



คอร์สการตลาดออนไลน์

สำหรับเจ้าของธุรกิจ · 2026

คู่มือ Claude Code สำหรับคนไม่เขียนโค้ด

ฉบับเริ่มต้น

รู้ให้ครบก่อนเริ่ม — มันทำอะไรได้ · ติดตังยังไ
ราคาเท่าไร · สั่งงานยังงให้ได้ผล

✓ ราคาแฟกเกจจริง

✓ วิธีติดตังทีละขั้น

✓ Desktop / เว็บ / CLI ต่างกันยังง

✓ เคล็ดลับสั่งงานที่ใช้ได้จริง

คู่มือนี้ไม่ได้สอนให้คุณเป็นโปรแกรมเมอร์
แต่สอนให้คุณ **สั่งงาน AI ให้สร้างเครื่องมือธุรกิจ** ที่คุณอยากได้
ด้วยภาษาคอนธรรมดา ไม่ใช่ภาษาโปรแกรมเมอร์



อ.เบิร์ด Pakorn.in.th

Google Premier Partner, Facebook Certified Partner, LINE Certified Coach
สอนผู้ประกอบการมาแล้ว 7,000+ ธุรกิจ

สารบัญ

1 Claude Code คืออะไร (ในภาษาคนธุรกิจ)

ทำไมมันไม่ใช่ ChatGPT อีกตัวหนึ่ง

2 เจ้าของธุรกิจใช้ทำอะไรได้บ้าง

6 หมวด · ตัวอย่างที่ใช้ได้จริงกับ SME ไทย

3 พื้นฐานที่ต้องรู้ก่อนใช้

เลือกโมเดล · effort · permission · context · usage limit

4 Desktop / เว็บ / CLI ต่างกันยังไง + วิธีติดตั้ง

เลือกให้ถูกแบบ แล้วลงเสร็จใน 1 นาที

5 ราคาแพ็คเกจ — ต้องจ่ายเท่าไร

Free / Pro / Max · ราคาจริง · แพ็คเกจไหนที่ผมแนะนำ

6 เคล็ดลับงานให้ได้ผล

6 กฎที่ทำให้ AI ทำงานออกมาใช้ได้จริง

7 Do / Don't + ความปลอดภัย

กฎเหล็กเรื่องรหัสลับและเงิน

8 ถ้าวัดต่อไป

จากอ่านจบ → มีเครื่องมือใช้จริง

8

บท อ่านจบใน 1 ชั่วโมง

1 นาที

เวลาติดตั้งจริง

\$0

เริ่มลองได้ ไม่ต้องจ่ายก่อน

อ่านคู่มือนี้ยังไงให้คุ้ม: ถ้ามีเวลาน้อย อ่าน **บทที่ 1, 5 และ 6** ก่อน — สามบทนี้คือหัวใจ (มันคืออะไร · จ่ายเท่าไร · สั่งยังไงให้ได้ผล) ที่เหลือค่อยกลับมาเปิดดูเป็นคู่มืออ้างอิงตอนใช้งานจริง

Claude Code คืออะไร

ในภาษาคนธุรกิจ — ไม่ใช่ภาษาโปรแกรมเมอร์

พูดสั้นที่สุด: **Claude Code คือ AI ที่ “ลงมือทำ” ไม่ใช่แค่ “ตอบ”**

ถ้าคุณเคยใช้ **ChatGPT / Gemini** คุณจะคุ้นกับการ *ถาม แล้วได้คำตอบมาเป็นข้อความ* — จากนั้นคุณต้องเอาคำตอบนั้นไปทำอะไรต่อเอง

Claude Code ต่างออกไป มันจะ **สร้างไฟล์จริง แก้ไฟล์จริง และรันโปรแกรมจริง** ให้คุณดูตรงหน้า คุณสั่งเป็นภาษาคนธรรมดา มันลงมือทำให้ — และใช้ได้หลายที่ ไม่ได้ผูกกับเครื่องเดียว: **บนเครื่องคอม** (Windows / Mac) · **บนเว็บไซต์** · **บนเบราว์เซอร์** หรือแม้แต่สั่งต่อจากมือถือ

ChatGPT / Gemini

- คุณได้ **ข้อความ** คำแนะนำ
- คุณ เอาไปทำต่อเอง
- ผลลัพธ์ = **รู้เพิ่ม**
- เหมาะกับ ถาม-ตอบ ร่างข้อความ

VS

Claude Code

- คุณได้ **ไฟล์ / โปรแกรมที่รันได้**
- มันทำให้เลย บนเครื่องคุณ
- ผลลัพธ์ = **มีเครื่องมือใช้ต่อได้ตลอด**
- เหมาะกับ สร้างระบบ ทำงานซ้ำอัตโนมัติ

