FIRST VIBE CODING FOR BEGINNERS · HANDBOOK

คู่มือ Workshop สร้าง Sales Page ขายสินค้า ด้วย AI Agent

คู่มือลงมือทำสำหรับมือใหม่ — สั่ง Claude Code สร้างหน้าเว็บขายสินค้าที่มีฐานข้อมูลจริง แก้เพิ่ม/ลบสินค้าเองได้ผ่านหลังบ้าน แล้วขึ้นออนไลน์ด้วย Cloudflare

ทุกขั้นตอนมี **prompt สำเร็จรูปให้คัดลอกใช้ได้ทันที** แค่แก้ข้อมูลใน [วงเล็บเหลี่ยม] แล้ววางให้ Agent ทำงานต่อ

รูปแบบ	เครื่องมือหลัก
ลงมือทำ · Hands-on	Claude Code
ฐานข้อมูล · DEPLOY	**ผลลัพธ์**
Supabase · Cloudflare	Sales Page + CRUD

ภาพรวม

เว็บแอปทำงานอย่างไร



ผู้ใช้ เปิดเว็บ → **Frontend** หน้าตาที่เห็น → **Backend** ตรรกะเบื้องหลัง → **Database** ที่เก็บข้อมูลจริง — คู่มือนี้พาคุณสั่ง AI Agent สร้างครบทั้งเส้นทาง

วิธีใช้คู่มือ

อ่านระบบสีให้เข้าใจ แล้วอ่านคู่มือได้ลื่นทั้งเล่ม

จำสี 4 สีนี้ไว้ แล้วจะรู้ทันทีว่าตรงไหนต้องพิมพ์ ตรงไหน Agent ทำให้ ตรงไหนต้องระวัง และตรงไหนคือผลลัพธ์ที่ควรได้

คุณพิมพ์ / สั่ง

กล่องสีเข้มพร้อมปุ่ม Copy คือ prompt ที่คุณคัดลอกไปวางให้ Claude Code

Agent ทำให้

สิ่งที่ AI Agent จะคิด สร้าง หรือแก้ให้อัตโนมัติหลังคุณสั่ง

ข้อควรระวัง

จุดที่มือใหม่มักพลาด — อ่านก่อนกด Enter จะช่วยประหยัดเวลา

คุณควรได้

ผลลัพธ์ที่จับต้องได้เมื่อจบแต่ละขั้น ใช้เช็คว่าทำถูกทาง

เคล็ดลับการใช้ prompt: ทุก prompt ในคู่มือเขียนเป็นภาษาไทยได้ Claude Code เข้าใจดี ส่วนคำที่อยู่ใน **[วงเล็บเหลี่ยม]** คือช่องที่คุณต้องแทนที่ด้วยข้อมูลจริงของคุณเองก่อนส่ง

ก่อนเริ่ม

เตรียม 3 บัญชี + เครื่องมือให้พร้อม

สมัครฟรีทั้งหมด ใช้อีเมลเดียวกันได้ทั้งสามที่ เตรียมโน้ตบุ๊ก 1 เครื่องพร้อมอินเทอร์เน็ต

01 · AI Agent

Claude Code

ตัว AI Agent หลักที่จะสร้างเว็บให้เรา ติดตั้งได้ที่ **claude.com/product/claude-code**

02 · ฐานข้อมูล

Supabase

ที่เก็บข้อมูลสินค้าจริง สมัครฟรีที่ **supabase.com** ด้วยบัญชี GitHub หรืออีเมล

03 · ขึ้นออนไลน์

Cloudflare

ใช้ Cloudflare Pages นำเว็บขึ้นออนไลน์ฟรี สมัครที่ **dash.cloudflare.com**

ถ้าติดตั้งเองไม่เป็น ให้ Agent ช่วยติดตั้ง

ไม่ต้องกลัว terminal — วาง prompt นี้ให้ Agent ตรวจเครื่องและบอกวิธีติดตั้งทีละขั้น

PROMPT · ติดตั้งเครื่องมือ

```
ฉันเป็นมือใหม่ ใช้ [Windows / macOS] อยากเริ่มทำเว็บด้วย Claude Code
ช่วยตรวจว่าเครื่องฉันมี Node.js และ Git แล้วหรือยัง
ถ้ายังไม่มี ให้บอกวิธีติดตั้งทีละขั้นแบบละเอียด ภาษาไทย
และอธิบายสั้น ๆ ว่าแต่ละอย่างมีไว้ทำอะไร
```

⚠ **ระวัง:** ทำขั้นนี้ก่อนวันเรียนถ้าทำได้ เพราะดาวน์โหลดโปรแกรมอาจใช้เวลา จะได้ไม่เสียเวลาช่วงเวิร์กช็อป

คู่มือเฉพาะทาง

ตั้งค่า Supabase ทีละขั้น — ตั้งแต่สมัครจนได้ Key ครบ

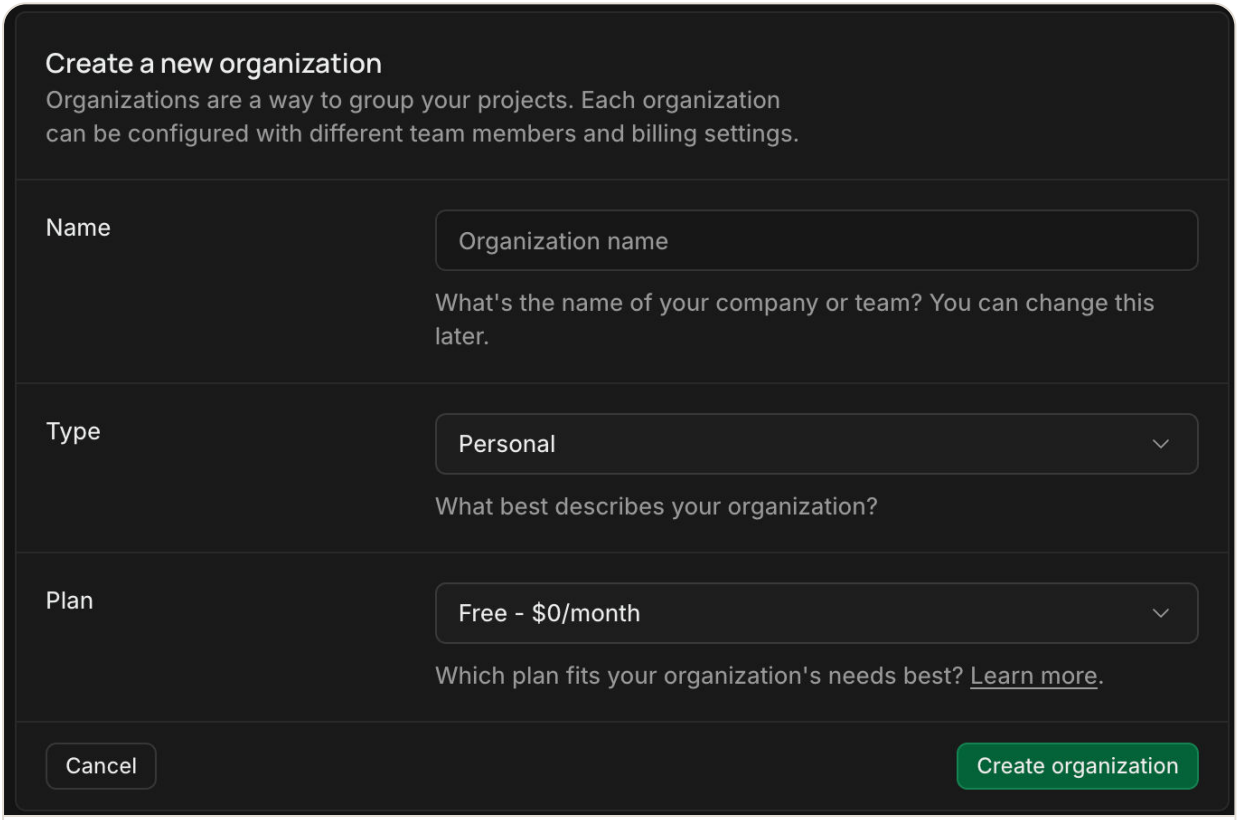
Supabase คือ “ฐานข้อมูล + หลังบ้าน” ที่เก็บข้อมูลสินค้าของเรา ทำตาม 6 ขั้นนี้ให้จบก่อนเข้า Workshop 2 แล้วคุณจะมีทั้ง **Project URL · anon key · Connection string** เก็บไว้ในไฟล์พร้อมใช้

