

Connecting with Java


This guide shows how to connect your Java app to a **TimescaleDB database** using the [PostgreSQL JDBC driver](#), parse command-line arguments, and run a basic query.

Variables

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

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


 **timescaledb-gi7jy**

TimescaleDB

Cluster

Running

>_ Open terminal


 Delete cluster

Add node

OverviewNodesBackupsAudit


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

Hide DB Credentials

| | | |
|----------|---|--|
| Host | timescaledb-gi7jy-u7774.vm.elestialio.app |  |
| Port | 25432 |  |
| User | postgres |  |
| Password | ***** | Show password  |
| CLI | PGPASSWORD=***** psql --host=timescaledb-gi7jy-u7774.vm.elestialio.app --port=25432 --username=postgres | Show password  |

Prerequisites

Install Java & JDBC driver

Check if Java is installed by running:

```
java -version
```

If not installed, install it first and then download and install **JDBC** driver from <https://jdbc.postgresql.org/download/> or if you have Maven installed, run the following command with updated version of the driver:

```
mvn org.apache.maven.plugins:maven-dependency-plugin:2.8:get \
-Dartifact=org.postgresql:postgresql:42.7.5:jar \
-Ddest=postgresql-42.7.5.jar
```

Code

Once all prerequisites are set up, create a new file named `TDB.java` and add the following code:

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.HashMap;
import java.util.Map;

public class TDB {

    static class Config {
        String host, port, database, username, password;

        Config(String host, String port, String database, String username, String password) {
            this.host = host;
            this.port = port;
            this.database = database;
            this.username = username;
            this.password = password;
        }

        String getJdbcUrl() {
            return String.format("jdbc:postgresql://%s:%s/%s?sslmode=require", host, port,
database);
        }

        boolean isComplete() {
            return host != null && port != null && database != null && username != null &&
password != null;
        }
    }

    static Map<String, String> parseArgs(String[] args) {
        Map<String, String> map = new HashMap<>();
        for (int i = 0; i < args.length - 1; i += 2) {
            map.put(args[i], args[i + 1]);
        }
        return map;
    }
}
```

```

}

public static void main(String[] args) {
    try {
        Class.forName("org.postgresql.Driver");

        Map<String, String> argMap = parseArgs(args);
        Config cfg = new Config(
            argMap.get("-host"),
            argMap.get("-port"),
            argMap.get("-database"),
            argMap.get("-username"),
            argMap.get("-password")
        );

        if (!cfg.isComplete()) {
            System.err.println("Missing required arguments. Example usage:");
            System.err.println("java -cp postgresql-42.7.5.jar:. TDB -host <HOST> -port  

<PORT> -database <DB> -username <USER> -password <PASS>");
            return;
        }

        try (Connection conn = DriverManager.getConnection(cfg.getJdbcUrl(), cfg.username,
cfg.password)) {
            System.out.println("Connected to TimescaleDB database successfully.");

            Statement stmt = conn.createStatement();
            ResultSet rs = stmt.executeQuery("SELECT current_database(), current_user,  

version()");

            while (rs.next()) {
                System.out.println("Database: " + rs.getString(1));
                System.out.println("User: " + rs.getString(2));
                System.out.println("Version: " + rs.getString(3));
            }

            rs.close();
            stmt.close();
        }
    }
}

```

```
    } catch (ClassNotFoundException e) {  
        System.err.println("PostgreSQL JDBC driver not found.");  
        e.printStackTrace();  
    } catch (SQLException e) {  
        System.err.println("Connection or query error:");  
        e.printStackTrace();  
    }  
}  
}
```

To execute the script, open the terminal or command prompt and navigate to the directory where `TDB.java`. Once in the correct directory, run the script with the command (Update the variables with actual values acquired from previous steps).

```
javac TDB.java
```

```
java -cp postgresql-42.7.5.jar:. TDB -host HOST -port PORT -database DATABASE -username  
USERNAME -password PASSWORD
```

If the connection is successful, the terminal will display output similar to:

```
Connected to TimescaleDB database successfully.  
Database: elestio  
User: postgres  
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 #2

Created 13 May 2025 07:35:13 by kaiwalya

Updated 13 May 2025 07:37:43 by kaiwalya