

Intro to APIs

API זה ראשי תיבות של **Application Programming Interface**. במנוחה זה משתמשים בעולמות התכנות והטכנולוגיה על מנת לתאר דרך או גישה למידע של שירות חיצוני, בדרך כלל באמצעות קוד.

ambil להכנס יותר מדי פרטים טכניים, API זה מעין "גישה" או דרך לקבל מידע שאתר או שירות אחר מחזיקם. מי שמספק את ה-API זה מי שמחזיק באותו מידע (ומעוניין לשთף אותו עם אחרים) וכי משתמש ב-API שלו מי שרצה לקבל אותו מידע. לדוגמה, אתם מפתחים אפליקציה לניהול זמנים ורוצים להתחבר ללוח שנה של המשתמשים בגוגל. גוגל חושפת API שיאפשר לכם לגשת לאותו לוח שנה של המשתמש.

בקצהו: API זו דרך של חברות או שירותי אינטרנט מסוימים לחשוף ולשתף מידע שהם מחזיקים לגורמי צד שלישי בצורה מסודרת ו邏邏. לדוגמה, כשאומרים משפט כמו "יש אפשרות לעבוד עם ה-API של פייסבוק" מתכוונים שפייסבוק מחזינה בצורה מסודרת דרך לגשת מידע שהוא מחזיקה ולבנות עליו פועלות כגון - לקבל את הפרטים האישיים של המשתמש, לשנות את ההופעה באתר, לפרסם פוסטים בשם זה וכו'.

האפליקציה הקצתה שלכם



כל האירועים בלוח שנה
של המשתמש



מחזיר תשובה
עם האירועים



Google Calendar



למה צריך API?

כדי להתמסך לשירותים אחרים - לפעמים אין ברירה ובסبيل לגשת למידע מסוים אתם חייבים גישה מסודרת אליו. לדוגמה, אתם יצרתם אוטומציה ששםה אף של ליצן לתמונות פרופיל של אנשים בפייסבוק. על מנת לגשת למידע של המשתמשים בפייסבוק (התמונה פרופיל שלהם) ולשנות אותו, תהיו חייבים לעבוד עם ה-API של פייסבוק שמציע דרך מוסדרת וمبוקרת לשנות פרטיים של המשתמשים.

לפעמים תגלו שלא קיים API לצורך שלכם מאחר שאו שהחברה התעכלה או לא ראה צורך לפתח API למשהו מסוים, או שהם פשוט לא רצים שירותים צד שלישי כמוכם יכולים יכולו לגשת למידע זהה (מטרומי פרטיות וכו').

איך עובדים עם API ומה זה REST API

לרוב נדרש עבדות קוד או אוטומציה על מנת לגשת ל-API של שירות מסוים. בדרך כלל מדובר בכתובות אינטרנט שאליה מכרפים כל מיני פרמטרים מסוימים על מנת לקבל את המידע המבוקש.

רבה APIs עובדים עם מתודולוגיה שנקראת REST API שזה בעצם מוסכמה על צורת העבודה עם API - כמו שפה משותפת, סטנדרט מסוים, כדי שמהר תلقו ל-API של חברת אחרת, הצורת העבודה תראה מאוד דומה.

מצגת APIs

Example API

<https://restful-api.dev/>

API Vault

<https://apivault.dev/>

משימה!

הצילו את דני

Advanced APIs (headers, Auth tokens) & Webhooks

מצגת

Webhook node

Test with: <https://reqbin.com/>

N8N Webhooks examples

<https://n8n.io/workflows/5171-learn-api-fundamentals-with-an-interactive-hands-on-tutorial-workflow/>

חלוקת השיעור היום

שעה 3-1: הרצאה על בוט טלגרם, התחברות לטלגרם בעצמכם ומשימה
שעה 4: שעת קבלה - שעה חופשית לעבודה, יישור קו ושאלות מכל סוג

Create Telegram bot

1. Sign up for Telegram at <https://web.telegram.org>
2. Open BotFather in Telegram
3. Send /newbot, provide a display name (e.g., "My Test Bot"), and a unique username ending in "bot" (e.g., "mytestbot_bot"). BotFather provides an API token like **123456789:AAEhBO...** - copy it.
4. Message your new bot with /start or any text. Visit https://api.telegram.org/bot<YOUR_TOKEN>/getUpdates in a browser; the JSON response shows **"from": {"id": 1043232644}** under result - that's your chat ID (positive number).
5. Use “Telegram On Message” trigger to receive messages from the bot
6. Use “Telegram Send Message” with the chat ID you copied before to send messages to the bot

Example - Telegram node

https://docs.n8n.io/integrations/builtin/credentials/telegram/?utm_source=n8n_app&utm_medium=credential_settings&utm_campaign=create_new_credentials_modal#using-api-bot-access-to-ken

משימה

תכינו הודעה ותשלחו אותה לעצמכם דרך Telegram node

משיגים את המזהה משתמש שלכם דרך - [@userinfobot](https://t.me/userinfobot)

משימה

תקבלו הודעה מטלגרם, תשנו אותה (תוסיפו "!!") ותשלחו אותה לעצמכם חזרה דרך
Telegram node

משיגים את המזהה משתמש שלכם דרך -
@userinfobot

משימת מחשבון

Google classroom assignment

מיני פרויקט להגשה

צרו telegram chat bot המשתמש בAPIים קיימים באינטרנט
לבחירה!