1 เข้าเว็บ Supabase แล้วล็อกอิน

ไปที่ supabase.com → กด **Start your project** / **Sign in** → ล็อกอินด้วย **GitHub** (เร็วสุด) หรืออีเมลก็ได้ · ใช้อีเมลเดียวกับที่สมัคร Claude Code และ Cloudflare จะได้จำง่าย

2 สร้าง Organization (กลุ่มของโปรเจกต์)

ครั้งแรกที่เข้าใช้ ระบบจะให้สร้าง **Organization** ก่อน — มองว่าเป็น “บริษัท/ทีม” ที่เอาไว้รวมโปรเจกต์และคุมเรื่องบิลอยู่ที่เดียว กรอกตามนี้แล้วกด **Create organization**



หน้าจอ “Create a new organization”

> **Name** — ชื่อกลุ่ม/ทีม ตั้งเป็นชื่อร้านหรือชื่อตัวเองก็ได้ (เช่น `Vibe Coding`) **แก้ทีหลังได้** ไม่ต้องคิดนาน

> **Type** — ประเภทองค์กร เลือก `Personal` ถ้าเรียน/ทำเองคนเดียว (ตัวเลือกอื่นเช่น Company, Agency มีผลแค่เชิงข้อมูล ไม่กระทบการใช้งาน)

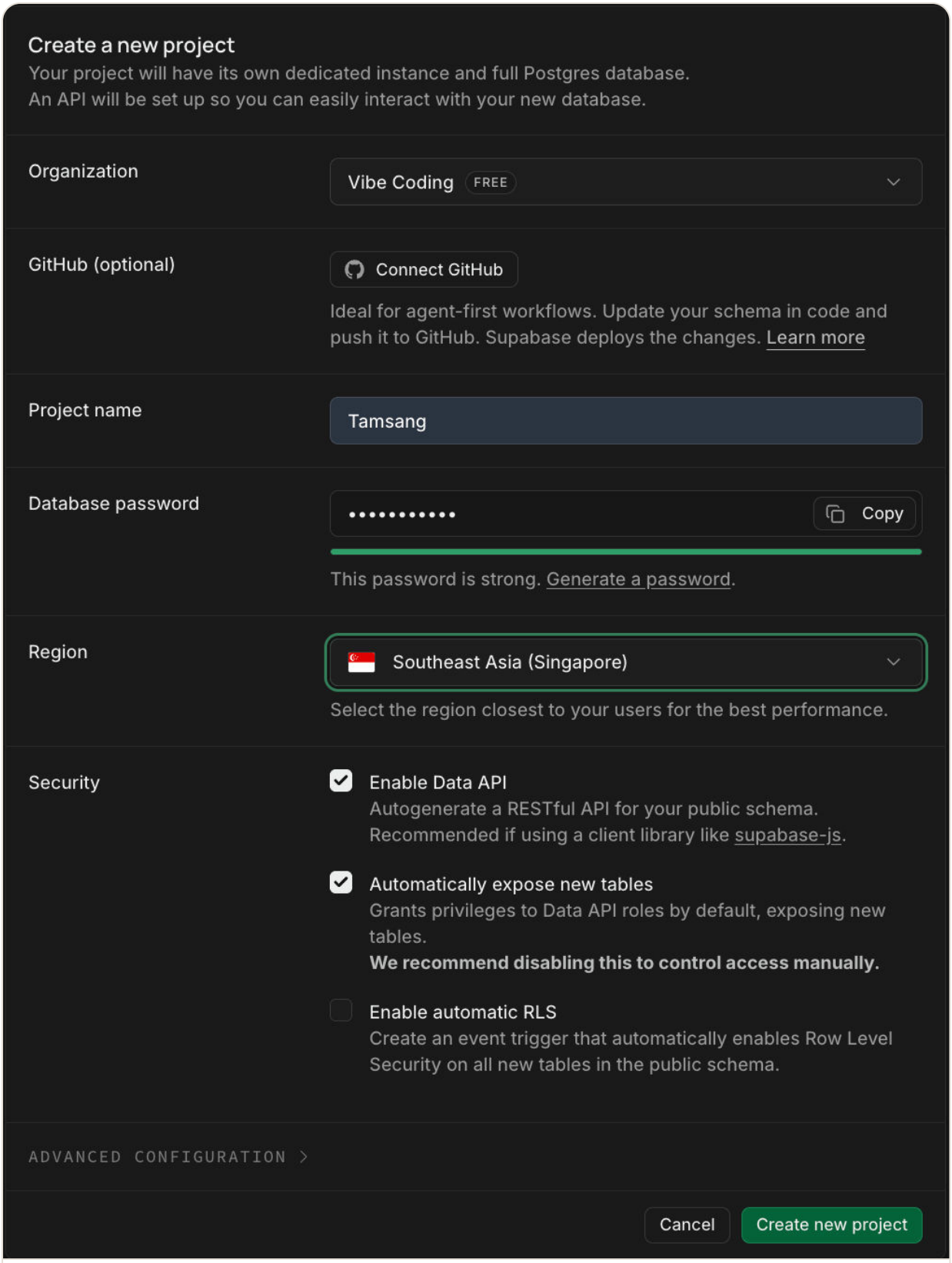
> **Plan** — เลือก `Free — $0/month` สำหรับคอร์สนี้ **เพียงพอทั้งหมด ไม่ต้องผูกบัตร**

ข้อจำกัดของ Plan Free ที่ควรรู้

- **สร้างได้ 2 โปรเจกต์ที่ใช้งานอยู่** ต่อ 1 บัญชี — ถ้าจะสร้างเพิ่มต้องลบ/หยุดตัวเก่า
- **พื้นที่ฐานข้อมูล 500 MB** และ **ไฟล์ (Storage) 1 GB** — เก็บข้อมูลสินค้าหลักพัน–หมื่นรายการได้สบาย
- **โปรเจกต์จะถูกพัก (pause) เมื่อไม่มีการใช้งาน ~7 วัน** — กดปุ่ม Restore ในแดชบอร์ดให้กลับมาใช้ได้ ข้อมูลไม่หาย
- ไม่มีระบบสำรองข้อมูลอัตโนมัติ (backup) ในแพลนฟรี — ถ้าใช้งานจริงจังควรอัปเป็น Pro

3 สร้าง Project (ฐานข้อมูลจริงของร้านเรา)

1 Project = ฐานข้อมูล Postgres 1 ชุด + API ของมันเอง กรอกให้ครบแล้วกด **Create new project** · ระบบจะใช้เวลาเตรียมประมาณ 1–2 นาที



หน้าจอ “Create a new project”

> **Organization** — เลือกกลุ่มที่เพิ่งสร้างในขั้นที่ 2 (จะมีป้าย `FREE` กำกับอยู่)

> **GitHub (optional)** — **ข้ามได้ ไม่ต้องกด** สำหรับคอร์สนี้ · มีไว้ให้คนที่อยากให้ Supabase อัปเดตโครงสร้างฐานข้อมูลตามโค้ดใน GitHub อัตโนมัติ

> **Project name** — ชื่อโปรเจกต์ ตั้งให้สื่อถึงร้าน เช่น `sales-page` หรือชื่อร้านของคุณ

Database password — **รหัสผ่านฐานข้อมูล สำคัญมาก** กด **Generate a password** ให้ระบบสุ่มแล้ว **กด Copy เก็บใส่ไฟล์ทันที** เพราะ **หน้านี้จะไม่แสดงให้ดูอีก** (ลืมแล้วต้องไปตั้งใหม่ที่ Reset database password) — จะได้ใช้ตอนต่อ Connection string ในขั้นที่ 6

Region — ที่ตั้งเซิร์ฟเวอร์ เลือกใกล้ผู้ใช้ที่สุดเพื่อให้เว็บเร็ว · คนไทยเลือก `Southeast Asia (Singapore)` · **เลือกแล้วเปลี่ยนทีหลังไม่ได้**

Enable Data API — **ติ๊กไว้** เพื่อให้ Supabase สร้าง API ให้อัตโนมัติ เว็บของเราถึงจะดึงข้อมูลได้ (ถ้าไม่ติ๊ก เว็บจะเชื่อมไม่ได้)

