

DATABASE TABLES

[database dev-guide]

1. Table Name: 'guilds'

guild_id	join_date

Table Relationships:

[guild_id] (PRIMARY Key)

Uses Indexing to ensure that guild_id exists where any command is being called.

TERMINAL COMMAND:

```
CREATE TABLE guilds (  
  guild_id BIGINT PRIMARY KEY,  
  join_date TIMESTAMP);
```

2. Table Name: 'group_categories'

guild_id	category_id	category_name	post_channel_id	emoji	admin_msg

Table Relationships:

[guild_id] (Foreign Key)

Ensures every 'group_category' is associated with an existing server.

[guild_id, category_id] (Composite Primary Key)

Ensures within a specific server, duplicate group_categories cannot exist.

POSTGRESQL COMMAND:

```
CREATE TABLE group_categories (  
  guild_id BIGINT REFERENCES guilds(guild_id),  
  category_id BIGINT NOT NULL,  
  category_name TEXT NOT NULL,
```

```
post_channel_id BIGINT NOT NULL,  
emoji TEXT DEFAULT '👍',  
admin_msg TEXT DEFAULT '',  
PRIMARY KEY (guild_id, category_id));
```

3. Table Name: 'groups'

guild_id	category_id	group_owner_id	Post_id	text_channel_id	Voice_channel_id	Created_date

Table Relationships:

[guild_id] (Foreign Key)

Ensures every group_category created is for an existing server.

[guild_id, category_id] (Composite Primary Key)

Ensures that a category_id is defined for that server. Also links to 'group_categories' Table giving access to details like [category_name], [post_channel_id], [emoji], and [std_admin_msg] when user forms a group.

[text_channel_id] (Foreign Key)

Links to 'text_channel_id' in the 'groups' table, indicating the variables associated with type for deletion process.

POSTGRESQL COMMAND:

```
CREATE TABLE groups (  
guild_id BIGINT REFERENCES guilds(guild_id),  
category_id BIGINT NOT NULL,  
group_owner_id BIGINT NOT NULL,  
post_id BIGINT NOT NULL,  
text_channel_id BIGINT UNIQUE NOT NULL,  
voice_channel_id BIGINT NOT NULL,  
created_date TIMESTAMP NOT NULL,  
FOREIGN KEY (guild_id, category_id) REFERENCES  
group_categories(guild_id, category_id));
```