



## What is a Release?

Releases are [semantically-versioned](#) iterations of your Tinybird project (including [Data Sources](#), [Pipes](#), [API Endpoints](#)). A Release marks a specific point in your projects history that is tied to a Git commit - just like creating a [git tag](#) allows you to create a tagged release on a provider like GitHub.

For a full explanation of how semantic versioning works in a Tinybird Workspace, read the [SemVer deployment behavior docs](#).

## What should I use Releases for?

Use Releases to control the release lifecycle of your changes. This includes:

- Releasing previews to share and validate your changes.
- Promoting validated changes into production.
- Rolling back to a previous version.

Releases should be used when development is nearing completion, and you are ready to begin validating and preparing for production. Releases should not be used for active development, instead, use [Branches](#).

## Limits and prerequisites

Releases can be created, deleted, promoted, and rolled back. You can have a maximum of **1 Preview, 1 Live, and 3 Rollback** Releases at any time.

Your Workspace must be connected to your remote Git provider (for example, GitHub) - see [Connecting your Workspace](#).

## Data in Releases

**All Releases share production data.** The Preview and Rollback Releases reference the production data of your Live Release.



This means that any data operation (such as deletes, truncates, inserts, etc.) performed in any Release, will affect all other Releases as well.

Data in Releases behaves differently than data in Branches, which allow you to work on segregated data that does not impact production. [Read more about Branches](#).

## Creating Releases

Releases can only be created via a Pull Request from your Git provider.

To create a new Release, push your local Git branch to your Git provider and open a new PR. The automated CI/CD actions will sync the branch to Tinybird, and create a preview Release in Tinybird once the PR is merged.

The structure of the version number is critical as it dictates the behavior of the Release deployment. For more information, see the [SemVer deployment behavior](#) docs

## Deleting Releases

There are two ways to delete a Release: via the Tinybird UI or as a Pull Request from your Git provider.

### Deleting a Release with the Tinybird UI

To delete a Release with the Tinybird UI, click the **Release** dropdown in the top bar, and click on the **Release** tab. Find your Release in the list, and click the trash button to the right.

### Deleting a Release from a Pull Request

When a PR is raised, the Release will be created in Tinybird. When the PR is closed, the Release will be deleted from Tinybird.

## Promoting Releases

This can be done via the Tinybird UI or via the CLI, as well as via Tinybird's default GitHub Actions.

### Promote a Release with the Tinybird UI

To promote a Release using the UI, navigate to the Release tab in the Tinybird UI. Find the Release you want to promote, and click the **...** actions button, then click **Promote**. You will be

prompted to review the Releases in scope (Live Release vs. Preview Release). Confirm the promotion with the `Promote x.x.x` button

## Promote a Release with the Tinybird CLI

To promote a Release using the CLI, run `tb release promote --semver <Release SemVer, e.g. 1.0.0>`.

## Rolling back Releases

This can be done via the Tinybird UI or via the CLI, as well as via Tinybird's default GitHub Actions.

### Rolling back a Release with the UI

To roll back a Release using the UI, navigate to the Release tab in the Tinybird UI. Find the Release you want to roll back, and click the `...` actions button, then click `Rollback`. You will be prompted to review the Releases in scope (current Release vs. rollback Release). Confirm the rollback with the `Rollback to x.x.x` button.

### Rolling back a Release with the CLI

To roll back a Release using the CLI, run `tb release rollback`. The output will show metadata of current Live Release, and the Rollback Release that will be promoted in its place.

Confirm the rollback by pressing `y`.

### Rolling back a Release with GitHub Actions

Tinybird provides you with a [set of .yaml actions](#) that are configurable if required (though this shouldn't be necessary for many use cases). In the GitHub UI, select the `rollback` job:

Go to your GitHub repository, and click on the `Actions` tab. Find the `Releases Workflow`, click on the `Run workflow` button. In the `Select a job to run manually` option, type `rollback` and then click the `Run workflow` button.

Company

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

Integrations

- Amazon S3
- Kafka Data Streams
- Google Cloud Storage
- Google BigQuery
- Snowflake
- Confluent

Resources

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

Use cases

- In-Product Analytics
- Operational Intelligence
- Realtime Personalization
- Anomaly Detection & Alerts
- Usage Based Pricing
- Sports Betting/Gaming
- Smart Inventory Management
- Serverless ClickHouse

Spain  
Calle del Dr. Fourquet, 27  
28012 Madrid

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