Automatically expose new tables — ติ๊กไว้ตอนเรียนเพื่อความสะดวก (ตารางใหม่จะใช้งานผ่าน API ได้ทันที) · ถ้าเป็นงานจริงที่เน้นความปลอดภัย Supabase แนะนำให้ปิดแล้วมาเปิดสิทธิ์เองทีละตาราง

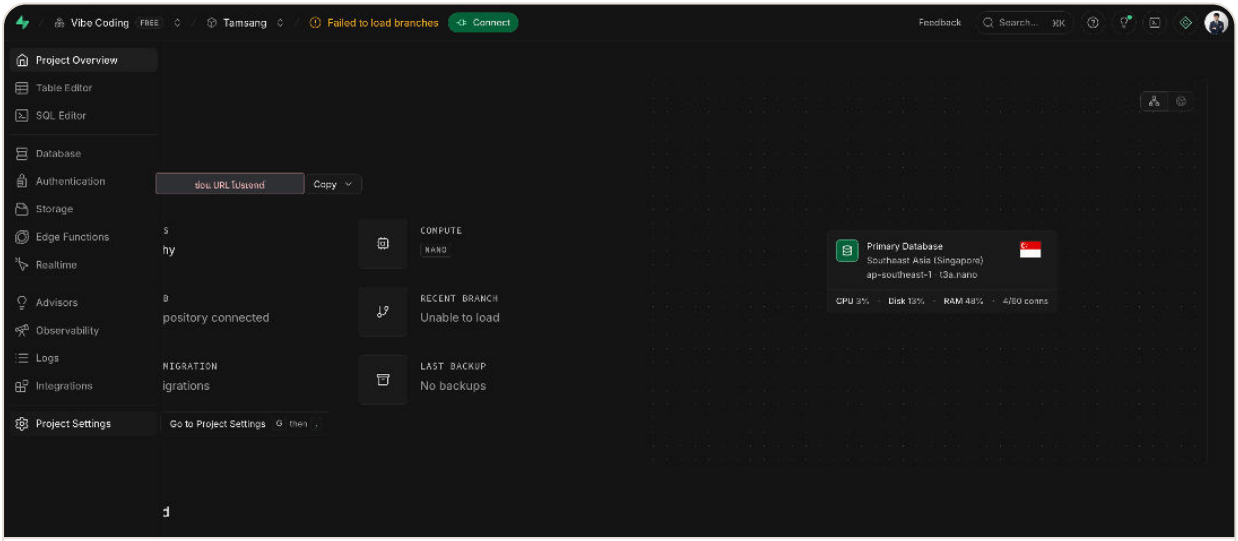
Enable automatic RLS — **Row Level Security** คือระบบล็อกสิทธิ์รายแถว · ตอนเรียนปล่อยว่างไว้ก่อนได้ เพื่อไม่ให้ติดขัด แต่ **ก่อนใช้งานจริงควรเปิดและตั้ง policy** ไม่งั้นใครก็แก้ข้อมูลได้

ข้อจำกัด Plan Free ในขั้นนี้

- เครื่องที่ได้คือขนาดเล็กสุด Nano (ราว 500 MB RAM · ~60 การเชื่อมต่อพร้อมกัน) — พอสำหรับเว็บเรียนและร้านเล็ก
- ตัวเลือก **ADVANCED CONFIGURATION** (เช่นเลือกเวอร์ชัน Postgres / เครื่องแรงขึ้น) ส่วนใหญ่ต้องเสียเงิน — **ไม่ต้องแตะ**
- ฟรีจำกัด **2 โปรเจกต์** ถ้าเคยสร้างไว้ครบแล้วจะสร้างใหม่ไม่ได้ ต้องลบตัวเก่าก่อน

4 เปิด Project Settings (แถบเมนูซ้ายมือ ล่างสุด)

เมื่อโปรเจกต์สร้างเสร็จ จะเข้าหน้า **Project Overview** · มองแถบเมนูด้านซ้าย เลื่อนลงล่างสุด แล้วกด **Project Settings** (ไอคอนฟันเฟือง) — ที่นี่คือที่เก็บ Key และการตั้งค่าทั้งหมดของโปรเจกต์

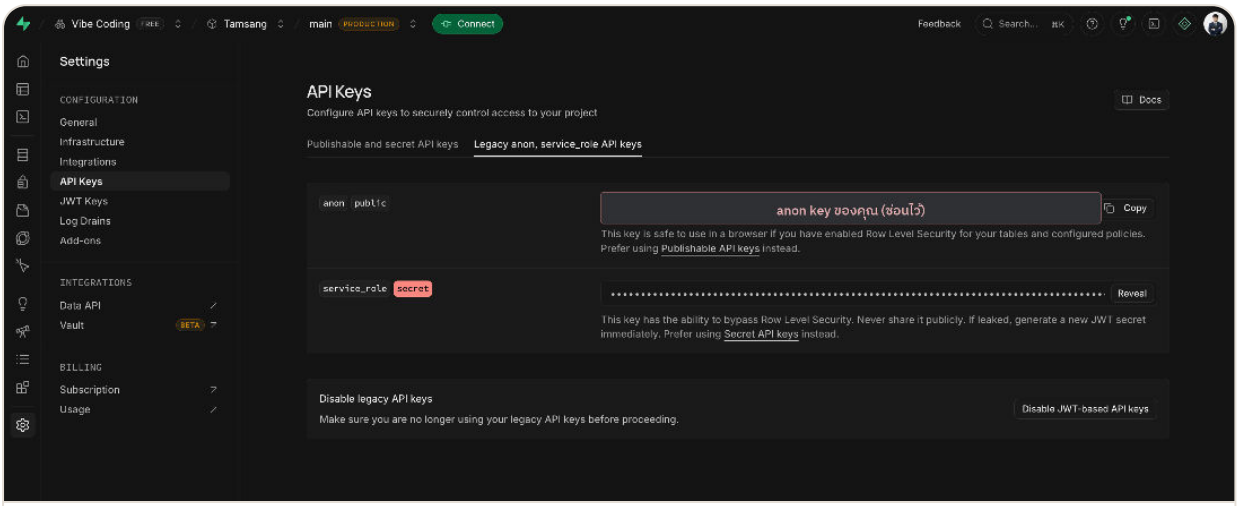


แถบเมนูซ้าย → **Project Settings** (ล่างสุด) · ปุ่ม **Connect** อยู่ด้านบนกลางจอ (ใช้ในขั้นที่ 6)

ตรงหน้านี้ยังบอกสถานะเครื่องด้วย เช่น `COMPUTE: NANO` และ Region ที่เลือกไว้ — ถ้าเห็น **Primary Database · Southeast Asia (Singapore)** แปลว่าตั้งค่าถูกแล้ว

5 คัดลอก API Key (แท็บ Legacy anon)

ใน Settings เลือกเมนู **API Keys** → กดแท็บ **Legacy anon, service_role API keys** → จะเห็นคีย์ 2 ตัว · คัดลอกเก็บใส่ไฟล์ของตัวเอง (เช่น `Key.txt`) ในโฟลเดอร์โปรเจกต์



Settings → **API Keys** → แท็บ **Legacy anon, service_role API keys**

> **anon / public** — **ตัวนี้คือตัวที่เราใช้** กด **Copy** แล้วเก็บไว้ · เป็นคีย์สำหรับฝั่งหน้าเว็บ ปลอดภัยพอที่จะอยู่ในโค้ดที่ผู้ใช้เห็นได้ **ถ้าเปิด RLS และตั้ง policy ไว้ถูกต้อง**

