



Build AI Multi-Agent with Claude Code

สร้างงานทีม AI Agent ใน Claude Code ตั้งแต่ Subagent ถึง Orchestration และ Guardrail

- 12 ชั่วโมง
- 2 วัน
- พื้นฐาน ใช้ Claude Code หรือ AI Coding Tool อยู่แล้ว
- ระดับ **Advanced**



ระยะเวลา: 2 วัน

12 ชั่วโมง (2 วัน วันละ 6 ชม.) · 09:00-16:00 น.



รูปแบบ: Classroom / Hybrid

Onsite หรือ Online ผ่าน Microsoft Teams · Public / In-house



รายละเอียดหลักสูตร

www.itgenius.co.th

นักพัฒนาจำนวนมากใช้ Claude Code เป็นผู้ช่วยเขียนโค้ดแบบคุยทีละคำสั่งได้คล่องแล้ว แต่เมื่องานใหญ่ขึ้น Context ของ Agent ตัวเดียวเริ่มเต็ม งานที่ทำพร้อมกันได้ถูกทำทีละขั้น และผลลัพธ์ขึ้นกับว่าใครพิมพ์ Prompt เก่งแค่ไหน Claude Code ในปัจจุบันจึงมีทั้ง Subagent ที่แยก Context ของตนเอง Skills, MCP, Hooks และ Agent Teams ให้แบ่งงานเป็นทีม AI ที่มีหน้าที่ชัดเจน คำถามสำคัญจึงเป็นว่าจะออกแบบทีม Agent อย่างไรให้ทำงานเร็วขึ้น ผลลัพธ์ตรวจสอบได้ และไม่ใช้ Token หรือสิทธิ์เกินจำเป็น

หลักสูตรนี้พาผู้เรียนสร้างงานทีม AI Agent ใน Claude Code ตั้งแต่พื้นฐานการทำงานของ Agent การเตรียม settings.json การให้ Subagent วิเคราะห์โค้ดพร้อมกันแบบ Swarm การสร้าง Specialty Agent ด้วย Markdown และ YAML Frontmatter การผูก Skills และ MCP Server ให้ Agent แต่ละตัว ไปจนถึง Orchestrator-Worker ที่ Leader วิเคราะห์ก่อนแจกงาน Pipeline แบบทำตามลำดับ การจัดการ Context และ Memory และการตั้ง Permission กับ Hooks เป็น Quality Gate และ Cost Gate ปิดท้ายด้วย Capstone ที่ทีม Agent เฉพาะทางสร้างเว็บแอปธุรกิจจริงและ Deploy ขึ้น Vercel (อบรม 2 วัน วันละ 6 ชั่วโมง รวม 12 ชั่วโมง ระดับ Advanced)

เครื่องมือ

CC

Claude Code

SA

Subagents

SK

Skills

MCP

MCP

Hk

Hooks

ราคาค่าอบรม · PUBLIC / IN-HOUSE

8,900 บาท / ท่าน

ราคายังไม่รวมภาษีมูลค่าเพิ่ม (VAT 7%) · รวมเอกสารประกอบการอบรม Lab Guide ชุดไฟล์ Agent, Skill, Hook และ settings.json ตัวอย่าง และ Certificate · ผู้เรียนใช้บัญชี Claude ที่มี Claude Code และบัญชี Vercel ของตนเอง · รับจัด In-house ขอใบเสนอราคาได้

สมัครผ่านเว็บไซต์ ลด 10%



เพิ่มเพื่อน
LINE

01 วัตถุประสงค์ Objectives

- อธิบายการทำงานของ Agent, Tool Calling และเลือกรูปแบบ Multi-Agent ให้เหมาะกับงานและงบ Token
- ใช้ Subagent ในตัวทำงานขนานกันแบบ Swarm แล้วรวมผลเป็นรายงานเดียวได้
- สร้าง Specialty Agent ใน .claude/agents พร้อมกำหนด Tools, Model, Skills และ MCP Server
- เขียน Skills และติดตั้ง MCP Server ในขอบเขตที่เหมาะสม แล้วผูกให้ Agent เฉพาะตัว
- ออกแบบ Orchestrator-Worker, เขียน Delegation Prompt และติดตามงานผ่าน Task List
- ต่อ Agent เป็น Pipeline ตามลำดับ และจัดการ Context กับ Memory ข้าม Session
- ตั้ง Permission และ Hooks เป็น Quality Gate และ Cost Gate ก่อนให้ Agent ทำงานอัตโนมัติ
- ใช้ทีม Agent สร้างเว็บแอปธุรกิจจริงและ Deploy ขึ้น Vercel ตาม Pre-flight Checklist

