

Loop Engineering

เช็กลิสต์ ออกแบบลูป

เลิกนั่งพิมพ์สั่ง AI ทีละรอบ แล้วออกแบบระบบที่วนทำงานเอง
แบบมีเบรก มีตัวตรวจ และไม่เผาเงินทิ้ง



ลูปคืออะไร

แบบแชก

เราพิมพ์สั่ง → รอผล → อ่าน → สั่งต่อ
เราเป็นคนขับทุกรอบ ถ้าเราลูปไป งานก็หยุด

แบบลูป

เราออกแบบระบบครั้งเดียว ให้มันวนทำงานเอง
เราเปลี่ยนจากคนขับ เป็นคนวางกติกาและคน
ตรวจ

ลูปที่ดีวนครบ 5 ขั้นนี้ แล้วเริ่มรอบใหม่เอง:



6 ขั้นส่วน (สรุปจากบทความของ Addy Osmani)

ขั้นส่วน	คำหน้าที่	ตัวอย่างใน Claude Code
1. ตั้งเวลา Automations	วนมาทำงาน คัดว่าอะไรต้องทำ ตามรอบ เวลา	<code>/loop</code> , งานตั้งเวลาบนคลาวด์หรือ แอปเดสก์ท็อป
2. แยกพื้นที่ทำงาน Worktrees	ให้ AI หลายตัวทำงานพร้อมกันโดยไฟล์ไม่ ชนกัน	git worktree แยกโฟลเดอร์ต่อหนึ่งงาน
3. สกิล Skills	เก็บความรู้และขั้นตอนของโปรเจกต์ไม่ ต้องอธิบายใหม่ทุกรอบ	skills / CLAUDE.md
4. ต่อเครื่องมือ Plugins / MCP	ให้ลูปอ่านและเขียนระบบที่ใช้อยู่ เช่น GitHub, บอร์ดงาน	MCP server, plugins (ดู EP.2)
5. แยกตัวทำ-ตัวตรวจ Sub-agents	ตัวหนึ่งเขียน อีกตัวตรวจ ไม่ให้ตัวเดียว ตรวจงานตัวเอง	sub-agent สำหรับรีวิว / รันเทสต์
6. ไฟล์จดงาน State file	จดว่าทำอะไรไปแล้ว ค้างอะไร ไว้นอก แชท รอบหน้าอ่านต่อได้	PROGRESS.md, issue, บอร์ดงาน

เครื่องมืออื่นก็ทำแนวเดียวกันได้ เช่น ตั้ง cron หรือ GitHub Actions ให้เรียก AI agent แบบ CLI ตามรอบ หลักคิดเหมือนกัน

เช็กลิสต์ก่อนปล่อยลูปวิ่ง

1 เป้าหมายและจุดหยุด

- ☐ เขียนเป้าหมายเป็นประโยคเดียวที่วัดผลได้ เช่น "CI เชี่ยวและรีวิวคอมเมนต์ตอบครบ"
- ☐ บอกชัดเจนว่าเมื่อไหร่ "เสร็จ" และเมื่อไหร่ให้หยุดแล้วเรียกคน
- ☐ บอกขอบเขต: งานไหนห้ามแตะ หรือห้ามเริ่มงานใหม่นอกเรื่อง

2 เบรกและเพดาน

- ☐ ตั้งรอบเวลาให้เหมาะกับงาน (งานที่เปลี่ยนซ้ำ ไม่ต้องเช็กทุกนาที)
- ☐ มีวันหมดอายุ (ลูปตั้งรอบใน Claude Code หมดอายุเองใน 7 วัน)
- ☐ ตั้งบหรือดูการใช้โทเคนเป็นระยะ ลูปที่วนถี่ ใช้โทเคนทุกรอบ
- ☐ รู้วิธีหยุด: `Esc` สำหรับลูปที่เลือกจังหวะเอง หรือสั่งให้ยกเลิกสำหรับลูปตั้งรอบ

3 สิทธิ์

- ☐ คำสั่งที่ย้อนกลับไม่ได้ (push, ลบไฟล์, แตะฐานข้อมูลจริง) ต้องรอคนอนุมัติ
- ☐ ลูปไม่มีสิทธิ์ production และไม่เห็นความลับที่ไม่จำเป็น
- ☐ ใช้ worktree หรือ branch แยก ไม่ให้ลูปแก้ main ตรงๆ