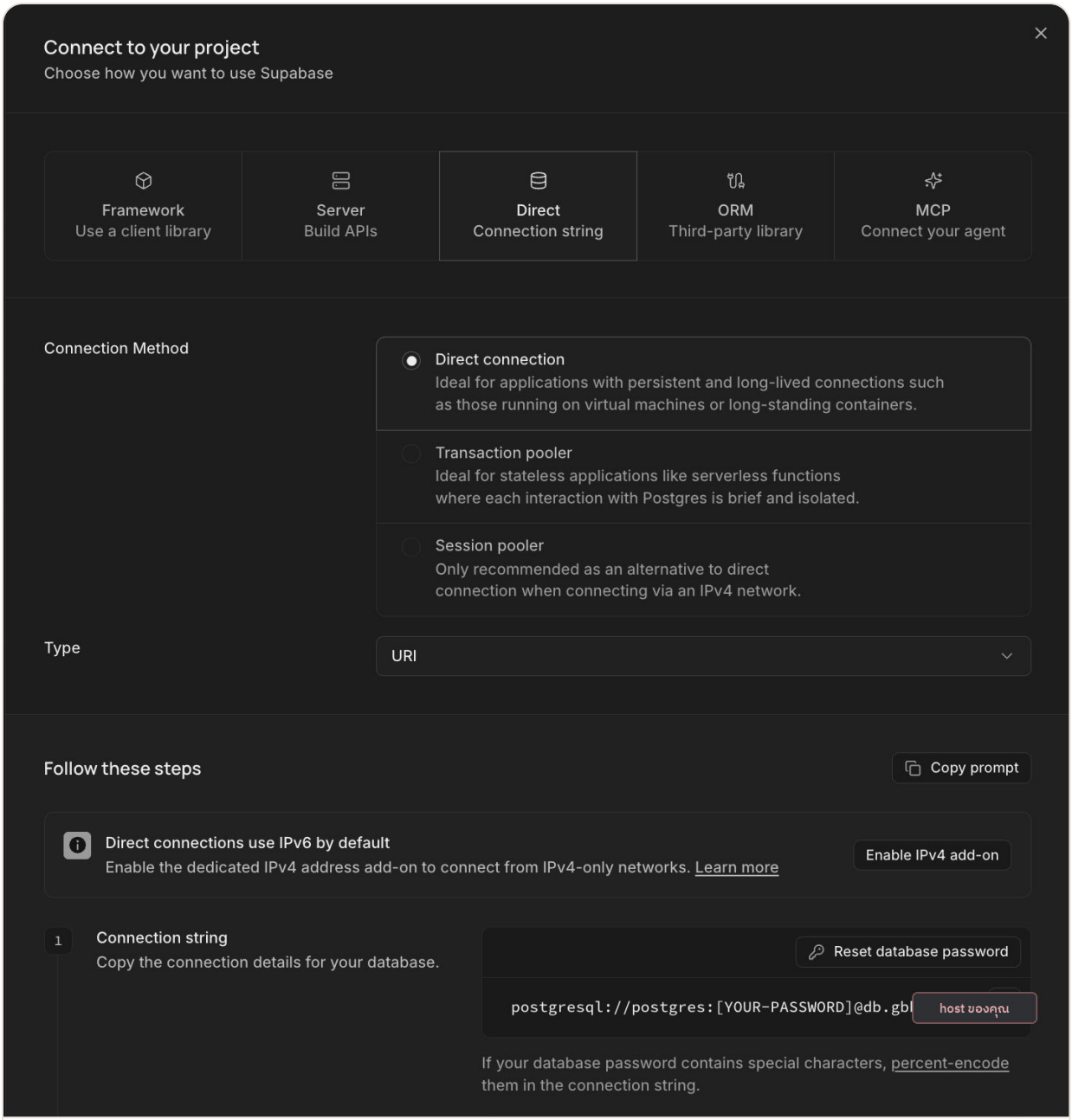
> **service_role / secret** — **ห้ามใช้ในหน้าเว็บเด็ดขาด** คีย์นี้ข้ามระบบสิทธิ์ทั้งหมด (ทำอะไรกับข้อมูลก็ได้) ถ้าหลุดออกไปคนอื่นลบข้อมูลเราได้ทันที · ใช้เฉพาะงานฝั่งเซิร์ฟเวอร์เท่านั้น ตอนเรียนนี้ **ไม่ต้องใช้**

> **Project URL** — อยู่ที่หน้า Project Overview หรือเมนู Data API หน้าตาเป็น `https://xxxxx.supabase.co` · คัดลอกเก็บคู่กับ anon key เพราะต้องใช้ทั้งคู่ในการเชื่อมเว็บ

> ⚠ **วิธีเก็บคีย์ที่ถูกต้อง:** เก็บลง**ไฟล์ในเครื่อง**แล้วบอก Agent ว่า “อ่านจากไฟล์นี้” — อย่าพิมพ์คีย์ลงในแชตตรง ๆ และ**อย่าอัปไฟล์คีย์ขึ้น GitHub**

6 คัดลอก Connection string (ปุ่ม Connect → Direct)

กดปุ่ม **Connect** ที่แถบบนสุดของจอ → เลือกแท็บ **Direct** → หัวข้อ Connection Method เลือก **Direct connection** → ช่อง Type เลือก **URI** → คัดลอก **Connection string** ด้านล่างไปเก็บในไฟล์เดียวกับคีย์



Connect → **Direct** → Direct connection → Type: **URI**

สิ่งที่ต้องแก้ก่อนใช้: สตริงที่คัดลอกมาจะมีคำว่า `[YOUR-PASSWORD]` อยู่ ให้แทนที่ด้วย **รหัสผ่านฐานข้อมูลที่ตั้งไว้ในขั้นที่ 3** (รวมวงเล็บเหลี่ยมออกด้วย)

> **หน้าตาก่อนแก้** — `postgresql://postgres:[YOUR-PASSWORD]@db.xxxxx.supabase.co:5432/postgres`

> **หลังแก้แล้ว** — `postgresql://postgres:รหัสจริงของคุณ@db.xxxxx.supabase.co:5432/postgres`

> **ถ้ารหัสมีอักขระพิเศษ** (เช่น `@ # ? / :`) ต้องแปลงเป็นรหัสพิเศษ (percent-encode) ก่อน ไม่งั้นเชื่อมไม่ติด — วิธีง่ายสุดคือกลับไป **Reset database password** แล้วให้ระบบสุ่มรหัสที่ไม่มีอักขระพิเศษ

> **ความต่างของ 3 โหมด** — **Direct connection** ใช้ตอนต่อจากเครื่องเราโดยตรง (ที่เราใช้ในคอร์สนี้) · **Transaction pooler** สำหรับงาน serverless ที่ต่อ–ตัดถี่ ๆ · **Session pooler** ใช้แทน Direct เมื่อเครือข่ายเป็น IPv4 เท่านั้น

ข้อจำกัด Plan Free ในขั้นนี้

- **Direct connection ใช้ IPv6 เป็นค่าเริ่มต้น** — ถ้าเน็ตที่บ้าน/ที่ทำงานเป็น IPv4 อย่างเดียวอาจต่อไม่ติด · ทางแก้ฟรีคือเปลี่ยนไปใช้ **Session pooler** (ส่วน IPv4 add-on เป็นบริการเสียเงิน)
- จำนวนการเชื่อมต่อพร้อมกันจำกัดตามเครื่อง Nano (~60 connections)

จบ 6 ขั้นแล้วคุณควรมีในไฟล์: Project URL · anon key · Database password · Connection string (ใส่รหัสแล้ว) — พร้อมเข้า Workshop 2 ได้เลย

แกนหลักของคู่มือ

4 Workshop · จากศูนย์สู่ Sales Page ที่ขึ้นออนไลน์จริง

ทำตามลำดับ 1 → 4 แต่ละ Workshop ต่อยอดจากอันก่อนหน้า ตัวอย่างใช้ร้านสมมติ **“NORTHPEAK”** (ร้านขายอุปกรณ์กิจกรรมกลางแจ้ง) — เปลี่ยนเป็นข้อมูลจริงของร้านคุณได้ทุกจุด ติ๊ก “ทำแล้ว” เพื่อนับความคืบหน้า

1

WORKSHOP 1 · FRONTEND

ออกแบบหน้า Sales Page (Front-end)

ประมาณ 60 นาที · ผลลัพธ์: เว็บขายสินค้า 4 หน้า รันบนเครื่องได้



ทำแล้ว

เป้าหมาย: สั่ง Agent สร้างเว็บขายสินค้าครบชุด 4 หน้า — **หน้าแรก · สินค้า · เกี่ยวกับเรา · ติดต่อกับทางร้าน** — ดีไซน์สวยและเปิดบนมือถือได้ดี

1 สร้างโฟลเดอร์โปรเจกต์และเปิด Claude Code

สร้างโฟลเดอร์ใหม่ เช่น `northpeak-shop` เปิด Claude Code ในโฟลเดอร์นั้น แล้ววาง prompt แรก