02 กลุ่มเป้าหมาย Target Audience

- Software Developer และ Full-stack Developer ที่ใช้ Claude Code อยู่แล้วและอยากขยายเป็นทีม Agent
- Tech Lead และ Solution Architect ที่ต้องออกแบบ Workflow ของ AI Agent ให้ทีมใช้ร่วมกัน
- AI Engineer และ Automation Engineer ที่สร้างระบบ Agent ทำงานต่อเนื่องหลายขั้นตอน
- DevOps และ Platform Engineer ที่ดูแลสิทธิ์ Hooks และค่าใช้จ่ายของ Agent ในองค์กร
- ผู้ที่เรียน Claude Code Roadmap หรือใช้ AI Coding Tool มาแล้วและต้องการระดับ Multi-Agent

03 จุดเด่นของหลักสูตร Highlights

- เจาะ **Multi-Agent** เต็มสองวัน Subagent, Agent Teams และ Background
- สร้าง **Agent เฉพาะทางเอง** Frontmatter, Skills และ MCP ต่อ Agent
- **Orchestration ที่ตรวจได้** Leader-first, Task List และ Pipeline
- มี **Guardrail ทุกชั้น** Permission, Hooks, Quality และ Cost Gate
- **Capstone ถึง Production** ทีม Agent สร้างเว็บแอปและ Deploy

04 ความรู้พื้นฐาน Prerequisites

- ใช้ Claude Code หรือ AI Coding Tool อื่นมาแล้ว และเขียน Prompt ทำงานได้ชัดเจน
- ใช้ Terminal, Git และ GitHub ได้ และอ่านเขียนไฟล์ JSON พื้นฐานได้
- อ่านโค้ดเว็บแอปได้ เช่น JavaScript หรือ TypeScript เพื่อตรวจสอบงานที่ Agent สร้าง
- มีบัญชี Claude ในแผนที่มี Claude Code หรือ API Key ของตนเอง และบัญชี Vercel แบบฟรี
- โน้ตบุ๊กที่ติดตั้ง Node.js, VS Code และ Git ได้ พร้อมสิทธิ์ติดตั้งโปรแกรม

05 สิ่งที่ต้องเตรียมและเครื่องมือ Setup & Tools

รายการ	รายละเอียด
อุปกรณ์	Notebook เชื่อมต่ออินเทอร์เน็ต มีสิทธิ์ติดตั้งโปรแกรม พร้อม Node.js, Git และ VS Code
Claude Code	Claude Code CLI หรือแอป Desktop รุ่นล่าสุด พร้อม settings.json, .claude/agents และ .claude/skills
บัญชีที่ใช้	บัญชี Claude Pro, Max, Team หรือ Enterprise หรือ API Key ของผู้เรียน และบัญชี GitHub
MCP และ Deploy	MCP Server ผ่าน npx, Docker หรือ Remote URL และบัญชี Vercel แบบฟรีพร้อม Vercel CLI
โปรเจกต์	Repository ตัวอย่างสำหรับ Swarm และโจทย์เว็บแอปธุรกิจสำหรับ Capstone

