities to have associations with other
entities in the system and you might want
        // to also retrieve this information in the same request. To
retrieve information on associated entities,
        // use the $expand query option on navigation properties.
        // 1) Expand using single-valued navigation properties (e.g.: via
the 'primarycontactid')
        // 2) Expand using partner property (e.g.: from contact to account
via the 'account_primary_contact')
        // 3) Expand using collection-valued navigation properties (e.g.:
via the 'contact_customer_accounts')
        // 4) Expand using multiple navigation property types in a single
request.

        // NOTE: Expansions can only go 1 level deep.
        // For performance best practice, always use $select statement in
an expand option.
        console.log("\n-- Expanding Results --");

        // 1) Expand using single-valued navigation properties (e.g.: via
the 'primarycontactid')
        var expand = "&$expand=primarycontactid($select=" +
contactProperties.join() + ")";
        var query = "?$select=" + accountProperties.join() + expand;
        return Sdk.request("GET", account1Uri + query, null, true);
    })
    .then(function (request) {
        var account = JSON.parse(request.response);
        var str = "Account '%s' has the following primary contact
person:\n\t" +
            "Fullname: '%s' \n\tJobtitle: '%s' \n\tAnnualincome: '%s'";
        console.log(str, account.name,
            account.primarycontactid.fullname,
            account.primarycontactid.jobtitle,
            account.primarycontactid.annualincome);

        // 2) Expand using partner property (e.g.: from contact to account
via the 'account_primary_contact')
        var expand = "&$expand=account_primary_contact($select=" +
accountProperties.join() + ")";
        var query = "?$select=" + contactProperties.join() + expand;
        return Sdk.request("GET", contact1Uri + query, null, true);
    })
    .then(function (request) {
        var contact = JSON.parse(request.response);
        var label = "Contact '" + contact.fullname + "' is the primary
contact for the following accounts:";
        Sdk.output(contact.account_primary_contact, label,
accountProperties);
    })
}

```

```

        // 3) Expand using collection-valued navigation properties (e.g.:
        // via the 'contact_customer_accounts')
        var expand = "&$expand=contact_customer_accounts($select=" +
contactProperties.join() + ")"
        var query = "?$select=" + accountProperties.join() + expand;
        return Sdk.request("GET", account1Uri + query, null, true);
    })
    .then(function (request) {
        var account = JSON.parse(request.response);
        var label = "Account '" + account.name + "' has the following
        contact customers:";
        var collection = account.contact_customer_accounts;
        Sdk.output(collection, label, contactProperties);

        // 4) Expand using multiple navigation property types in a single
        request.
        // For example: expanding on primarycontactid,
        contact_customer_accounts, and Account_Tasks.
        console.log("\n-- Expanding multiple property types in one request -
        - ");
        var expand = "&$expand=primarycontactid($select=" +
contactProperties.join() + "," +
            "contact_customer_accounts($select=" + contactProperties.join()
+ "," +
                "Account_Tasks($select=" + taskProperties.join() + ")");
        var query = "?$select=" + accountProperties.join() + expand;
        return Sdk.request("GET", account1Uri + query, null, true);
    })
    .then(function (request) {
        var account = JSON.parse(request.response);
        var label = "Account '%s' has the following primary contact
        person:\n\t" +
            "Fullname: '%s' \n\tJobtitle: '%s' \n\tAnnualincome: '%s'";
        console.log(label, account.name,
            account.primarycontactid.fullname,
            account.primarycontactid.jobtitle,
            account.primarycontactid.annualincome);

        // Handling each collection separately.
        label = "Account '" + account.name + "' has the following related
        contacts:";
        var collection = account.contact_customer_accounts;
        Sdk.output(collection, label, contactProperties);

        label = "Account '" + account.name + "' has the following tasks:";
        collection = account.Account_Tasks;
        Sdk.output(collection, label, taskProperties);

        // FetchXML
        // Using FetchXML to query for all contacts whose fullname contains
        '(sample)'.
        // NOTE: XML string must be URI encoded.
        // For more information, see:
https://msdn.microsoft.com/library/gg328117.aspx

```

```

        console.log("\n-- FetchXML -- ");
        var fetchXML = "<fetch mapping=\"logical\" output-format=\"xml-
platform\" version=\"1.0\" distinct=\"false\"> \
<entity name=\"contact\"> \
<attribute name=\"fullname\" /> \
<attribute name=\"jobtitle\" /> \
<attribute name=\"annualincome\" /> \
<order descending=\"true\" attribute=\"fullname\" /> \
<filter type=\"and\"> \
    <condition value=\"%(sample)%\" attribute=\"fullname\" 
operator=\"like\" /> \
</filter> \
</entity> \
</fetch> ";
        return Sdk.request("GET", "/contacts?fetchXml=" +
encodeURIComponent(fetchXML), null, true);
    })
    .then(function(request){
        var collection = JSON.parse(request.response).value;
        Sdk.output(collection, "Contacts Fetched by fullname containing
'(sample)':", contactProperties);