PROMPT 1.1 · สร้างโปรเจกต์

```
ช่วยสร้างโปรเจกต์เว็บใหม่ในโฟลเดอร์นี้ให้หน่อย
- เป็นเว็บแบบ static ใช้ HTML, CSS, JavaScript ธรรมดา (ยังไม่ต้องใช้ framework)
- ทำหน้า index.html ที่มีข้อความว่า "NORTHPEAK — เร็ว ๆ นี้" ไว้ก่อน
- ตั้งค่าให้รันดูในเบราว์เซอร์ได้ด้วยคำสั่งเดียว
เสร็จแล้วบอกฉันว่าต้องพิมพ์คำสั่งอะไรเพื่อเปิดดูเว็บ
และอธิบายสั้น ๆ ว่าแต่ละไฟล์ที่สร้างมีหน้าที่อะไร
```

ทำตามคำสั่งรันที่ Agent บอก แล้วเปิด `http://localhost` ในเบราว์เซอร์ — เห็นข้อความ NORTHPEAK แปลว่าพร้อมทำต่อ

2 วาง brief ร้านแล้วสั่งสร้าง 4 หน้า

ยิ่ง brief ชัด เว็บยิ่งตรงใจ เติมข้อมูลร้านคุณในช่อง [...] ให้ครบ แล้วส่ง prompt นี้

PROMPT 1.2 · brief + สร้าง 4 หน้า

```
ฉันอยากได้เว็บ Sales Page สำหรับร้านนี้:
- ชื่อร้าน: [NORTHPEAK]
- ขายอะไร: [อุปกรณ์และของใช้สำหรับกิจกรรมกลางแจ้ง]
- กลุ่มลูกค้า: [นักเดินป่าและคนชอบตั้งแคมป์]
- โทนแบรนด์: [น่าเชื่อถือ ทันสมัย โทนเขียวเข้ม–ดำ]

ช่วยสร้างเว็บ 4 หน้า: หน้าแรก, สินค้า, เกี่ยวกับเรา, ติดต่อกับทางร้าน
- มีเมนูนำทางด้านบนที่ลิงก์ทุกหน้าถึงกัน
- หน้าแรกมี hero พร้อมหัวข้อใหญ่และปุ่ม call-to-action ("ดูสินค้า")
- หน้าสินค้าแสดงสินค้าเป็นการ์ด (รูป ชื่อ ราคา ปุ่มดูรายละเอียด) ใส่สินค้าตัวอย่างไปก่อน
- หน้าติดต่อมีข้อมูลร้าน ช่องทางติดต่อ และแผนที่
- ดีไซน์สะอาด ทันสมัย และต้อง responsive เปิดบนมือถือสวย
ทำทีละหน้าและให้ฉันเปิดดูผลได้เรื่อย ๆ
```

3 ขัดเกลา: ใส่ของจริงแทนของตัวอย่าง

พอโครงหน้าตาลงตัว มักมีรอบ “ใส่ของจริง” เสมอ — เอาโลโก้ รูปสินค้า ฟอนต์จริงมาให้ Agent สลับให้ในคำสั่งเดียว

PROMPT 1.3 · ใส่ของจริง + ขัดเกลาให้มืออาชีพ

```
ช่วยแก้ไขเพิ่มเติมให้หน่อย:
- เปลี่ยนโลโก้เป็นไฟล์จริงที่ [วางพาธ/ชื่อไฟล์โลโก้]
- เปลี่ยนรูป Banner หน้าแรกเป็นรูปจริงที่ [วางพาธ/ชื่อไฟล์รูป]
- ใช้ฟอนต์ [ชื่อฟอนต์ เช่น Kanit] ทั้งเว็บ
- ห้ามใช้อีโมจิในหน้าเว็บเด็ดขาด ให้เปลี่ยนเป็นไอคอนที่สื่อความหมายแทนทั้งหมด
ช่วยปรับให้หน้าเว็บดูมีความเป็นมืออาชีพมากยิ่งขึ้น
```

Agent ทำให้: เขียน HTML/CSS ทุกหน้า จัด layout ให้ responsive และเชื่อมเมนูให้อัตโนมัติ — คุณแค่บอก “ชอบ/ไม่ชอบ” แล้วสั่งแก้เป็นรอบ

ระวัง: สั่งแก้ทีละเรื่อง อย่ารวบหลายข้อในครั้งเดียว — จะคุมทิศทางง่ายกว่าและรู้ทันทีถ้าพังตรงไหน

คุณควรได้: เว็บ 4 หน้า (หน้าแรก/สินค้า/เกี่ยวกับเรา/ติดต่อร้าน) ที่กดเมนูสลับหน้าได้ ดีไซน์เรียบร้อย และเปิดบนมือถือแล้วยังสวย

2

WORKSHOP 2 · CONNECT & KEYS

เชื่อมต่อ Supabase + Cloudflare (วิธีหา Key)

ประมาณ 60 นาที · ผลลัพธ์: หน้าสินค้าดึงข้อมูลจาก Supabase จริง + มี Key พร้อม deploy



ทำแล้ว

เป้าหมาย: สร้างตารางสินค้าใน Supabase, รู้ **วิธีไปหา Key ที่หน้าเว็บของ Supabase และ Cloudflare**, เก็บ Key ไว้อย่างปลอดภัย แล้วเชื่อมให้หน้า “สินค้า” ดึงข้อมูลจริง

1 ให้ Agent อธิบายและออกแบบตารางสินค้า

PROMPT 2.1 · ออกแบบตารางสินค้า

```
ฉันอยากเก็บ "รายการสินค้า" ของร้านไว้ในฐานข้อมูล Supabase
เพื่อให้แก้เพิ่ม/ลบสินค้าได้ทีหลังโดยไม่ต้องแก้โค้ด
ช่วยอธิบายแบบมือใหม่ว่าควรมีตารางอะไร มีคอลัมน์อะไรบ้าง
(เช่น ชื่อสินค้า, รายละเอียด, ราคา, รูปสินค้า, สถานะขาย/หมด)
แล้วเขียนขั้นตอนสร้างตารางนี้ใน Supabase ให้ฉันทำตามทีละคลิก
```

2 ไปหา Key ที่หน้าเว็บ แล้วเก็บไว้ในไฟล์

Supabase key: เปิด Supabase → เลือกโปรเจกต์ → **Project Settings → API Keys** (แท็บ Legacy anon) จะเห็น **anon key** คู่กับ **Project URL** — คัดลอกทั้งสองค่ามาเก็บในไฟล์ข้อความในโฟลเดอร์โปรเจกต์ (เช่น `Key.txt`) ไม่พิมพ์ลงแชตตรง ๆ แล้วให้ Agent อ่านจากไฟล์แทน

ยังไม่ได้สร้างโปรเจกต์ Supabase หรือหา Key ไม่เจอ? **ดูคู่มือตั้งค่า Supabase ทีละขั้น (6 ขั้น พร้อมภาพหน้าจอ) →**

Cloudflare key: เตรียม API Token ของ Cloudflare ไว้ในไฟล์เดียวกันด้วย (จะได้ใช้ต่อตอน Deploy ใน Workshop 4) — ไม่รู้จะเริ่มตรงไหน กดดูคู่มือทีละขั้นได้เลย

คู่มือสร้าง Cloudflare API Key →

PROMPT 2.2 · เชื่อม Supabase แล้วดึงสินค้ามาแสดง

```
ฉันเก็บวิธีเชื่อมต่อ Supabase (Project URL และ anon key) ไว้ที่ไฟล์ [ชื่อไฟล์] แล้ว
ช่วยเชื่อมเว็บนี้เข้ากับ Supabase ให้หน่อย
- ทำให้หน้า "สินค้า" ดึงรายการสินค้าจากตารางใน Supabase มาแสดงแทนข้อมูลที่
hardcode ไว้
- ถ้าฐานข้อมูลยังว่าง ให้แสดงข้อความว่า "ยังไม่มีสินค้าในขณะนี้"
อธิบายให้ฉันเข้าใจด้วยว่าข้อมูลเดินทางจากฐานข้อมูลมาที่หน้าเว็บอย่างไร
```

⚠ **ระวัง — เรื่อง key:** ใช้ **anon key** ฝั่งเว็บเท่านั้น ห้ามใส่ **service_role key** ในโค้ดหน้าเว็บเด็ดขาด เพราะมีสิทธิ์เต็มและจะรั่วสู่สาธารณะ

ถ้า Agent หยุดถามยืนยันก่อนสร้างตาราง: เป็นเรื่องปกติ — การแก้ฐานข้อมูลจริงย้อนกลับยาก Agent ที่ดีจะสรุปให้ฟังก่อนแล้วถามยืนยัน อ่านสรุปแล้วพิมพ์ยืนยันได้เลยถ้าตรงกับที่ต้องการ

