

# Connecting with Node.js

This guide walks you through the process of connecting a Node.js application to a Hydra database using the `pg` package. You'll learn how to set up the environment, configure the connection, and run a simple SQL query.

## Variables

To connect to a Hydra database, the following parameters are required. You can find these details in the **Elestio service overview page** of your Hydra service.

Variable	Description	Purpose
<code>USER</code>	Hydra (PostgreSQL) username	Identifies the database user with access privileges
<code>PASSWORD</code>	Hydra password	Authenticates the user against the Hydra database
<code>HOST</code>	Hostname of the Hydra instance	Specifies the server address of the database
<code>PORT</code>	Port for Hydra (usually 5432)	Specifies the network port for connections
<code>DATABASE</code>	Name of the Hydra database	Specifies which database to access

These values can usually be found in the Elestio service overview details as shown in the image below, make sure to take a copy of these details and add it to the code moving ahead.

hydra-y2e0a

Hydra Cluster Running

Open terminal Delete cluster Add node

Overview Nodes Backups Audit

Termination protection Disabled. VM can be powered off and terminated. Protection deactivated

Auto-Failover Enabled. In case of failure, the cluster will automatically attempt to recover. Auto-Failover activated

Node 1 Primary Node

Database Admin Display your database credentials

Host	hydra-y2e0a-u7774.vm.elestialo.app	<input type="button" value="Copy"/>
Port	15432	<input type="button" value="Copy"/>
User	postgres	<input type="button" value="Copy"/>
Password	*****	Show password <input type="button" value="Copy"/>
CLI	PGPASSWORD=***** psql --host=hydra-y2e0a-u7774.vm.elestialo.app --port=15432 --username=postgres	Show password <input type="button" value="Copy"/>

## Prerequisites

- **Install Node.js and NPM**
  - Check if Node.js is installed:

```
node -v
npm -v
```

- If not, download and install it from <https://nodejs.org>.
- **Install the pg Package**
  - Hydra is PostgreSQL-compatible, so use the pg package:

```
npm install pg --save
```

## Code

Once all prerequisites are set up, create a new file named `hydra.js` and add the following code.

```
const { Client } = require("pg");

// Database connection configuration
const config = {
  host: "HOST",
  user: "USER",
  password: "PASSWORD",
  database: "DATABASE",
  port: PORT,
  ssl: {
    rejectUnauthorized: false, // Only if Hydra requires SSL (check Elestio settings)
  },
};

// Create a new client instance
const client = new Client(config);

// Connect to the Hydra database
client.connect((err) => {
  if (err) {
    console.error("Connection failed:", err.stack);
    return;
  }

  console.log("Connected to Hydra");

  // Run a test query
  client.query("SELECT version()", (err, res) => {
    if (err) {
      console.error("Query failed:", err.stack);
    } else {
      console.log("Hydra/PostgreSQL Version:", res.rows[0].version);
    }

    // Close the connection
    client.end((err) => {
      if (err) console.error("Error closing connection:", err.stack);
    });
  });
});
```

To execute the script, open the terminal or command prompt and navigate to the directory where `hydra.js`. Once in the correct directory, run the script with the command

```
node hydra.js
```

If successful, you'll see:

```
Connected to Hydra  
Hydra/PostgreSQL Version: PostgreSQL 14.13 (Debian 14.13-1.pgdg120+1) on x86_64-pc-linux-gnu,  
compiled by gcc (Debian 12.2.0-14) 12.2.0, 64-bit
```

---

Revision #3

Created 2025-05-07 07:21:12 UTC by kaiwalya

Updated 2025-05-07 15:21:41 UTC by kaiwalya