หมายเหตุ: สถาบันไม่ได้จัดเตรียมบัญชีหรือ Subscription ของ Claude, GitHub หรือ Vercel ให้ ผู้เรียนใช้บัญชี Claude ส่วนตัวหรือของบริษัทในแผน Pro, Max, Team หรือ Enterprise หรือ API Key ของตนเอง (แผน Free ไม่มี Claude Code) ปริมาณการใช้งานขึ้นกับแผนนั้น และงานแบบ Multi-Agent ใช้ Token มากกว่า Session เดียวอย่างชัดเจน Agent Teams ยังเป็นฟีเจอร์ทดลองที่ต้องเปิดเอง ฟีเจอร์ของ Claude Code เปลี่ยนเร็ว เนื้อหาจะอิงเวอร์ชันล่าสุดในวันอบรม หลักสูตรนี้จะเจาะ Multi-Agent โดยเฉพาะ หากต้องการพื้นฐาน Claude Code แบบครบถ้วนดู Claude Code Roadmap

06 ภาพรวมการเรียนรู้ 2 วัน Big Picture

วัน	หัวข้อหลัก
Day 1 Section 1-6	เข้าใจว่า Agent ทำงานเป็นทีมอย่างไร, เตรียม Claude Code และ settings.json, ให้ Subagent วิเคราะห์โค้ดพร้อมกันแบบ Swarm, สร้าง Specialty Agent ใน .claude/agents, เขียน Skills และเชื่อม MCP Server ให้ Agent แต่ละตัว
Day 2 Section 7-12	วาง Orchestrator-Worker ที่ Leader วิเคราะห์ก่อนแจกงาน, คมกับ Frontend, Backend และ QA ผ่าน Task List, ต่อ Agent เป็น Pipeline, จัดการ Context และ Memory, ตั้ง Permission และ Hooks เป็น Quality Gate กับ Cost Gate แล้วสร้างเว็บแอปจริงและ Deploy ขึ้น Vercel

DAY
1

สร้างทีม Agent: Subagent, Skills และ MCP

เนื้อหาการอบรม · Section 1-6

รวม
6 ชั่วโมง

1 AI Agent ทำงานเป็นทีมได้อย่างไร

- Agent ต่างจาก Chatbot อย่างไร และ Tool Calling ทำให้ Claude Code อ่านไฟล์ แก้โค้ด และรับคำสั่งได้อย่างไร
- รูปแบบ Multi-Agent ใน Claude Code: Subagent, Background Subagent และ Agent Teams
- Context Window ของแต่ละ Agent และเหตุผลที่การแยก Context ช่วยให้งานใหญ่แม่นยำขึ้น
- ต้นทุน Token ของทีม Agent และเกณฑ์ตัดสินใจว่างานใดกลุ่มที่จะแบ่งให้หลาย Agent

2 Lab: เตรียมสภาพแวดล้อมสำหรับทีม Agent

- Terminal, VS Code, Git, Node.js และการอ่านเขียน JSON กับ Environment Variable
- ติดตั้ง Claude Code แบบ CLI หรือแอป Desktop แล้วตรวจสอบด้วย `claude --version` และ `claude doctor`
- `settings.json` ระดับผู้ใช้และโปรเจกต์ และค่าที่เกี่ยวข้องกับ Agent
- Lab: ตั้งโปรเจกต์ตัวอย่าง ตรวจสอบ `/agents`, `/context` และ `/usage` ก่อนเริ่ม

3 Workshop: Swarm ด้วย Subagent ทำงานขนานกัน

- Subagent ในตัว Explore, Plan และ general-purpose แต่ละตัวมี Context แยกของตนเอง
- ให้ Leader แบ่งงานแล้วส่ง Subagent หลายตัวทำพร้อมกัน เทียบกับการทำทีละขั้น
- ติดตาม Background Subagent ด้วย `/tasks` แล้วรวมผลเป็นข้อสรุปเดียวเมื่อผลขัดแย้งกัน
- Workshop: Prompt เดียวให้ Leader ส่ง Subagent วิเคราะห์ Codebase พร้อมกันแล้วรวมรายงาน

4 Lab: Specialty Agent ด้วย Markdown และ YAML

- `CLAUDE.md` เป็นกติกากลางของทีม ส่วนไฟล์ Agent อยู่ใน `.claude/agents` หรือ `~/.claude/agents`
- ให้ Claude สร้างไฟล์ Agent จาก Prompt แล้วปรับ name, description, tools และ model ต่อ Agent
- เรียกใช้ด้วย Auto-delegation, `@-mention` หรือ `claude --agent`
- Lab: สร้าง Researcher Agent ที่ค้นข้อมูลแล้วสรุปพร้อมแหล่งอ้างอิง