คุณควรได้: หน้าสินค้าดึงข้อมูลจาก Supabase จริง (แก้ในตาราง แล้ว refresh เว็บ ข้อมูลเปลี่ยนตาม) และมี Key ของ Supabase + Cloudflare เก็บในไฟล์พร้อมใช้ต่อ

3

WORKSHOP 3 · ADMIN CRUD

หน้า Admin หลังบ้าน — เพิ่ม/ลบ/แก้ไข ด้วย PIN

ประมาณ 60 นาที · ผลลัพธ์: จัดการสินค้าผ่านหน้าเว็บได้เอง เข้าด้วย PIN



ทำแล้ว

เป้าหมาย: นี่คือหัวใจของ “แก้เองได้ ใช้ระยะยาว” — สร้างหน้า admin หลังบ้านที่เพิ่ม ลบ แก้ไขสินค้าได้เอง ผ่านหน้าเว็บ (CRUD) เข้าผ่าน **Modal กรอก PIN 6 หลัก** โดยไม่ต้องทำระบบ Login — Basic ง่าย ๆ แต่ยืดหยุ่นพอสำหรับใช้งานจริง

1 สร้างหน้า admin ที่ทำ CRUD ครบ + Modal PIN

ระบุทุกอย่างในคำสั่งเดียว: การเข้าด้วย PIN, ฟอร์ม, ตาราง และปุ่มแก้/ลบ — ไม่ต้องทำระบบ Login เต็มรูปแบบ

PROMPT 3.1 · หน้า admin CRUD + เข้าด้วย PIN 6 หลัก

```
ช่วยสร้างหน้า admin สำหรับจัดการ "รายการสินค้า" ในตาราง Supabase
โดยวิธีการเข้าถึงหน้านี้ ไม่ต้องทำหน้า Login อะไรให้มันยิ่งใหญ่ ขอแค่เมื่อเข้ามาที่หน้า
admin.html
ให้มี Modal เด้งขึ้นมาให้กรอกรหัส PIN 6 หลัก ถ้ากรอกถูกเข้าหน้า admin ได้ ถ้ากรอกผิด
จะเข้าไม่ได้

- มีฟอร์มเพิ่มสินค้าใหม่ (Create)
- แสดงสินค้าทั้งหมดเป็นตาราง (Read)
- แต่ละแถวมีปุ่มแก้ไข (Update) และปุ่มลบ (Delete) พร้อมถามยืนยันก่อนลบ
- เมื่อเพิ่ม/แก้/ลบ แล้ว ให้ตารางอัปเดตทันทีโดยไม่ต้องรีเฟรชเอง
ทำให้ใช้งานง่ายสำหรับคนที่ไม่ใช่โปรแกรมเมอร์
```

Agent จะถามรหัส PIN — ตอบเลข 6 หลักที่จำง่ายแต่คนอื่นเดายาก แล้วปล่อยให้ Agent เก็บให้

2 เชื่อมหน้าสินค้าสาธารณะให้อัปเดตตามด้วย

จุดพลาดบ่อย: มีหน้า admin ไม่ได้แปลว่าหน้าสินค้าที่ลูกค้าเห็นจะอัปเดตเอง ต้องสั่งอีกรอบให้ดึงข้อมูลจากตารางเดียวกันไปแสดง

PROMPT 3.2 · ให้หน้าสินค้าจริงอัปเดตตาม

```
ตอนนี้ฉันเพิ่ม/แก้สินค้าผ่านหน้า admin ได้แล้ว
ช่วยทำให้หน้า "สินค้า" ที่ลูกค้าเห็นดึงข้อมูลจากตารางเดียวกันมาแสดงสดด้วย
ถ้ายังไม่มีสินค้าในตาราง ให้ขึ้นข้อความที่เข้าใจง่ายแทน เช่น "ยังไม่มีสินค้าในขณะนี้"
```

⚠ **ระวัง:** ตั้งค่าสิทธิ์ (RLS) ใน Supabase ให้ถูก ไม่งั้นข้อมูลถูกแก้จากภายนอกได้ — และ **PIN แบบนี้กันแค่ระดับหน้าเว็บ** ไม่ใช่การป้องกันฐานข้อมูลจริงจัง เหมาะกับงานใช้ภายในร้าน

✓ **คุณควรได้:** หน้า admin ที่เข้าด้วย PIN 6 หลักได้ เพิ่ม/แก้/ลบสินค้าได้จริง และผลไปแสดงบนหน้าสินค้าทันที — เว็บ “แก้เองได้” แล้ว

4 WORKSHOP 4 · DEPLOY + โดเมน

พา Deploy ขึ้นออนไลน์จริงด้วย Cloudflare



ทำแล้ว

ประมาณ 60 นาที (Deploy 30 นาที · โดเมน 30 นาที) · ผลลัพธ์: เว็บ live เปิดจากมือถือใครก็ได้

เป้าหมาย: นำ Sales Page ขึ้น Cloudflare Pages ให้ได้ลิงก์จริง (*.pages.dev) ที่เปิดได้ทุกอุปกรณ์ พร้อมต่อ Supabase บน production — และถ้ามีโดเมนของตัวเอง ต่อโดเมนกับแตก subdomain ได้ด้วย

1 ใช้ Cloudflare API Token ที่เตรียมไว้แล้ว deploy

Token ที่คุณสร้างและเก็บไว้ในไฟล์ตั้งแต่ Workshop 2 จะได้ใช้ตรงนี้ — ถ้ายังไม่ได้สร้าง กดดูคู่มือทีละขั้นตอนได้

คู่มือสร้าง Cloudflare API Key →

PROMPT 4.1 · deploy ขึ้น Cloudflare

```
ฉันอยากนำเว็บนี้ขึ้นออนไลน์ด้วย Cloudflare Pages
ฉันเก็บ Cloudflare API Token ไว้ที่ไฟล์ [ชื่อไฟล์] แล้ว
ช่วยพาฉันทำทีละขั้นแบบมือใหม่ ตั้งค่าให้ deploy สำเร็จ
อย่าลืมตั้งค่า Supabase URL / key บน production ให้เว็บจริงดึงข้อมูลได้ด้วย
บอกฉันทุกปุ่มที่ต้องกด และถ้าต้องใช้คำสั่งใน terminal ให้เขียนให้ครบ
```

ถ้า Agent หยุดถามยืนยันก่อน deploy: เป็นเรื่องปกติ — deploy คือการเปิดเว็บสู่สาธารณะจริง Agent ที่ดีจะสรุปก่อนแล้วถามยืนยัน อ่านแล้วตอบ “ยืนยัน” ได้เลยถ้าตรงกับที่ต้องการ

2 ตรวจว่าเว็บจริงใช้งานได้ครบ

PROMPT 4.2 · ตรวจหลัง deploy

```
เว็บขึ้นออนไลน์แล้วที่ลิงก์ [วางลิงก์ .pages.dev ของคุณ]
ช่วยไล่เช็คกับฉันว่าอะไรที่ต้องทดสอบบ้าง เช่น
- ทุกหน้าเปิดได้และเมนูใช้ได้
- หน้าสินค้าดึงข้อมูลจาก Supabase บน production ได้
- หน้า admin เข้าด้วย PIN แล้วเพิ่ม/แก้/ลบ ข้อมูลอัปเดตจริง
ถ้าเจอจุดที่พัง ให้ช่วยไล่หาสาเหตุทีละอย่าง
```

3 ต่อโดเมนของตัวเอง (ถ้ามี)

ลิงก์ `*.pages.dev` ใช้งานได้จริงและฟรีตลอด — แต่ถ้าจะขายของ ชื่อเป็นของตัวเองน่าเชื่อถือกว่ามาก เลือกใช้ prompt ตามสถานการณ์ของคุณ ไม่มีโดเมนก็ข้ามขั้นนี้ไปได้

กรณี ก — ยังไม่มีโดเมน อยากซื้อใหม่ ซื้อที่ Cloudflare เลยจะง่ายที่สุด เพราะอยู่บัญชีเดียวกับเว็บ ไม่ต้องตั้งค่า DNS เอง