“ChatGPT / Gemini ให้ **คำตอบ**
Claude Code ให้ **เครื่องมือ**”

เปรียบเทียบแบบบ้านๆ: เหมือนคุณจ้างช่างมาต่อเติมบ้าน — คุณไม่ต้องรู้วิธีผสมปูน คุณแค่บอกว่า “อยากได้ห้องเก็บของตรงนี้ กว้างประมาณนี้” ช่างจัดการให้
Claude Code คือช่างคนนั้น แต่เป็นช่างซอฟต์แวร์ ที่ฟังภาษาคนและทำงานให้ทันที

แล้วเราต้องอ่านโค้ดเป็นหรือเปล่า

มันเขียนโค้ดครับ — **แต่คุณไม่ต้องอ่านออก** เหมือนคุณไม่ต้องอ่านแบบแปลนเป็นก็สั่งช่างสร้างบ้านได้ สิ่งที่คุณต้องทำคือ **บอกให้ชัดว่าอยากได้อะไร** แล้ว **ดูผลลัพธ์ด้วยตาว่าใช่ไหม**

ความจริงที่ต้องรู้: Claude Code ไม่ใช่เวทมนตร์ มันจะทำได้ดีมากถ้าคุณสั่งชัดและทำทีละขั้น แต่ถ้าสั่งกว้างๆ ว่า “ทำเว็บไซต์ให้หน่อย” ผลลัพธ์จะออกมาไม่ตรงใจ — **บทที่ 6 คือวิธีสั่งให้ได้ผล**

เจ้าของธุรกิจใช้ทำอะไรได้บ้าง

6 หมวด · ตัวอย่างที่ใช้ได้จริงกับ SME ไทย

02

คนส่วนใหญ่คิดว่า “ของโปรแกรมเมอร์” — จริงๆ แล้วงานที่คุ้มค่าที่สุดคืองานธุรกิจซ้ำๆ ที่กินเวลาคุณทุกวัน



การตลาด & คอนเทนต์

- ระบบโพสต์ Facebook / IG / LINE อัตโนมัติ — AI เขียนแคปชั่น + ทำรูป + ตั้งเวลาโพสต์
- เจนคอนเทนต์ทั้งเดือนรวดเร็ว 30 โพสต์
- ทำหน้าขาย / landing page เอง ไม่ต้องจ้าง
- ทำของแจก (เช็คลิสต์ / อีบุ๊ก) เก็บรายชื่อลูกค้า
- ปรับบทความให้ติด Google และ AI search



ดูแลลูกค้า & ขาย

- บอกร่างคำตอบลูกค้าใน LINE / เพจ จากข้อมูลร้านเราเอง
- ระบบตอบคำถามที่พบบ่อย ดึงจากเอกสารบริษัท
- สรุปและจัดหมวดหมู่ลูกค้าที่ค้างเป็นร้อย
- ทำใบเสนอราคา / ใบแจ้งหนี้อัตโนมัติ



ข้อมูล & รายงาน

- ล้างและรวมไฟล์ Excel / CSV ที่รกให้เป็นระเบียบ
- ทำรายงานยอดขาย / สต็อก อัตโนมัติ
- ดึงราคาคู่แข่งหรือข่าวจากเว็บมาสรุปให้
- แปลง PDF หรือรูปใบเสร็จ ให้กลายเป็นตารางข้อมูล



เครื่องมือภายใน

- สร้างแอปเล็กๆ ใช้ในร้าน — คำนวณต้นทุน/ราคา, จอจิว, เช็คสต็อก
- Dashboard สรุปตัวเลขธุรกิจในหน้าเดียว
- ตั้งให้รันเองทุกวัน เช่น ส่งสรุปยอดขาย LINE ทุกเช้า



วิเคราะห์ & Research

- อ่านรีวิว/คอมเมนต์ลูกค้าเป็นพัน สรุปว่าคนชอบอะไร บ่นอะไร
- research คู่แข่งและตลาด ออกมาเป็นตารางเทียบ
- สรุปสัญญาหรือเอกสาร 50 หน้า ในไม่กี่นาที
- ทำ persona ลูกค้าจากข้อมูลจริง ไม่ใช่เดา



งานประจำวัน

- ร่างอีเมล / เอกสาร / โพสต์
- แปลภาษาโดยไม่มีข้อผิดพลาด
- สรุปคลิปหรือมีตติ้งเป็นข้อๆ
- ทำรูป / thumbnail / ฝาพื้นหลัง