4 การตรวจ

คำว่า "เสร็จแล้ว" จาก AI เป็นแค่คำบอก ยังไม่ใช่หลักฐาน

- ☐ ทุกรอบต้องมีหลักฐาน: เทสต์ผ่าน, build ผ่าน, ลิงก์ CI
- ☐ ให้ตัวตรวจแยกจากตัวทำ (sub-agent รีวิว หรือคนรีวิว)
- ☐ เราอ่านสรุปและ diff ก่อนรวมโค้ดทุกครั้ง

5 ความจำและความเข้าใจ

- ☐ มีไฟล์จดงาน (เช่น PROGRESS.md) ที่ลูปอ่านก่อนเริ่มและเขียนหลังจบทุกรอบ
- ☐ ให้ลูปอธิบายว่าทำอะไร เพราะอะไร ถ้าเราอธิบายต่อไม่ได้ อย่าเพ่งรวม

ตัวอย่างใช้ใน Claude Code

คำสั่ง /loop

```
# ทุก 5 นาที
/loop 5m check if the deployment finished and tell me what happened

# ไม่ใส่เวลา: Claude เลือกเอง (1 นาที ถึง 1 ชั่วโมง)
/loop check whether CI passed and address any review comments

# ไม่ใส่อะไรเลย: ใช้พรอมต์ดูผลงานในตัว หรือไฟล์ loop.md ถ้ามี
/loop
```

.claude/loop.md (พรอมต์ตั้งต้นของ /loop)

เป้าหมาย: ดูแล PR ของ branch ปัจจุบันให้พร้อมรวม

ทบทวน:

1. อ่าน PROGRESS.md ก่อน
2. ถ้า CI แดง: ดู log หาสาเหตุ แก้ไขน้อยที่สุด แล้วรันเทสในเครื่อง
3. ถ้ามีรีวิวคอมเมนต์ใหม่: แก้ที่ละข้อ แล้วตอบว่าแก้อะไร
4. ถ้าทุกอย่างเขียวและเจียบ: ตอบบรรทัดเดียวว่า "เรียบร้อย"
5. จัดสิ่งที่ทำ และสิ่งที่ค้าง ลง PROGRESS.md

ห้าม: push ขึ้น main, ลบไฟล์, และฐานข้อมูลจริง, เริ่มงานใหม่นอกเรื่อง
ถ้าแก้ปัญหาค้างไม่ผ่าน 2 รอบติด ให้หยุดแล้วสรุปให้ฉันตัดสินใจ

PROGRESS.md (ไฟล์งาน)

```
# PROGRESS
## เป้าหมาย
PR #42 พร้อมรวม: CI เขียว + รีวิวครบ

## ทำแล้ว
- [x] แก้เทส login ที่พัง (รอบ 3)

## ค้าง / ต้องให้คนตัดสินใจ
- [ ] รีวิวขอเปลี่ยนชื่อ API (กระทบแอปมือถือ)
```

ไฟล์ `.claude/loop.md` ใช้ในโปรเจกต์นั้น ส่วน `~/.claude/loop.md` ใช้ทุกโปรเจกต์ที่ไม่มีไฟล์ของตัวเอง แก้ไฟล์ได้ระหว่างลูปวิ่ง มีผลรอบถัดไป ลูปแบบนี้ทำงานเฉพาะตอนเปิด Claude Code ค้างไว้ ถ้าอยากให้วิ่งตอนปิดเครื่อง ใช้งานตั้งเวลาบนคลาวด์แทน

ที่มาของข้อมูลในคลิป

- คำว่า Loop Engineering และ 6 ชั้นส่วน: บทความ "Loop Engineering" ของ Addy Osmani (addyosmani.com, 7 มิ.ย. 2026)
- คำพูด "My job is to write loops." (Boris Cherny หัวหน้าทีม Claude Code) และ "designing loops that prompt your agents" (Peter Steinberger คนสร้าง OpenClaw): อ้างอิงที่บทความข้างบนยกมา
- ความเสี่ยง: "เสร็จ" ไม่ใช่หลักฐาน, โค้ดออกเร็วกว่าที่เข้าใจ, เชื้อลูปจนเลิกคิดเอง, ค่าโทเคน (จากบทความเดียวกัน)
- คำสั่ง /loop, loop.md และการหมดอายุ 7 วัน: เอกสาร Claude Code หน้า "Run prompts on a schedule" (code.claude.com/docs)

ลูปแรกของคุณจะให้มันทำอะไร? เล่าให้ฟังหน่อย

ติดตาม **keaykub.dev** · เกดยากๆ เล่าให้ง่ายใน 1 นาที

เอกสารนี้แจกฟรี แชร์ต่อได้ ข้อมูล ณ ต.ค. 2026 · פיเจอร์ของเครื่องมืออาจเปลี่ยนได้ เช็คเอกสารล่าสุดก่อนใช้