<https://apivault.dev>

N8N Supabase API

<https://n8n.io/workflows/8258-learn-secure-webhook-apis-with-authentication-and-supabase-integration/>

חלוקת השיעור היום

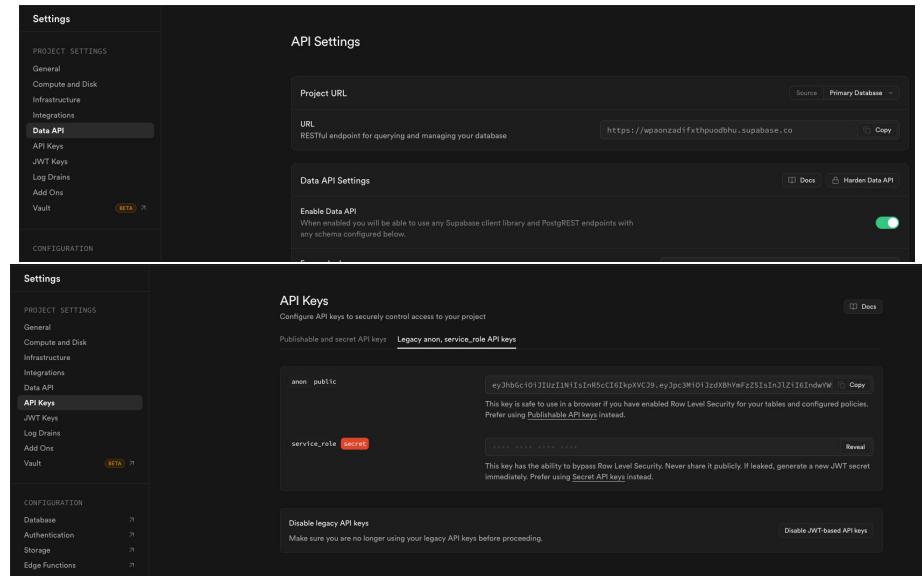
שעה 1: מעבר על פרויקטים

שעה 2-3: עבודה עם Supabase + מושימה

שעה 4: התחלה עבודה עם AI

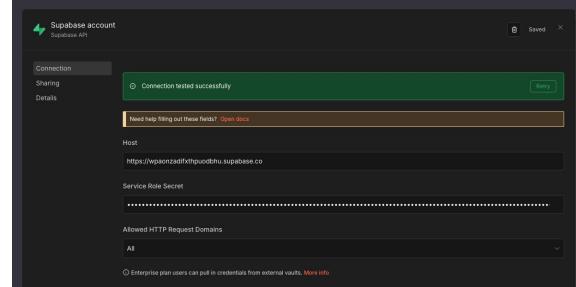
Create Supabase database

1. Create a project in Supabase
2. Retrieve the Host by going to “Project Settings” -> “Data API” -> “URL”
3. Get “Service Role Secret” by going to “Project Settings” -> “API Keys” -> “Legacy anon, service_role API keys” tab -> “service_role” secret
4. Paste them into Supabase Credentials



The image contains three screenshots of the Supabase Project Settings interface:

- Top Screenshot:** Shows the "Settings" page with the "Data API" tab selected. It displays sections for General, Compute and Disk, Infrastructure, Integrations, Data API (selected), API Keys, JWT Keys, Log Drains, Add Ons, and Vault (Beta). Below is a "CONFIGURATION" section.
- Middle Screenshot:** Shows the "API Settings" page. It includes a "Project URL" section with a URL field containing "https://woeonzaadifxthpuodbh.supabase.co" and a "Data API Settings" section where "Enable Data API" is turned on.
- Bottom Screenshot:** Shows the "API Keys" page with the "Legacy anon, service_role API keys" tab selected. It lists two keys:
 - anon public:** A public key with a copy button and a note about Row Level Security.
 - service_role secret:** A secret key with a reveal button and a note about bypassing Row Level Security.A "Disable legacy API keys" button is at the bottom.



The image shows a "Supabase account" dialog box with the "Saved" status. It has tabs for "Connection", "Sharing", and "Details". The "Connection" tab shows a green success message: "Connection tested successfully". The "Sharing" tab has a "Need help filling out these fields? Open docs" link. The "Details" tab shows the "Host" field with the value "https://woeonzaadifxthpuodbh.supabase.co" and the "Service Role Secret" field, which is currently empty. Below these are "Allowed HTTP Request Domains" set to "All" and a note about enterprise plan users pulling from external vaults.

פרויקט

בוט Telegram לניהול משתמשות עם Supabase