เลือกงานแรกยังงี้: หยิบงานที่คุณ **ทำซ้ำทุกสัปดาห์ + น่าเบื่อ + ไม่ต้องใช้วิจารณญาณมาก** — นั่นคืองานที่คุ้มค่าที่สุดที่จะให้ AI ทำแทน

พื้นฐานที่ต้องรู้ก่อนใช้

7 เรื่องที่ไม่รู้แล้ว “เจ็บตัว” — เสียเงินเกิน งานหาย หรือใช้ไม่คุ้ม

1 เลือกโมเดล — จ่ายให้ตรงกับงาน

Claude Code เลือก “สมอง” ที่จะใช้ได้ สิ่งด้วยคำสั่ง `/model` แล้วเลือกจากรายการ — แต้่งยาว = ทั้งเก่งกว่าและแพงกว่า

opus	คิดหนัก · ตัดสินใจซับซ้อน · แก้ปัญหายาก — แพงสุด
sonnet	งานประจำวันทั่วไป — สมดุลที่สุด
haiku	งานง่ายๆ ซ้ำๆ — เร็วและถูก

ทำไมเจ้าของธุรกิจต้องสนใจ: เลือกโมเดลให้ตรงงาน = ใช้โควตาคุ้มขึ้นทันที อย่าใช้ตัวแพงสุดกับงานพิมพ์แคปชั่นสั้นๆ · ถ้าไม่ยากลเลือกเอง เลือก **default** ให้ระบบจัดให้ตามแพลนที่คุณใช้

2 Effort — สั่งให้ “คิดลึก” แค่ไหน

สั่งด้วย `/effort` — ยิ่งสูง AI ยิ่งใช้เวลาคิดในหัวมากขึ้น แก้ปัญหายากได้ดีขึ้น แต่**กินโควตามากขึ้น**ตาม

low งานง่าย	medium ปานกลาง	high ★ ค่าเริ่มต้น	xhigh ตอบติด	max ติดหนัก
----------------	-------------------	-----------------------	-----------------	----------------

ใช้อย่างไรให้คุ้ม: งานง่ายใช้ **low** · งานทั่วไปปล่อย **high** ตามค่าเริ่มต้น · ใช้ **xhigh / max** เฉพาะตอน**ติดจริงๆ** แก้ไม่ได้สักที

3 Permission mode — เบรกก่อนคันเร่ง

คุณว่า AI ทำอะไรได้เองบ้างโดยไม่ต้องถามคุณ — กด `Shift+Tab` เพื่อสลับโหมด

● default ★ เริ่มต้น	อ่านอย่างเดียว — จะแก้ไขต้องขออนุญาตคุณทุกครั้ง
● plan	วางแผนให้ดูก่อน ยังไม่ลงมือ — ดีตอนสำรวจงานใหญ่
● acceptEdits	แก้ไขไฟล์ได้เอง (คุณไปไล่ดูทีหลัง) — เร็วขึ้นเยอะ
● auto	ทำได้เกือบทุกอย่าง มีระบบตรวจสอบความปลอดภัยคอยกัน
● bypassPermissions ⚠️ อย่าใช้	ทำได้ทุกอย่างไม่ถามเลย — ใช้เฉพาะเครื่องทดลองที่พังได้เท่านั้น

คำแนะนำ: เริ่มที่ **default** ก่อนเสมอ พอเริ่มไวใจค่อยขยับเป็น **acceptEdits** — อย่ากระโดดไป **bypassPermissions** เพราะรำคาญคำถาม

ความจำ · คำสั่ง · โควตา

4 เรื่องที่เหลือ — รู้แล้วใช้คุ้มขึ้นทันที

4 Context window — “ความจำ” ของ AI มีขีดจำกัด

AI จำบทสนทนาได้จำกัด (ราว **200,000 tokens** — บางโมเดลถึง 1 ล้าน) พอคุยยาวมากๆ ความจำจะเต็ม

คุยไปแล้ว — AI ยังจำได้ครบ

ใกล้เต็ม → ย่ออัตโนมัติ

เริ่มคุย

เช็คด้วย `/context`

เต็ม 200k

ข่าวดีคือ มันย่อความจำให้เองอัตโนมัติ เมื่อใกล้เต็ม (auto-compact) คุณไม่ต้องทำอะไร

สังเกตอาการ: ถ้า AI เริ่ม “ลืม” เรื่องที่คุยไว้ตอนต้น = ความจำโดนย่อไปแล้ว ไม่ใช่มันจืดจาง — เป็นเรื่องปกติ

