

Events API

The Events API allows you to ingest JSON events with a simple HTTP POST request. [Read more about the Events API.](#)

All endpoints require authentication using an Auth Token with `DATASOURCE: APPEND` or `DATASOURCE: CREATE` scope.

POST /v0/events


Use this endpoint to send NDJSON (new-line delimited JSON) events to a [Data Source](#).

Request parameters

KEY	TYPE	DESCRIPTION
name	String	name or ID of the target Data Source to append data to it
wait	Boolean	'false' by default. Set to 'true' to wait until the write is acknowledged by the database. Enablin this flag makes possible to retry on database errors, but it introduces additional latency. It's recommended to enable it in use cases in which data loss avoidance is critical. It's recommended to disable it otherwise.

Return HTTP status codes

STATUS CODE	DESCRIPTION
200	The data has been inserted into the database. The write has been acknowledged. The request 'wait' parameter was enabled.
202	The data has been processed, and it will be send to the database eventually. The write has not been acknowledged yet. The request 'wait' parameter was disabled.
400	The request is invalid. The body will contain more information. A common cause is missing the 'name' parameter. No data has been inserted, but the request shouldn't be retried.



STATUS CODE	DESCRIPTION
403	The token is not valid. The request shouldn't be retried.
404	The token's workspace doesn't belong to this cluster. The workspace is probably removed or in another cluster. The request shouldn't be retried, ensure the token's region and the Tinybird domain matches.
422	The ingestion has been partially completed due to an error in a Materialized View. Retrying may result in data duplication, but not retrying may result in data loss. The general advice is to not retry, review attached Materialized Views, and contact us if the issue persists.
429	The request/second limit has been reached. Default limit is 1000 requests/second, contact us for increased capacity. The request may be retried after awhile, we recommended using exponential backoff with a limited amount of retries.
500	An unexpected error has occurred. The body will contain more information. Retrying is the general advice, contact with us if the issue persists.
503	The service is temporarily unavailable. The body may contain more information. A common cause is to have reached a throughput limit, or to have attached a Materialized View with an issue. No data has been inserted, and it's safe to retry. Contact with us if the issue persists.
0×07 GOAWAY	HTTP2 only. Too many alive connections. Recreate the connection and retry.

Compression

You can compress JSON events with Gzip and send the compressed payload to the Events API. You must include the header `Content-Encoding: gzip` with the request.

Examples

SEND INDIVIDUAL JSON MESSAGES



```
curl \
  -H "Authorization: Bearer <import_token>" \
  -d '{"date": "2020-04-05 00:05:38", "city": "Chicago"}' \
  'https://api.tinybird.co/v0/events?name=events_test'
```

SEND MANY NDJSON EVENTS. NOTICE THE '\$' BEFORE THE JSON EVENTS. IT'S NEEDED IN ORDER FOR BASH TO REPLACE THE '\N'. CURL DOESN'T DO IT AUTOMATICALLY.



```
curl \
  -H "Authorization: Bearer <import_token>" \
  -d $'{"date": "2020-04-05 00:05:38", "city": "Chicago"}\n{"date": "2020-
'https://api.tinybird.co/v0/events?name=events_test'
```

SEND A GZIP COMPRESSED PAYLOAD, WHERE 'BODY.GZ' IS A BATCH OF NDJSON EVENTS

```
curl \
  -H "Authorization: Bearer <import_token>" \
  -H "Content-Encoding: gzip" \
  --data-binary @body.gz \
  'https://api.tinybird.co/v0/events?name=events_example'
```

Company

- Product
- Pricing
- ROI Calculator
- About Us
- Shop
- Careers
- Request a demo

Resources

- Docs
- Blog
- Community
- Live Coding
- Customer Stories
- RSS Feed

Integrations

- Amazon S3

Use cases

- In-Product Analytics

Kafka Data Streams

Operational Intelligence

Google Cloud Storage

Realtime Personalization

Google BigQuery

Anomaly Detection & Alerts

Snowflake

Usage Based Pricing

Confluent

Sports Betting/Gaming

Smart Inventory Management

Serverless ClickHouse

Copyright © 2024 Tinybird. All rights reserved

[Terms & conditions](#) [Cookies](#) [Security](#)

Spain
Calle del Dr. Fourquet, 27
28012 Madrid

USA
41 East 11th Street 11th floor
New York, NY 10003