**Guide to Setting Up and Using Supabase**

**Introduction**

Supabase is an open-source Backend-as-a-Service (BaaS) platform that provides features like real-time databases, authentication, and serverless functions. It simplifies backend development while offering robust integrations for modern applications.

**Prerequisites**

1. Basic knowledge of programming.
2. A web browser.
3. An account with [Supabase](https://supabase.com/).
4. A code editor like VS Code.

**Setting Up Supabase**

**Step 1: Create a Supabase Account**

1. Visit the [Supabase website](https://supabase.com/) and click on **Sign Up**.
2. Use your email or GitHub account to create an account.

**Step 2: Create a New Project**

1. After logging in, go to the **Dashboard**: <https://supabase.com/dashboard/projects>
2. Click **New Project**.
3. Setup your **Organization** (just leave the default values and click **Create organization**).
4. Fill in the project details:
   * **Project Name**: A unique name for your project.
   * **Database Password**: Choose a strong password (**yourself** or click **Generate a password** and copy it to note it down).
   * **Region**: Select **Central Europe (Zurich)** or **Central EU (Frankfurt)**.
5. Click **Create New Project**.

**Step 3: Wait for the Setup** to finish.

**Using Supabase**

**Step 4: Database**

1. Navigate to the **Table Editor** tab.

A screenshot of a computer

Description automatically generated



1. Click **Create a new table**.
2. Fill in the **name** and the **columns** (specify for each column the type of data you want to store).
3. Click **Save**.
4. Example: To create a "users" table:

A white background with black text

Description automatically generatedA screenshot of a computer

Description automatically generated

A close-up of a white background

Description automatically generatedA screenshot of a computer

Description automatically generated



**Step 5: Access from Python**

1. Navigate to the **Home** tab.

A screenshot of a computer

Description automatically generated



1. Scroll down to find your **Project URL** and **API Key (click the yellow marked text to find the SERVICE\_ROLE SECRET key and use this one)**

A screenshot of a computer

Description automatically generated



A screenshot of a chat

Description automatically generatedA screenshot of a chat

Description automatically generated

1. Install the library using pip in your cmd/terminal: **pip install supabase**
2. Initializing Supabase in Python:

from supabase import create\_client, Client

# Replace these with your Supabase project details

SUPABASE\_URL = "https://<your-project-ref>.supabase.co"

SUPABASE\_KEY = "<your-supabase-key>"

supabase: Client = create\_client(SUPABASE\_URL, SUPABASE\_KEY)

1. Write data to Supabase:

def insert\_user(username: str, password: str):

try:

response = supabase.table("users").insert({"username": username, "password": password}).execute()

return response.data

except Exception as e:

return e

insert\_user("john\_doe\_5", "secure\_password123")

1. Read data from Supabase:

*Getting all users*

def get\_all\_users():

try:

response = supabase.table("users").select("\*").execute()

return response.data

except Exception as e:

return e

get\_all\_users()

*Getting specific user*

def get\_user\_by\_username(username: str):

try:

response = supabase.table("users").select("\*").eq("username", username).execute()

return response.data

except Exception as e:

return e

get\_user\_by\_username("john\_doe")

1. Update data to Supabase:

def update\_password(username: str, new\_password: str):

try:

response = supabase.table("users").update({"password": new\_password}).eq("username", username).execute()

return response.data

except Exception as e:

return e

update\_password("john\_doe", "new\_secure\_password456")

1. Delete data from Supabase:

def delete\_user(username: str):

try:

response = supabase.table("users").delete().eq("username", username).execute()

return response.data

except Exception as e:

return e

delete\_user("john\_doe")

**Step 6: View your Data**

You can always get a live view of your table data when going back to the table editor (Step 4) and select the table you are interested in.