5 `/compact` กับ `/clear` — ต่างกันยังไง

`/compact`

- ย่อบทสนทนาเก่าให้สั้นลง แต่ยังจำสรุปไว้
- ใช้ตอน: คุยยาวแล้ว อยากทำงานเรื่องเดิมต่อ
- ไม่ลืมสิ่งที่ตกลงกันไว้

≠

`/clear`

- ล้างทิ้ง **เริ่มใหม่หมด** (ของเก่ากู้ได้ด้วย `/resume`)
- ใช้ตอน: เปลี่ยนไปทำงานคนละเรื่องเลย
- เริ่มสะอาด ไม่มีเรื่องเก่าปน

เคล็ดลับโควตาให้คุ้ม: เปลี่ยนงานคนละเรื่อง ให้ `/clear` — ไม่จิ้นคุณ “ลากความจำเรื่องเก่า” ไปด้วยทุกข้อความ เปลืองโควตาฟรีๆ

6 `/remote-control` — สั่งงานจากมือถือได้

ของดีที่คนไม่ค่อยรู้: คุณเริ่มงานบนคอมที่ออฟฟิศ แล้วเปิดดู/สั่งต่อจากมือถือหรือเบราว์เซอร์เครื่องอื่นได้

ข้อจำกัดสำคัญ: งานยังรันอยู่บนเครื่องคุณ — ถ้าปิดคอมหรือปิดหน้าต่าง งานหยุดทันที · และต้องใช้แพน Pro/Max ขึ้นไป

7 Usage limit — โควตาไม่ได้นับเป็น “รายวัน”

คนเข้าใจผิดกันเยอะ ความจริงคือโควตานับเป็น **ช่วง 5 ชั่วโมงแบบวิ่งไปเรื่อยๆ (rolling)** บวกกับโควตารายสัปดาห์ — และใช้ร่วมกันระหว่าง Claude แชน กับ Claude Code (ถ้าเดียวกัน)

- เช็คค่าเหลือเท่าไรด้วยคำสั่ง `/usage`
- ถ้าชนลิมิต = รอรอบรีเซต หรือซื้อ **usage credits** เพิ่ม (จ่ายตามใช้)
- ปี 2026 Anthropic **เพิ่มลิมิตรอบ 5 ชั่วโมงให้เป็น 2 เท่า** และเลิกลดโควตาช่วงคนใช้เยอะแล้ว

วางแผนงานให้เป็น: อย่าเผางานหนักรวดเดียวจนโควตาสัปดาห์หมดกลางสัปดาห์ — ถ้ามีงานใหญ่ แบ่งทำเป็นรอบ

Desktop / เว็บ / CLI ต่างกันยังไง

ต่างกันตรงที่ “งานไปรันที่ไหน” — จำแค่นี้พอ



Desktop / CLI = รันบนเครื่องคุณ

- เข้าถึงไฟล์ในเครื่องคุณได้เต็มที่
- ทำงานกับข้อมูลธุรกิจจริงได้ (Excel, รูป, ข้อมูลลูกค้า)
- ปิดคอม = งานหยุด

VS



เว็บ = รันบนเครื่อง Anthropic

- ไม่ต้องลงอะไรในเครื่อง — เปิด claude.ai/code
- แตะไฟล์ในเครื่องคุณไม่ได้ (ต้องผ่าน GitHub)
- ปิดคอมได้ งานยังรันต่อ

“บนเครื่องคุณ” มี 3 แบบย่อย เลือกอันไหนดี



Desktop app

★ แนะนำสำหรับ SME

ไม่ต้องแตะ Terminal · เห็นก่อนว่าจะแก้ไขอะไร แล้วค่อยกดรับ



CLI

พิมพ์คำสั่งใน Terminal
สายเทค · ทำสคริปต์ · ระบบอัตโนมัติ



ส่วนขยาย IDE

VS Code / JetBrains
คนที่ทำงานในโปรแกรมเขียนโค้ดอยู่แล้ว

สำหรับเจ้าของธุรกิจ ผมแนะนำ “Desktop app” — เพราะงานคุณคือไฟล์ Excel ไฟล์รูป ข้อมูลลูกค้า ที่อยู่ในเครื่องคุณ · แคมไม่ต้องพิมพ์คำสั่งใน Terminal ให้อีก

วิธีติดตั้ง — ทางที่ 1: Desktop app

ข่าวดีที่คนเข้าใจผิดกันเยอะ: ไม่ต้องลง Node.js หรืออะไรก่อนเลย — Claude Code เป็นโปรแกรมสำเร็จรูป ลงแล้วใช้ได้เลย ใช้เวลาจริงประมาณ 1 นาที