PROMPT 4.3 · ซื้อโดเมนใหม่ที่ Cloudflare แล้วผูกเลย

```
ฉันยังไม่มีโดเมน อยากซื้อใหม่ที่ Cloudflare ให้อยู่บัญชีเดียวกับเว็บที่เพิ่ง deploy
ชื่อที่อยากได้คือ [yourshop.com] · ธุรกิจของฉันคือ [ขายรองเท้าผ้าใบ]
Pages project ของฉันชื่อ [ชื่อโปรเจกต์]

ช่วยพาฉันทีละขั้นแบบมือใหม่:
1. เช็คว่าชื่อนี้ยังว่างไหม ถ้าไม่ว่าง เสนอชื่อสำรองที่สั้น จำง่าย ไม่มีขีดกลาง มาให้ 5 ชื่อ
2. บอกทุกเมนูที่ต้องกดใน Cloudflare ตั้งแต่ค้นหาชื่อจนจ่ายเงินเสร็จ
3. เตือนฉันเรื่องเปิด Auto-renew และค่าต่ออายุรายปีที่ต้องจ่ายทุกปี
4. ซื้อเสร็จแล้วพาฉันผูกโดเมนเข้ากับ Pages project ให้เปิดด้วย https ได้
   ผูกให้ครบทั้งแบบมี www และไม่มี www
บอกด้วยว่าแต่ละขั้นควรเห็นอะไรบนหน้าจอ ฉันจะได้รู้ว่าทำถูก
```

กรณี ข — มีโดเมนอยู่แล้วที่เจ้าอื่น เช่นเคยซื้อไว้กับ GoDaddy, Namecheap หรือผู้ให้บริการไทย — เอามาใช้ได้ **ไม่ต้องซื้อใหม่ ไม่ต้องย้ายทะเบียนโดเมน**

PROMPT 4.4 · โดเมนอยู่เจ้าอื่น เอามาเชื่อม Cloudflare

```
ฉันมีโดเมน [yourshop.com] อยู่แล้ว จดไว้ที่ [ชื่อผู้ให้บริการ เช่น GoDaddy]
อยากเอามาใช้กับ Cloudflare Pages project ชื่อ [ชื่อโปรเจกต์] โดยไม่ย้ายทะเบียน
โดเมน

ช่วยพาฉันทีละขั้นแบบมือใหม่:
1. อธิบายให้ฉันเลือกก่อนว่าควรใช้ทางไหน ระหว่าง
   – ย้าย nameserver มาให้ Cloudflare ดูแล DNS
   – หรือเพิ่ม CNAME ไว้ที่ผู้ให้บริการเดิม
   บอกข้อดีข้อเสียของแต่ละทางด้วยภาษาคนธรรมดา
2. พอเลือกได้แล้ว บอกทุกปุ่มที่ต้องกด ทั้งฝั่ง Cloudflare และฝั่ง [ชื่อผู้ให้บริการ]
3. สำคัญมาก – โดเมนนี้ฉันใช้รับส่งอีเมล [มี / ไม่มี]
   ถ้ามี ให้ไล่เช็คกับฉันว่า MX record ถูกย้ายมาครบก่อนเปลี่ยน nameserver
   ห้ามให้ฉันกดเปลี่ยนจนกว่าจะเทียบครบ
4. บอกว่าต้องรอนานแค่ไหน และจะดูตรงไหนว่าสำเร็จแล้ว
```

⚠ **ระวัง – อีเมลล่มตอนย้าย nameserver:** ถ้าโดเมนนั้นใช้รับส่งอีเมล (ชื่อคุณ@yourshop.com) อยู่ ต้องเทียบ **MX record** ให้ครบก่อนกดเปลี่ยน ไม่งั้นอีเมลจะหยุดรับทันที · โดเมน `.co.th` ซื้อที่ Cloudflare ไม่ได้ แต่ **ชี้ nameserver มาใช้แบบนี้ได้ปกติ**

4 เพิ่มเว็บที่ 2 บนโดเมนเดิม (subdomain)

โดเมนที่ซื้อไว้ **1 ชื่อ แตกเป็นเว็บย่อยได้ไม่จำกัด ฟรี ไม่ต้องซื้อเพิ่ม** – เช่น `promo.yourshop.com` สำหรับหน้าแคมเปญ หรือ `blog.yourshop.com` สำหรับบทความ โดยหน้าร้านหลักไม่กระทบเลย

PROMPT 4.5 · deploy เว็บที่ 2 เป็น subdomain

```
เว็บร้านหลักของฉันอยู่ที่ [yourshop.com] ผูกกับ Pages project [ชื่อโปรเจกต์หลัก] แล้ว
ตอนนี้อยากได้เว็บที่ 2 สำหรับ [หน้าแคมเปญ 11.11] ให้เปิดที่ [promo.yourshop.com]

ช่วยพาฉันทีละขั้นแบบมือใหม่:
1. สร้างเว็บที่ 2 ในโฟลเดอร์ใหม่ แยกจากของเดิม – ห้ามแก้ไฟล์ของเว็บร้านหลัก
2. deploy ขึ้น Cloudflare Pages เป็น project ใหม่
   ย้ำ: ห้าม deploy ทับ project เดิม ไม่งั้นหน้าร้านหลักจะหาย
3. ผูก [promo.yourshop.com] เข้ากับ project ใหม่นี้ ให้เปิดด้วย https ได้
4. ตรวจยืนยันกับฉันว่า [yourshop.com] เดิมยังเปิดได้ปกติ ไม่ได้รับผลกระทบ
ถ้าจะเพิ่ม subdomain ตัวที่ 3 ในอนาคต ต้องทำอะไรบ้าง สรุปให้ฉันเก็บไว้ด้วย
```

กฎง่าย ๆ: 1 subdomain = 1 Pages project — จุดพลาดที่เจอบ่อยที่สุดคือเอาเว็บที่ 2 ไป deploy ทับ project เดิม แล้วหน้าร้านหลักหายไป ต้องสร้าง project ใหม่เสมอ ส่วนโดเมนจ่ายครั้งเดียวต่อปี แตก subdomain กี่ตัวก็ได้ ไม่มีค่าใช้จ่ายเพิ่ม

ระวัง — Deploy แล้วข้อมูลไม่ขึ้น: ส่วนใหญ่เพราะยังไม่ได้ตั้งค่า Supabase key บน production ถ้าเจอ ให้คัดลอก error จากหน้า Cloudflare มาวางให้ Agent ช่วยไล่แก้ทีละขั้น

คุณควรได้: ลิงก์ Sales Page จริงที่ส่งให้ใครเปิดก็ได้ ใช้ CRUD หลัง PIN ได้ครบ และเชื่อมฐานข้อมูลบน production เรียบร้อย — ถ้าต่อโดเมนแล้วก็จะเปิดด้วยชื่อร้านตัวเองได้ พร้อมแตก subdomain เพิ่มเองเป็น

แก้เอง · ใช้ระยะยาว

คลัง Prompt สำหรับดูแลเว็บเองหลังจบคอร์ส

หัวใจคอร์สนี้คือ “เว็บที่ดูแลต่อเองได้” — เก็บ prompt ชุดนี้ไว้ใช้ตอนอัปเดตข้อมูล เพิ่มหน้าใหม่ แก้ปัญหา หรือ deploy รอบใหม่

อัปเดตเนื้อหา / สินค้า แล้ว deploy รอบใหม่

PROMPT · แก้แล้ว deploy ซ้ำ

```
ช่วยแก้เนื้อหาในเว็บให้หน่อย:
- [เปลี่ยนเบอร์โทรในหน้าติดต่อเป็น 02-xxx-xxxx]
- [แก้ข้อความแนะนำร้านในหน้าเกี่ยวกับเราตามนี้: ...]
แก้แล้วให้ฉันดูผลบนเครื่องก่อน ถ้าโอเคค่อยพาฉัน deploy ขึ้น Cloudflare อีกรอบ
และบอกวิธีตรวจว่าเวอร์ชันใหม่ขึ้นแล้วจริง
```

เพิ่มหน้าหรือฟีเจอร์ใหม่

PROMPT · เพิ่มหน้าใหม่

```
ฉันอยากเพิ่มหน้าใหม่ชื่อ [โปรโมชั่น / รีวิวลูกค้า]
- ให้ดึงข้อมูลจากตารางใหม่ใน Supabase เหมือนหน้าสินค้า
- เพิ่มลิงก์หน้านี้เข้าไปในเมนูด้วย
- ทำให้จัดการข้อมูลหน้านี้ในหน้า admin ได้เหมือนกัน
ช่วยออกแบบตารางที่ควรใช้ให้ก่อน แล้วค่อยลงมือทำ
```

