



## Instructions: Migrating to KURRENTDB cloud

### Overview

KurrentDB can run in the cloud. This option is a popular choice among our customers because the infrastructure is managed by Kurrent. Availability in multiple regions is built in, and processes such as scheduled backups are easy to configure.

If you register for a cloud account and create a KurrentDB cluster, you could follow these steps and run the code in github codespaces against a KurrentDB cloud cluster.

Information on Kurrent Cloud is available at <https://kurrent.io/Kurrent-cloud>

### Step 1: Create a Cluster with Remote Access Enabled

Kurrent Cloud allows the creation of both dedicated and shared clusters.

Note that "shared" in this case means, "running on our kubernetes infrastructure". Your cluster is isolated and can only be used by authorized accounts in your organization.

A screenshot of the "New Cluster" form in the Kurrent Cloud interface. The form has a title "New Cluster" in a purple font. Below the title is a "Cluster name" field with a placeholder "Please enter a cluster name". To the right of this field is a red rectangular box containing the text "Create a shared cluster" with an arrow pointing to the "Shared" option in the "Infrastructure Type" section. The "Infrastructure Type" section has two options: "Shared" and "Dedicated". The "Shared" option is highlighted with a red oval and a "PREVIEW" label in a purple box. The "Dedicated" option is also labeled with a "PREVIEW" label in a purple box. To the right of the "Infrastructure Type" section is a light blue box titled "What is Shared Infrastructure?". It contains the text: "With this option, clusters are hosted in shared compute infrastructure. Each cluster is isolated from other deployments, making them **highly secure** and **cost effective**." Below this text is a "PREVIEW" label in a purple box and a note: "Please note that this feature is in preview and not suitable for production workloads. Scheduled backups and integration features will be available in a future release."

### Public Access Security Notes

This cluster is secured in the following ways.

A unique username and password for the administrator account for that cluster is generated when the cluster is created.

A list of IP addresses that can access the cluster must be created, the default access list is empty.

## Step 2: Determine your codespace IP address

GitHub codespaces are docker containers running on private networks in GitHub's datacenters.

Web access for the codespace web interface is routed through to the internal docker containers. Running a command like `ipconfig`, that would work in most instances will not help us when ran in codespaces, as it will return the local, non publicly routable address.

A remote request from codespaces to an external resource, such as a website, will expose the currently used public IP address.

A shell script `find_my_IP.sh` is included.

Run that script to return the IP address currently used by your codespace container.

:warning: Your codespace IP will change, if there is no activity for a long period of time (tens of minutes), or if the codespace is stopped and restarted. This IP is assigned out of a pool of addresses by github, the IP address your codespace has today, may be in use by some other codespace tomorrow. You should remove the IP address from your cluster's access list when you terminate your codespace.

In a terminal enter the command. `./find_my_IP.sh`

Your IP address will be displayed to the terminal, `Your IP address is 11.111.111.115 enter this in the access list on Kurrent Cloud`

Your IP address will also be written to the file `my_IP.txt`

## Step 3: Enter your IP address into the Kurrent Cloud clusters IP access list

eventstore\_training / Projects / apex\_test / Public IP Access lists / Kurrent\_DB\_Cloud\_Access\_List

## Edit IP access list

IP access list ID

cv0br4jtv1lmm9o71p8g



IP Access list name

Kurrent\_DB\_Cloud\_Access\_List

Addresses

76.189.112.124/32

Github Codespace IP

Add my IP address  Add Address 



**Add your codespace IP address here**


## Step 4: Use the Kurrent Cloud Console to Generate a Connection String

Click the [connect to <cluster\\_name>](#) button.


eventstore\_training / Projects / apex\_test / Managed ESDB / Clusters


## Clusters

 New Cluster 

Name	Kind	Cluster ID	Provider	Region	Type	Version	Status	Created	Actions
second_try_Feb_20	Shared								<a href="#">Connect to second_try_Feb_20</a> 

**Details** Infrastructure Database Security Addresses

**Name:** second\_try\_Feb\_20 **Status:**  Available

**Cluster ID:** curr4icgdub1aodg1sh0  **Created:** 2/20/2025

**Mutual TLS:** Disabled

**Click Here**


Choose your language, in this case python.

☒


☒

☐


ConnectionChooseConnect




Connect with the Dotnet client




Go  
Connect with the go client




Java  
Connect with the Java client



Node.js  
Connect with the Node.js client



Rust  
Connect with the Rust client



Python  
Connect with the Python client

Other

Choose Python

Copy the generated connection string.

✓

ConnectionChooseConnect

## Required Packages

Install the client SDK package to your project.

**PyPI** **Poetry**

```
pip install esdbclient
```

**Create a Client**

First things first, we need a client.

```
client = EventStoreDBClient(  
    uri="esdb+discover://admin:6906b7791d76454bb4fae6faf590ab06@kurrentdb.  
)
```

The client instance can be used as a singleton across the whole application. It doesn't need to open or close the connection.

Go to documentation

**warning** This connection string includes the admin password. Do not share this in public forums, or public github repositories.

## Step 5:

Comment out the line

```
client = EventStoreDBClient(uri="esdb://localhost:2113?tls=false")
```

And add the line copied in step 4

```
client = EventStoreDBClient(  
    uri="esdb+discover://admin:6906b7791d76454bb4fae6faf590ab06@kurrentdb.curr4icgdub1aod
```

```
)
```

## Step 6: Append an event to verify

```
python sample_append.py
```

**Congratulations you have modified the client connection string to connect to a Kurrent Cloud cluster !!**