1

โหลดตัวติดตั้ง

มีทั้ง Windows และ Mac — โหลดที่ claude.com/download

2

ดับเบิลคลิกติดตั้ง

เหมือนลงโปรแกรมทั่วไป ไม่มีอะไรพิเศษ

3

เปิดโปรแกรม → ล็อกอินด้วยบัญชี Claude

เลือก “Sign in with claude.ai” (ไม่ใช่ API key) — ระบบเช็คแพนให้อัตโนมัติ

4

เข้าแท็บ Code → เลือก Local → เลือกโฟลเดอร์งาน

ชี้ไปที่โฟลเดอร์ที่คุณอยากให้มันทำงานด้วย

5

พิมพ์สั่งงานได้เลย

ไม่ต้องแตะ Terminal สักครั้ง · ล็อกอินครั้งเดียวจบ ไม่ต้องทำอีก

ติดตั้งแบบ CLI + งานแรกที่คุณควรรลอง

ถ้าอยากใช้แบบพิมพ์คำสั่ง — 1 บรรทัดจบ

ทางที่ 2 — CLI (พิมพ์คำสั่ง)

เครื่อง	เปิดโปรแกรมอะไร แล้วพิมพ์คำสั่งนี้
 Windows	เปิด PowerShell (ไม่ใช่ CMD) แล้ววาง: <code>irm https://claude.ai/install.ps1 iex</code>
 Mac	เปิด Terminal แล้ววาง: <code>curl -fsSL https://claude.ai/install.sh bash</code>

เสร็จแล้วเปิดไฟล์เดสก์ทอปของคุณ แล้วพิมพ์ `claude` — เท่านั้น

ติดตั้งไหน? พิมพ์ `claude doctor` — มันจะตรวจให้ว่าอะไรผิด แล้วบอกวิธีแก้

งานแรกที่คุณควรรลอง (อย่าเริ่มจากของใหญ่)

อย่าเปิดมาสั่งว่า “ทำระบบขายของให้หน่อย” — ผลลัพธ์จะพัง เริ่มจากของเล็กที่เห็นผลใน 10 นาที เช่น

ลองสั่งแบบนี้

“ช่วยรวมไฟล์ Excel 3 ไฟล์นี้เป็นไฟล์เดียว แล้วสรุปยอดรวมต่อเดือนให้หน่อย”

หรือแบบนี้

“ทำหน้าเว็บง่ายๆ หน้าเดียว โชว์ชื่อร้านกับเมนู แล้วเปิดให้ดูในเบราว์เซอร์”

หรือแบบนี้

“อ่านรีวิวลูกค้าในไฟล์นี้ แล้วสรุปว่าคนบ่นเรื่องอะไรมากที่สุด 5 อันดับ”

กฎเหล็กของงานแรก: ให้มันทำ **ทีละอย่าง** และ **เห็นผลได้ด้วยตา** — ความมั่นใจของคุณมาจากการเห็นของจริงทำงาน ไม่ใช่การอ่านโค้ด

เป้าหมายวันแรก ไม่ใช่ “ทำระบบใหญ่ให้เสร็จ”

แต่คือ **เห็นของจริงทำงาน 1 ชิ้น**

ราคาแพ็คเกจ ต้องจ่ายเท่าไร

ราคาจริงจากเว็บ Claude · กรกฎาคม 2026

ข้อสำคัญที่สุดก่อนเลย: **แพ็คเกจฟรีใช้ Claude Code ไม่ได้** — ต้องเริ่มที่ Pro ขึ้นไป

Free	Pro	Max 5x	Max 20x
\$0	\$20/เดือน	\$100/เดือน	\$200/เดือน
ฟรี	≈ 720 บาท/เดือน	≈ 3,600 บาท/เดือน	≈ 7,200 บาท/เดือน
✗ ใช้ Claude Code ไม่ได้	✓ ใช้ Claude Code ได้	✓ ใช้ Claude Code ได้	✓ ใช้ Claude Code ได้
- ใช้ได้แค่แชท Claude บนเว็บ - ไว้ลองคุยดูก่อนว่าชอบไหม	- ถูกสุดที่ใช้ Claude Code ได้ - ไว้ลองให้รู้ว่ามันทำอะไรได้ - ทำงานจริงจัง ชนลิมิตเร็วมาก	- โควตา 5 เท่าของ Pro ต่อรอบ 5 ชม. - ทำงานได้ยาวๆ ไม่ต้องนั่งรอโควตา - จุดที่ผมว่า “ทำงานได้จริง” เริ่มที่นี่	- โควตา 20 เท่าของ Pro ต่อรอบ 5 ชม. - สำหรับคนสร้างระบบจริงจังทั้งวันทุกวัน