5 Skills ความรู้เฉพาะงานที่ Agent หยิบใช้ได้

- Skill คือไฟล์เดอร์ใน `.claude/skills` ที่มี `SKILL.md` พร้อมไฟล์ประกอบ
- โครงสร้าง `SKILL.md`: Frontmatter, description ที่ทำให้ถูกเรียกถูกงาน และคำสั่งทีละขั้น
- ตัวอย่างการใช้งาน และการ Preload Skill ให้ Subagent ผ่านฟิลด์ `skills`
- Lab: สร้าง Skill มาตรฐานรายงานแล้วผูกให้ Researcher Agent ใช้ทุกครั้ง

6 Lab: MCP Server สำหรับ Agent แต่ละตัว

- Transport แบบ stdio และ HTTP (SSE เลิกแนะนำแล้ว) และการติดตั้งผ่าน npx, Docker หรือ Remote URL
- ขอบเขต local, project ใน `.mcp.json` และ user ควรเลือกแบบใดเมื่อทำงานเป็นทีม
- ผูก MCP Server ให้เฉพาะ Subagent ด้วยฟิลด์ `mcpServers` และจำกัด Tool ที่ใช้ได้
- Lab: ให้ Researcher Agent ใช้ MCP ค้นข้อมูลภายนอกโดยไม่เปิดเผยให้ Agent อื่นใช้

DAY
2

Orchestration, Guardrails และ Capstone

เนื้อหาการอบรม · Section 7-12

รวม
6 ชั่วโมง

7 Orchestrator-Worker และ Leader-first

- Orchestrator-Worker: Leader วิเคราะห์โจทย์ก่อน แล้วจึงแจกงานให้ Worker
- Delegation Prompt: เป้าหมาย ขอบเขต ไฟล์ รูปแบบผลลัพธ์ และเกณฑ์ว่างานเสร็จ
- Effort Scaling: จำนวน Agent และระดับ Effort ที่พอดีกับงาน และ Task List ของ Leader
- Lab: ให้ Leader วิเคราะห์ Feature ใหม่แล้วแตกเป็นงานย่อยพร้อม Prompt ให้ Worker

8 Workshop: ทีม Frontend, Backend และ QA

- เปิด Agent Teams (ฟีเจอร์ทดลอง) และความแตกต่างจาก Subagent ที่รายงานกลับ Leader
- Team Lead สร้าง Shared Task List ให้ Teammate รับงาน ส่งข้อความหากัน และแยกไฟล์ที่แต่ละคนดูแล
- Workshop: ทีม Frontend, Backend และ QA สร้าง Feature เดียวกันผ่าน Task List

9 Workshop: Sequential Workflow แบบ Pipeline

- งานแบบใดต้องทำตามลำดับ และเมื่อใดการทำขนานกันกลับเสียมากกว่าได้
- ต่อ Subagent เป็น Pipeline ผ่านไฟล์ Spec หรือใช้ Task ที่ขึ้นต่อกันใน Agent Teams
- Workshop: Pipeline วิเคราะห์ความต้องการ ออกแบบ เขียนโค้ด และตรวจทาน

10 Context Management และ Persistent Memory

- CLAUDE.md ระดับองค์กร ผู้ใช้ โปรเจกต์ และ CLAUDE.local.md รวมถึง Auto Memory
- Memory ของ Subagent ระดับ user, project และ local สำหรับจำข้าม Session
- แยก Context ให้ Subagent ทำงานหนักแล้วส่งเฉพาะสรุปกลับ Leader และใช้ /compact กับ /clear
- Lab: ให้ Reviewer Agent จำข้อผิดพลาดที่พบบ่อยของโปรเจกต์ข้าม Session

