Discord Bot with MySQL Integration Documentation

# Introduction

This Python script showcases the implementation of a Discord bot using the `discord.py` library. The bot integrates with a MySQL database to store authentication tokens for each server it joins. These authentication tokens are crucial for identifying and authenticating servers.

# Prerequisites

Before running the script, ensure you have the following dependencies installed:

| pip **install** mysql **discord** mysql-connector-python |
| --- |

# Configuration

## Discord Bot Token:

| bot.**run**('MTE4NDQ0NjM3MzY4NTM3NDk3Ng.G2XcF0.OXYQvy8RX6qtus7XEGu0M5fER3zf6-DBVjf5LE') |
| --- |

## - MySQL Database Configuration:

| db\_config = {  "host": "sql12.freesqldatabase.com",  "user": "sql12670091",  "password": "VCI1TgY3Zx",  "database": "sql12670091",  } |
| --- |

# Bot Functionality

## Authentication Token Generation

* When the bot joins a new server (`on\_guild\_join` event), it generates a unique authentication token using `secrets.token\_urlsafe(16)` and inserts the server ID and token into the MySQL database.

| **@bot.event** **async** **def** on\_guild\_join(guild):  # Generate a unique auth token for the new server  auth\_token = secrets.token\_urlsafe(16) # Change the length as needed   # Insert the server ID and auth token into the database  insert\_query = "INSERT INTO auth\_tokens (server\_id, auth\_token) VALUES (%s, %s)"  data = (guild.id, auth\_token)  cursor.execute(insert\_query, data)  db.commit() |
| --- |

## Hello Command

* The bot responds to the `g!hello` command by printing "Hello World" along with the server name. It checks if the authentication token for the server exists in the database.

| **@bot.command(name='hello', help='Print "Hello World" along with the server name')** **async** **def** hello(ctx):  server\_id = ctx.guild.id  auth\_token = get\_auth\_token(server\_id)  **if** auth\_token:  **await** ctx.send(f'Hello World {ctx.guild.name}!')  **else**:  **await** ctx.send('Authentication token not found. Please add the bot to the server again.') |
| --- |

## Get Auth Token Function

* The `get\_auth\_token` function retrieves the authentication token for a given server ID from the MySQL database.

| **def** get\_auth\_token(server\_id):  cursor.execute("SELECT auth\_token FROM auth\_tokens WHERE server\_id = %s", (server\_id,))  result = cursor.fetchone()  **return** result[0] **if** result **else** **None** |
| --- |

# Bot Execution

1. Run the bot using the following command:

| **python** your\_script\_name.**py** |
| --- |

1. Invite the bot to the Discord server:

Use the following link to invite the bot: <https://discord.com/api/oauth2/authorize?client_id=1184446373685374976&permissions=2147486720&scope=bot>

## Notes:

* Run the script before inviting the bot
* Ensure your bot is invited to the server and has the necessary permissions. Use the `g!hello` command to test the bot's functionality.

# Test