ทำไมผมถึงแนะนำ Max 5x ไม่ใช่ Pro: \$20 มันใช้ได้จริงครับ — แต่พอคุณเริ่มสร้างของจริง คุณจะ**ชนลิมิต**กลางทางแล้วต้องนั่งรอเป็นชั่วโมง · จังหวะที่กำลังสนุกแล้วโดนตัดบทนี้แหละที่ทำให้คนล้มเลิก · ส่วนต่าง 2,900 บาท แลกกับการทำงานได้ไม่สะดุด ผมว่าคุ้มกว่ามาก

ถ้าบงจำกัดจริงๆ: เริ่ม Pro \$20 ก่อนก็ได้ — แต่รู้ไว้ว่าคุณจะเจอลิมิตเร็ว และการอัปเป็น Max ทีหลังทำได้ตลอด ไม่เสียอะไร

ทุกแพ็คเกจที่ใช้ Claude Code ได้ ใช้ “ถังโควตาเดียวกัน” — ไม่ว่าคุณจะใช้ผ่าน Desktop, เว็บ, CLI หรือมือถือ ก็ดึงจากโควตาก้อนเดียวกันของแพลนคุณ ไม่ต้องซื้อแยก

โควตาต่างกันแค่ไหน (ต่อรอบ 5 ชั่วโมง)

Pro	1 เท่า	\$20/เดือน
Max 5x	5 เท่า	\$100/เดือน
Max 20x	20 เท่า	\$200/เดือน

จ่ายเพิ่ม 5 เท่า (\$20→\$100) ได้โควตาเพิ่ม 5 เท่า — แต่จ่ายเพิ่มอีกเท่าตัว (\$100→\$200) ได้เพิ่มอีก 4 เท่า · **Max 5x คือจุดที่คุ้มที่สุด** สำหรับคนส่วนใหญ่

เช็คราคาล่าสุด + สมัครที่ claude.com/pricing

จ่ายแล้วได้อะไร และไม่ได้อะไร

เส้นแบ่งที่คนสับสนที่สุด — ค่าแพ็คเกจ ≠ ค่า API

รวมอยู่ในค่าแพ็คเกจแล้ว

- ✓ ใช้ Claude Code **ไม่คิดเงินเพิ่ม** รายครั้ง
- ✓ ใช้ได้ทุกโมเดล (opus / sonnet / haiku)
- ✓ ใช้ได้ทั้ง Desktop, เว็บ, CLI, มือถือ — **ถึงไควตาเดียวกัน**
- ✓ ชนลิมิตแล้วซื้อไควตาเพิ่มได้ (usage credits)

ไม่ได้รวม — จ่ายแยก

- ✗ ระบบที่คุณสร้าง เรียก **AI ผ่าน API** เองตอนรันจริง
- ✗ ค่าทำรูปด้วย AI ของเจ้าอื่น (เช่น OpenAI)
- ✗ ค่าโฮสต์เว็บ / เซิร์ฟเวอร์ ถ้าเอาระบบขึ้นออนไลน์

2 ก้อนนี้คนสับสนกันมากที่สุด

ค่าแพ็คเกจ (เจอแน่ตั้งแต่วันแรก)

- = ค่า “จ้าง AI มานั่งทำงานให้คุณ”
- จ่ายเดือนละครั้ง **จบ** ไม่มีบิลออก
- ตอนคุณนั่งสั่งงานมันอยู่ = ใช้ก้อนนี้

≠

ค่า API (เจอตอนสร้างระบบจริง)

- = ค่า “ระบบของคุณเรียก AI เอง”
- คิดตามใช้จริง ต้องตั้งเพดานเอง
- ตอนหัดใช้ **ยังไม่เจอก้อนนี้**

แล้วผมควรซื้อแพ็คเกจไหน

1

จะใช้ทำงานจริง → Max 5x \$100 (ผมแนะนำ)

ทำงานได้ต่อเนื่องไม่สะดุด · เป็นจุดที่คนส่วนใหญ่ “ใช้มันได้จริง” ไม่ใช่แค่ลองเล่น

2

งบจำกัด อยากชิมलगก่อน → Pro \$20

ใช้ได้ แต่เตรียมใจเรื่องชนลิมิต · อัปเป็น Max ที่หลังได้ตลอด

3

สร้างระบบทั้งวันทุกวัน → Max 20x \$200

สำหรับคนที่ใช้ Claude Code เป็นเครื่องมือทำมาหากินหลัก

อย่าจ่ายรายปีตั้งแต่วันแรก — ถึงแม้จะถูกกว่า แต่ต้องจ่ายรวดเดียวก้อนใหญ่ · ลองรายเดือนให้แน่ใจก่อนว่าคุณใช้มันจริงสัก 2-3 เดือน ค่อยเปลี่ยนไปรายปี

