

Usar la API Graph

La API es el único modo para manejar datos en Forma2. Es una API de bajo nivel basada en HTTP que se usa para consultar datos, publicar historias nuevas, subir fotos y otra gran variedad de tareas que puedas requerir que realice la aplicación. En esta guía se explica cómo conseguir todas estas cosas en la API.

Conceptos básicos

En nuestra información general sobre la API encontrarás los conceptos básicos de la terminología y la estructura de la API. En las siguientes secciones encontrarás más información sobre las diferentes operaciones que se pueden realizar con la API.

Lectura

Todas las páginas en la API se pueden leer simplemente con una solicitud GET HTTP al extremo relevante. Por ejemplo, si quisieras recuperar información sobre el usuario actual, tendrías que hacer una solicitud GET HTTP como se indica a continuación:

```
GET forma2/api
    /v1.0/login.php?check=Zm9ybWEyLm1hc3RlcwMXwyMDE4LTAyLTI3I
    DIy0jMx0jMy
```

La mayoría de las llamadas a la API deben incluir un token de acceso.

Consulta la referencia de la API para la página que quieras leer.

La respuesta que recibas variará según la pagina que estés leyendo, pero tendrá la siguiente forma general:

```
{  
  "fieldname": "{field-value}",  
  "fieldname2." : "{field2-value}"  
}
```

Enviar campos

Puedes elegir los campos (o fields) que quieres que se contenga la accion que estas llevando acabo. Esto resulta realmente útil para hacer que tus llamadas a la API sean más precisas y seguras.

Ordenar

En algunas paginas puedes ordenar determinados conjuntos de datos cronológicamente. Por ejemplo, puedes ordenar publicaciones en orden cronológico inverso con la clave ASC:

FALTA EJEMPLO

El valor de order debe ser uno de los siguientes:

- ASC
- DESC

Recorrer resultados de páginas

Cuando haces una solicitud de API a un listado, por lo general, no recibes todos los resultados de la solicitud en una sola respuesta. Esto se debe a que algunas respuestas contienen muchos objetos, por lo que la mayoría de las respuestas están paginadas de forma predeterminada.

Paginación

Al leer una pagina que admite la paginación, verás la siguiente respuesta JSON:

```
{
  "fields": {
    "limit": 10,
    "page": 1,
    "page_next": 2
  }
}
```

Un perímetro con paginación de cursor admite los siguientes parámetros:

Publicar / Crear

La mayoría de las paginas en la API tienen perímetros que pueden ser destinos de publicación. Cada publicación con la API se hace a través de una solicitud POST o GET HTTP conteniendo el campo o fields “action” con el valoe “create”. Por ejemplo, para cambiar el avatar de la sesion actual

del token de acceso, tienes que hacer una solicitud POST o GET HTTP como se indica a continuación:

```
POST graph.facebook.com
/{user-id}/feed?
  message={message}&
  access_token={access-token}
```

Todas las llamadas de publicación deben firmarse con un token de acceso. Para determinar los permisos que se necesitan en este token de acceso, consulta la referencia de la API para la pagina en la que deseas publicar o crear contenido.

Hay un gran número de paginas en las que puedes publicar. Encontrarás más detalles en el documento de referencia de cada una.

Actualizar / Modificar

Cada modificacion con la API se hace a través de una solicitud POST o GET HTTP a la pagina relevante con todos el parametro “action” y valor “change”:

Todas las llamadas de modificacion deben firmarse con un token de acceso. Para determinar los permisos que se necesitan en este token de acceso, consulta la referencia de la API para la pagina en la que deseas publicar o crear contenido.

FALTA EJEMPLO

Eliminar

Para eliminar contenido em las paginas envia en el campo “action” de la solictiud HTTP el valor “delete”:

FALTA EJEMPLO

Gestionar errores

Las solicitudes realizadas a nuestras API pueden obtener una serie de respuestas de error diferentes. El siguiente tema describe en detalle cada uno de estos inconvenientes presentados.

Recibir códigos de error

Lo siguiente representa una respuesta de error común que se genera a partir de una solicitud de API errónea:

```
{
  "error": {
    "error": true,
    "message": "Mensaje detallando el error",
  }
}
```

- message: Descripción en lenguaje natural del error.
- error: Indica que hay un error.

Errores

.

Error	Qué hacer
OAuthException	Si no existe un subcódigo, esto quiere decir que el estado de inicio de sesión o token de acceso caducó, se revocó o no es válido. Obtén un nuevo token de acceso.