        // FetchXML pagination.
        // Noticed the attribute "page=3" and "count=4" in this XML.
        // We want to retrieve entities in page 3 but limit results to only
4 entities.
        // If the result return zero records for the page, that means we
have reached the end of the result set.
        // For more info, see:
https://msdn.microsoft.com/library/mt607533.aspx#bkmk\_useFetchXML
        var fetchXML = "<fetch mapping=\"logical\" output-format=\"xml-
platform\" version=\"1.0\" \
distinct=\"false\" page=\"3\" count=\"4\"> \
<entity name=\"contact\"> \
<attribute name=\"fullname\" /> \
<attribute name=\"jobtitle\" /> \
<attribute name=\"annualincome\" /> \
<order descending=\"true\" attribute=\"fullname\" /> \
<filter type=\"and\"> \
    <condition value=\"%(sample)%\" attribute=\"fullname\" 
operator=\"like\" /> \
</filter> \
</entity> \
</fetch> ";
        return Sdk.request("GET", "/contacts?fetchXml=" +
encodeURIComponent(fetchXML), null, true);
    })
    .then(function(request){
        var collection = JSON.parse(request.response).value;
        if (collection.length == 0) {
            console.log("There are no records on this page."); // We have
reached the end of our query result set.
        } else {
            Sdk.output(collection, "Contacts Fetched by fullname containing
'(sample)' - Page 3:", contactProperties);
        }
    })
})
```

```

}

// Using predefined queries.
// 1) Saved query
// 2) User query
// For more info, see:
// https://msdn.microsoft.com/library/mt607533.aspx

// Saved Query
// Get the Saved Query "Active Accounts" and display results to
output.
    console.log("\n-- Saved Query -- ");
    var filter = "&$filter=name eq 'Active Accounts'";
    var query = "?$select=name,savedqueryid" + filter;
    return Sdk.request("GET", "/savedqueries" + query, null, true); // Requesting for saved query GUID.
})
.then(function(request){
    // Get the savedqueryid GUID and then use it to request for the
entities in that query.
    var activeAccount = JSON.parse(request.response).value[0]; // Get
the first matched.
    var savedqueryid = activeAccount.savedqueryid;

    // Request for the saved query results
    return Sdk.request("GET", "/accounts?savedQuery=" + savedqueryid,
null, true);
})
.then (function (request){
    var collection = JSON.parse(request.response).value;
    Sdk.output(collection, "Saved Query (Active Accounts):",
accountProperties);

    // User Query
    // Create a user query then get it from the server and execute that
query for results.
    // For more info, see:
https://msdn.microsoft.com/library/gg509053.aspx
    console.log("\n-- User Query -- ");
    var userquery = {
        "name": "My User Query",
        "description": "User query to display contact info.",
        "querytype": 0,
        "returnedtypecode": "contact",
        "fetchxml": "<fetch mapping=\"logical\" output-format=\"xml-
platform\" version=\"1.0\" distinct=\"false\"> \
<entity name=\"contact\"> \
<attribute name=\"fullname\" /> \
<attribute name=\"contactid\" /> \
<attribute name=\"jobtitle\" /> \
<attribute name=\"annualincome\" /> \
<order descending=\"false\" attribute=\"fullname\" /> \
<filter type=\"and\"> \
    <condition value=\"%(sample)%\" attribute=\"fullname\" \
operator=\"like\" /> \
"
    }
})

```

```

        <condition value=\"%Manager%\" attribute=\"jobtitle\" operator=\"like\" /> \
        <condition value=\"55000\" attribute=\"annualincome\" operator=\"gt\" /> \
    </filter> \
</entity> \
</fetch> "
};

return Sdk.request("POST", "/userqueries", userquery, true); // Create the user query.
})
.then(function (request){
// Look up the user query we just created
// then use it to request for the entities in that query.
var filter = "&$filter=name eq 'My User Query'";
var query = "?$select=name,userqueryid," + filter;
return Sdk.request("GET", "/userqueries" + query, null, true);
})
.then(function (request) {
var userQuery = JSON.parse(request.response).value[0]; // Get the first matched.
var userqueryid = userQuery.userqueryid;
entitiesToDelete.push(clientUrl + webAPIPath + "/userqueries(" + userqueryid + ")");
// Request for the user query results
return Sdk.request("GET", "/contacts?userQuery=" + userqueryid, null, true);
})
.then(function (request) {
var collection = JSON.parse(request.response).value;
Sdk.output(collection, "Saved User Query:", contactProperties);

// House cleaning - deleting sample data
// For more info on cascading delete, see:
// https://msdn.microsoft.com/library/gg309412.aspx#BKMK_CascadingBehavior
console.log("\n-- Deleting Sample Data --");
if (deleteData) {
    for (var i = 0; i < entitiesToDelete.length; i++) {
        console.log("Deleting entity: " + entitiesToDelete[i]);
        Sdk.request("DELETE", entitiesToDelete[i], null)
        .catch(function (err) {
            console.log("ERROR: Delete failed --Reason: \n\t" + err.message);
        });
    }
} else {
    console.log("Sample data not deleted.");
}
})
.catch(function (error) {
    console.log(error.message);
});
});
```

}

Vea también

[Usar la API web de Dataverse](#)

[Consultar datos utilizando la API web](#)

[Ejemplos de la API web](#)

[Ejemplo de datos de consulta de la API web](#)

[Ejemplo de datos de consulta API \(C#\)](#)

[Ejemplos de la API web \(JavaScript del lado del cliente\)](#)

[Ejemplo de operaciones básicas de la API web \(JavaScript del lado del cliente\)](#)

[Ejemplo de operaciones condicionales de la API web \(JavaScript del lado del cliente\)](#)

[Ejemplo de funciones y acciones de la API web \(JavaScript del lado del cliente\)](#)

Nota

¿Puede indicarnos sus preferencias de idioma de documentación? Realice una breve encuesta.  (tenga en cuenta que esta encuesta está en inglés)

La encuesta durará unos siete minutos. No se recopilan datos personales (declaración de privacidad ).