ราคาอ้างอิงจาก claude.com/pricing (ก.ค. 2026) · เงินบาทคิดที่ราว 36 บาท/ดอลลาร์ เป็นตัวเลขประมาณเพื่อให้เห็นภาพ · Anthropic ปรับราคาและไควตาเป็นระยะ เช็คของล่าสุดที่เว็บทางการก่อนสมัครทุกครั้ง

เคล็ดสั่งงานให้ได้ผล

บทที่สำคัญที่สุดในเล่ม — คนที่บอกว่า “ใช้ AI แล้วไม่เวิร์ค” ส่วนใหญ่พลาด 6 ข้อนี้

1 สั่งทีละขั้น อย่างรวบ

1 คำสั่ง = 1 ผลลัพธ์ที่เห็นได้ ถ้าสั่งรวบ 5 อย่างพร้อมกัน มันจะทำได้ครึ่งๆ กลางๆ แล้วคุณจะไม่ถูกจุด

✗ อย่างแบบนี้

“ทำเว็บโพสต์ FB พร้อม AI เขียนแคปชั่น ทำรูป แล้วตั้งเวลาโพสต์”

✓ สั่งแบบนี้แทน

“ทำหน้าเว็บเปล่าๆ โชว์ชื่อร้านก่อน แค่นี้พอ” → เห็นผล → ค่อยสั่งขั้นต่อไป

2 บอกให้ชัด และบอกว่า “อย่าทำอะไร”

AI ชอบเสนอของแถมที่คุณไม่ได้ขอ ปิดมันด้วยประโยคเดียว:

พูดตามนี้

“ทำแค่นี้พอ ยังไม่ต้องทำ feature อื่น ไม่ต้องเสนอเพิ่ม”

3 เชฟงานทุกจุดที่พอใจ

สั่งให้มันเชฟเวอร์ชันไว้ (git) พร้อมชื่อและคำอธิบาย พอทำพียงตอนหลังจะย้อนกลับมาจุดที่ดีได้

พูดตามนี้

“ช่วยเชฟเวอร์ชันตอนนี้ไว้ ระบุวันเวลาและคำอธิบายว่าตอนนี้ทำอะไรได้แล้วบ้าง”

4 ให้ลองก่อน ค่อยทำจริง

โดยเฉพาะงานที่ยังออกข้างนอก (โพสต์จริง ส่งอีเมลจริง จ่ายเงินจริง) สั่งให้ทำ “โหมดทดสอบ” ก่อนเสมอ

พูดตามนี้

“ก่อนทำระบบจริง ขอลองทำตัวอย่างให้ดูก่อน 1 อัน ถ้าโอเคผมจะบอกให้ทำต่อ”

5 เห็นผลด้วยตา ทุกขั้น

อย่าเชื่อคำว่า “เสร็จแล้วครับ” — สั่งให้มันเปิดให้ดู รันให้ดู หรือโชว์ตัวอย่างจริง คุณตัดสินใจด้วยตาตัวเอง

6 ค่อยยาวแล้วอย่าลืมจัดการความจำ

งานยาวๆ ใช้ `/compact` เพื่อย่อความจำ · เปลี่ยนเรื่องใหม่ใช้ `/clear` — ทั้งประหยัดโควตาและทำให้ AI ไม่สับสน

สรุปหัวใจ: ทำทีละขั้น · เห็นผลทุกขั้น · เชฟทุกจุดที่ดี

Do / Don't + ความปลอดภัย

เรื่องนี้ไม่มีใครสอน แต่พลาดแล้วเสียหายจริง

🔑 กฎเหล็กเรื่องรหัสลับ (API key / Token)

เวลาต่อ AI หรือต่อ Facebook คุณจะได้ “กุญแจ” มาชุดหนึ่ง — ใครได้กุญแจนี้ไป **ใช้เงินคุณได้ โฟสต์แทนพวกคุณได้**

ควรทำ

- ✓ เก็บไว้ในไฟล์ `.env` เท่านั้น
- ✓ ใส่ `.gitignore` กันหลุดขึ้น GitHub
- ✓ หลุดแล้ว **ยกเลิกทันที แล้วสร้างใหม่**

ห้ามทำ

- ✗ พิมพ์ลงเอกสารสาธารณะ
- ✗ ให้ขึ้นกล้องตอนอัดคลิป / แรจ้อ
- ✗ ส่งในไลน์กลุ่ม
- ✗ คิดว่า “คงไม่มีใครเห็น”