ให้ Agent ตรวจสุขภาพเว็บและความปลอดภัย

PROMPT · health & security check

```
ช่วยตรวจเว็บนี้ให้ฉันหน่อย:
- มี key หรือข้อมูลลับหลุดอยู่ในโค้ดฝั่งหน้าเว็บไหม
- สิทธิ์การเข้าถึงตารางใน Supabase ตั้งค่าปลอดภัยหรือยัง
- มีจุดไหนที่ควรปรับปรุงก่อนใช้งานจริงจัง
สรุปเป็นรายการให้ฉันเข้าใจง่าย พร้อมระดับความสำคัญ
```

เป้าหมายระยะยาว: ใช้ prompt เหล่านี้จนคล่อง Sales Page ของคุณจะอัปเดตเองได้ตลอด ไม่ต้องจ้างแก้ทีละครั้ง และ workflow เดียวกันนี้ใช้สร้างโปรเจกต์อื่นต่อได้

คำศัพท์

ศัพท์ที่ได้ใช้จริงในวันนี้ อธิบายด้วยภาษาคน

ไม่ต้องท่องจำ อ่านผ่าน ๆ ให้คุ้นตา เวลาเจอในคู่มือหรือตอนคุยกับ Agent จะได้ไม่งง

Sales Page หน้าขายสินค้า

หน้าเว็บที่ทำมาเพื่อขายของ — โชว์สินค้า ราคา และชวนให้ลูกค้าติดต่อ/สั่งซื้อ

Frontend หน้าบ้าน

หน้าตาเว็บที่ผู้ใช้เห็นและกดใช้งาน — ปุ่ม เมนู รูป ข้อความ

Backend หลังบ้าน

ตรรกะและระบบเบื้องหลังที่ผู้ใช้ไม่เห็น เช่น การจัดการข้อมูลสินค้า

Database ฐานข้อมูล

ที่เก็บข้อมูลจริงอย่างเป็นระเบียบ เช่น รายการสินค้า ราคา ข้อมูลติดต่อ

API

ช่องทางให้สองระบบคุยกัน เช่น เว็บขอข้อมูลจากฐานข้อมูลผ่าน API

CRUD

4 การกระทำกับข้อมูล: Create เพิ่ม · Read อ่าน · Update แก้ · Delete ลบ

Modal

กล่องเล็ก ๆ ที่เด้งซ้อนขึ้นมาบนหน้าเว็บ เช่น กล่องให้กรอก PIN ก่อนเข้าหน้า admin

PIN

รหัสตัวเลขสั้น ๆ (ในคอร์สนี้ใช้ 6 หลัก) ใช้กันคนทั่วไปไม่ให้เข้าหน้าหลังบ้านแบบง่าย ๆ

Deploy

การนำเว็บจากเครื่องเราขึ้นไปอยู่บนออนไลน์ให้คนอื่นเปิดได้

Domain โดเมน

ชื่อเว็บที่คนพิมพ์เข้ามา เช่น yourshop.com

Localhost

เว็บที่รันอยู่บนเครื่องเราเองเพื่อดูผลก่อนขึ้นจริง (มีแค่เราเห็น)

Terminal

หน้าต่างพิมพ์คำสั่งกับคอมพิวเตอร์ — ที่ที่เราคุยกับ Claude Code

Prompt

ข้อความที่เราพิมพ์สั่ง AI Agent ให้ทำงาน ยิ่งชัดยิ่งได้งานตรงใจ

Supabase

บริการฐานข้อมูลสำเร็จรูปที่ต่อกับเว็บได้ง่าย ใช้ในคอร์สนี้

Cloudflare Pages

บริการนำเว็บขึ้นออนไลน์ฟรี ได้ลิงก์ *.pages.dev ทันที

anon key

กุญแจสาธารณะของ Supabase ใช้ฝั่งหน้าเว็บได้ ปลอดภัยกว่า service key

API Token Cloudflare

กุญแจที่ให้ Agent สั่ง deploy ขึ้น Cloudflare แทนเราได้ เก็บเป็นความลับเหมือน key อื่น

RLS Row Level Security

กติกาที่ตั้งไว้ในฐานข้อมูลว่าใครอ่าน/แก้/ลบข้อมูลแถวไหนได้บ้าง — ต้องตั้งให้ถูกก่อนเปิดเว็บจริง

Environment Variable

ที่เก็บค่าลับ (เช่น key) แยกจากโค้ด เพื่อไม่ให้หลุดสู่สาธารณะ

Responsive

เว็บที่ปรับหน้าตาให้อ่านสวยทั้งบนคอมและมือถือโดยอัตโนมัติ

แก้ปัญหา

เจอปัญหาบ่อย ๆ ตอนไหน ใช้ prompt ไหนแก้

มือใหม่เกือบทุกคนเจอสถานการณ์เหล่านี้ กดเปิดดูแล้วคัดลอก prompt ไปวางให้ Agent ช่วยแก้ได้เลย

? Agent แก้แล้วเว็บพัง / ขึ้นหน้าขาว

อย่าเพิ่งตกใจ ให้ย้อนกลับก่อน แล้วค่อยแก้ทีละจุด คัดลอก prompt นี้:

PROMPT · ย้อนกลับ

```
หลังจากแก้ล่าสุด เว็บพัง/ขึ้นหน้าขาว
ช่วยย้อนการแก้ครั้งล่าสุดกลับไปสถานะที่ยังใช้ได้
แล้วอธิบายว่าอะไรทำให้พัง จากนั้นค่อยแก้ใหม่ทีละขั้น
```

? เปิด localhost แล้วไม่ขึ้นอะไรเลย

มักเป็นเพราะ server ยังไม่รัน หรือใช้ลิงก์/พอร์ตผิด ลองส่ง prompt นี้:

PROMPT · เปิดเว็บไม่ได้

```
ฉันเปิด localhost แล้วไม่ขึ้นอะไรเลย
ช่วยตรวจว่า server รันอยู่ไหม ต้องเปิดที่ลิงก์และพอร์ตอะไร
บอกฉันทีละขั้นว่าต้องพิมพ์คำสั่งอะไรและเปิดลิงก์ไหน
```

? สินค้าจาก Supabase ไม่แสดงบนเว็บ

เช็คได้หลายจุด ทั้ง key, ชื่อตาราง, และสิทธิ์ ให้ Agent ไล่ให้:

PROMPT · ข้อมูลไม่ขึ้น

```
หน้าสินค้าดึงข้อมูลจาก Supabase มาแสดงไม่ได้ (ว่างเปล่า)
ช่วยไล่ตรวจให้ทีละอย่าง: ค่า URL/key ถูกไหม ชื่อตารางตรงไหม
สิทธิ์ (RLS/policy) เปิดให้อ่านหรือยัง และมี error ใน console ไหม
เจออะไรให้บอกวิธีแก้ทีละขั้น
```

? กรอก PIN ถูกแล้วแต่เข้าหน้า admin ไม่ได้

ให้ Agent ไล่ตรวจ Modal และการตรวจ PIN ทีละจุด:

```
PROMPT · PIN ไม่ผ่าน

ฉันกรอก PIN ในหน้า admin ถูกแล้วแต่ยังเข้าไม่ได้ (หรือ Modal ไม่ยอมปิด)
ช่วยไล่ตรวจว่าโค้ดเทียบ PIN ถูกต้องไหม เก็บค่า PIN ไว้ที่ไหน
และมี error ใน console หรือเปล่า เจออะไรให้บอกวิธีแก้ทีละขั้น
```

? Deploy ขึ้น Cloudflare แล้ว build ไม่ผ่าน

คัดลอกข้อความ error สีแดงจากหน้า Cloudflare มาแนบไปกับ prompt นี้:

```
PROMPT · build error

Deploy บน Cloudflare Pages แล้ว build ไม่ผ่าน
นี่คือ log ที่ขึ้น: [วาง error ทั้งหมดตรงนี้]
ช่วยอ่านและบอกสาเหตุกับวิธีแก้แบบทีละขั้น ภาษาไทย
```

? ไม่รู้จะสั่ง Agent ยังไงให้ได้งานตรงใจ

สูตรง่าย ๆ: บอก (1) สิ่งที่อยากได้ (2) บริบท (3) รูปแบบผลลัพธ์ แล้วขอให้ทำทีละขั้น ถ้ายังไม่ตรงให้บอก “ยังไม่ใช่ตรงนี้...” แล้วชี้จุดที่อยากแก้