11 Workshop: Permission, Hooks, Quality Gate และ Cost Gate

- permissions แบบ allow, ask และ deny ใน settings.json และกับ Agent อ่านไฟล์ .env หรือ Secret
- จำกัดสิทธิ์ต่อ Agent ด้วย tools, disallowedTools, permissionMode และ maxTurns
- Gate Logic: เช็คนโยบายหยุดงาน และจุดที่ต้องให้คนอนุมัติก่อนทำต่อ
- Hooks อย่าง PreToolUse, SubagentStop และ TaskCompleted ดักและบล็อกคำสั่งอันตราย
- Workshop: Quality Gate ที่ไม่ยอมปิดงานจนกว่า Test ผ่าน และ Cost Gate ที่คุมจำนวนรอบและงบ Token

12 Capstone: ทีม Agent สร้างเว็บแอปและ Deploy

- Leader วิเคราะห์โจทย์ธุรกิจและเขียน Spec กลางที่ทุก Agent ยึดเป็นข้อตกลง
- ทีม Planner, Competitive Research, Full-stack Developer, Writer และ QA/Reviewer
- ติดตามงานผ่าน Task List ใช้ Synthesize Pattern รวมผล และตรวจ Diff กับ Test ก่อนรับงาน
- Deploy ขึ้น Vercel ด้วย Vercel CLI หรือ Vercel MCP พร้อมตั้ง Environment Variable
- Pre-flight Checklist ก่อนรับ Agent อัตโนมัติ แล้วนำเสนอผลงานและแผนนำไปใช้จริง

07 เรียนจบแล้วได้อะไร Outcomes

- ออกแบบทีม **Agent** ได้ เลือกรูปแบบให้เหมาะกับงานและงบ
- สร้าง **Agent เฉพาะทาง** Frontmatter, Skills และ MCP
- ทำงานแบบ **Orchestration** Leader-first, Task List และ Pipeline
- คุม **Context** และ **Memory** สรุปกลับ Leader และจำข้าม Session
- มี **Guardrail** ที่ใช้ได้จริง Permission, Hooks และ Cost Gate
- ส่งงานถึง **Production** Capstone เว็บแอปบน Vercel

08 สิ่งที่คุณเรียนจะได้รับ Deliverables

- เอกสารประกอบการอบรมฉบับสมบูรณ์ครบทั้ง 2 วัน และ Lab Guide ที่ละเอียด
- ชุดไฟล์ Agent, Skill, Hook, .mcp.json และ settings.json ตัวอย่างสำหรับปรับใช้กับโปรเจกต์
- Template Delegation Prompt, Spec กลางของทีม และ Pre-flight Checklist ก่อนรัน Agent อัตโนมัติ
- ประกาศนียบัตรจบหลักสูตร (Certificate of Completion) และช่องทางสอบถามวิทยากรหลังจบหลักสูตร

09 ข้อมูลผู้สอน Instructor

อ.สามิตร ไทยม ผู้สอนและนักพัฒนาที่ใช้ Claude Code กับทีม Agent ในงานพัฒนาซอฟต์แวร์จริง ทั้งการวิเคราะห์ Codebase การสร้าง Feature และการตรวจงานอัตโนมัติ จุดเด่นคือการสอนจากโจทย์ที่ลงมือทำได้จริงทีละขั้น ให้ผู้เรียนเห็นถึงจุดที่ทีม Agent ช่วยให้งานเร็วขึ้นและจุดที่เปลือง Token หรือเสี่ยงเกินไป ฝึกให้ตั้ง Guardrail และตรวจผลงานของ Agent ทุกครั้ง เพื่อกลับไปออกแบบทีม Agent ในงานของตนเองได้อย่างมั่นใจและปลอดภัย

ราคาค่าอบรม 8,900 บาท/ท่าน · สนใจ IN-HOUSE ขอใบเสนอราคา

8,900 บาท / ท่าน (ยังไม่รวม VAT 7%)

สมัครผ่านเว็บไซต์ ลด 10%

☎ 02-570-8449

📞 @itgenius

☎ 088-807-9770

✉ contact@itgenius.co.th



เพิ่มเพื่อน
LINE