💰 กฎเหล็กเรื่องเงิน

- รู้ก่อนว่าปั๊มไหนกดแล้วเสียเงิน — ให้ AI เขียนกำกับไว้เลยว่ากดทีหนึ่งประมาณกี่บาท
- ตอนกด ใช้โมเดลถูก + effort ต่ำ ก่อน แล้วค่อยขยับ
- ถ้าระบบคุณเรียก AI ผ่าน API ตั้ง **เพดานค่าใช้จ่ายรายเดือน**ไว้เสมอ

🔒 กฎเหล็กเรื่องความปลอดภัยของงาน

ควรทำ

- ✓ เริ่มที่โหมด `default` ให้มันถามก่อนแก้
- ✓ เชฟ git ก่อนสั่งงานใหญ่
- ✓ ทำ “โหมดทดสอบ” ก่อนยิงของจริง
- ✓ อ่านสิ่งที่มันจะทำก่อนกดอนุญาต

ห้ามทำ

- ✗ ใช้ `bypassPermissions` บนเครื่องงานจริง
- ✗ ปลอมใจให้แก้ไฟล์สำคัญโดยไม่ได้เชฟไว้ก่อน
- ✗ ต่อระบบโฟสต์/ส่งอีเมลจริงตั้งแต่วันแรก
- ✗ กด “อนุญาต” รัวๆ เพราะรำคาญ

ประโยคที่ควรจำ: AI ทำงานเร็วมาก — ซึ่งแปลว่าถ้าคุณสั่งผิด มัน**ทำผิดได้เร็วมาก**เช่นกัน เสรก (permission mode + git + โหมดทดสอบ) จึงสำคัญกว่าคันทรง

ก้าวต่อไป

อ่านจบแล้วอย่าเพิ่งปิด — เปิดคอมแล้วลองสั่งงานแรกภายในวันนี้

เช็คลิสต์ 7 วันแรก

- วันที่ 1 — ติดตั้ง + งานแรก**
ลง Claude Code แล้วสั่งของเล็กๆ (ทำหน้าเว็บเปล่า 1 หน้า) ให้เห็นว่ามันทำงานจริง
- วันที่ 2 — หัดเซฟ**
ลองสั่งให้เซฟ git แล้วลองย้อนเวอร์ชันดู จะได้กลิ่นอายของ
- วันที่ 3 — เอางานจริงมาลอง**
หยิบงานที่ทำซ้ำๆ ทุกสัปดาห์มา 1 งาน ให้มันช่วยทำ
- วันที่ 4-5 — ต่อยอด**
ขยายงานนั้นทีละขั้น จนใช้งานได้จริงในธุรกิจ
- วันที่ 6 — ดูค่าใช้จ่าย**
ลอง `/model` · `/effort` · `/usage` ดูว่าโคตบาทหมดเร็วแค่ไหน
- วันที่ 7 — ทบทวน**
งานไหนที่ AI ทำแทนได้ถาวร → ทำให้เป็นระบบ แล้วเลิกทำเอง

“คนที่ได้เปรียบในยุคนี้
ไม่ใช่คนที่รู้จัก AI
แต่คือคนที่ใช้มันสร้างของได้”

อยากไปให้ไกลกว่านี้ — ลองเรียนฟรีก่อนได้

คู่มือเล่มนี้ทำให้คุณ **เริ่มได้** — ถ้าอยากได้ **ระบบทำคอนเทนต์อัตโนมัติของธุรกิจคุณเอง** ผมสอนจับมือทำทีละคลิป และเปิดให้ทดลองเรียนฟรี ไปดูก่อนได้ว่าถูกใจไหม

คอร์ส AI Content Automation ด้วย Claude Code

สอนตั้งแต่เว็บเปล่า → ระบบโพสต์อัตโนมัติของธุรกิจคุณเอง · ไม่ต้องเขียนโค้ดเป็น

 ทดลองเรียนฟรี + ดูรายละเอียดที่

pakorn.in.th/claude-code-ai-content-automation-101



อ.เบิร์ด Pakorn.in.th

Google Premier Partner, Facebook Certified Partner, LINE Certified Coach

ผู้ก่อตั้งเอเจนซี · วิทยากรรับเชิญ Google Thailand · สอนผู้ประกอบการมาแล้ว 7,000+ ธุรกิจ



คู่มือฉบับนี้แจกฟรี — ส่งต่อให้เพื่อนที่กำลังหาทางใช้ AI กับธุรกิจได้เลย · ข้อมูลอ้างอิงจากเอกสารทางการ code.claude.com/docs และ claude.com/pricing (ก.ค. 2026)