

Youtube API

Recurso: Entidad de datos individual con un identificador exclusivo

En nuestro caso, vamos a usar principalmente dos recursos:

Search Result: Un resultado de búsqueda contiene información acerca de un vídeo, lista de reproducción ... que coincide con los parámetros de búsqueda, un resultado de búsqueda apunta a un recurso con una única identificación (la búsqueda carece de persistencia)

Video: Representa a un video en youtube (tiene sus propias propiedades anidadas, de las cuales, usaremos *Player*)

Operaciones Admitidas: (Principales CRUD)

List: Recupera (Get) una lista de cero o más recursos.

Insert: Escribe (Post) un recurso.

Update: Actualiza un recurso.

Delete: Borra un recurso.

	C	R	U	D
Search Result	<i>NO</i>	<i>SÍ</i>	<i>NO</i>	<i>NO</i>
Video	<i>SÍ</i>	<i>SÍ</i>	<i>SÍ</i>	<i>SÍ</i>

Recursos Parciales

Evitan transparencia, almacenamiento y análisis innecesarios en pos de la eficiencia.

Parámetro Part

Obligatorio para cualquier solicitud que recupere o muestre recursos.

Identifica una o más propiedades de recursos de nivel superior (no anidadas) que se debe incluir en la respuesta de la API (cada recurso tiene los suyos)

El parámetro requiere que selecciones los componentes de recursos que tu aplicación realmente usa.

Parámetro Fields

Filtra la respuesta de la API, que solo contiene las *part* de los recursos identificados en el valor del parámetro *part*, de modo, que la respuesta solo incluye un conjunto específico de campos.

El parámetro *fields* permite eliminar propiedades anidadas de una respuesta de la API para reducir aún más el ancho de banda.