t03 — Fine-Tuning Prompt Design

2-Pass Architecture (Page Classify → Per-Pattern Extract)
Qwen3.6-35B-A3B fine-tuning, Constistant / Training repo

Compiled 2026-08-04
Contains: design README (2-pass architecture, per-pattern file-count audit, decisions log) · Pass 0 prompt (page classifier, draft v1) · Pass 1 I/O contract (all 15 patterns)
Source of truth: `primary_rawjson_schema.md` — this document does not override it.

t03 — 2-Pass Fine-Tuning Prompt Design

t03 prompt design — 2 passes, per-pattern extraction

> **สถานะ: ออกแบบ ยังไม่ได้เขียน prompt จริง** — เอกสารนี้คือแบบร่างให้ตัดสินใจก่อนลงมือ เขียน 2026-08-04

อำนาจสูงสุด = `primary_rawjson_schema.md`

ทุก prompt ในโฟลเดอร์นี้ต้อง**ผลิต output ที่ผ่าน** `tools/check_format.py` ซึ่งบังคับตาม [rawjson ยังไม่ได้แก้ไขโดยคน/00file for making rawjson from claude/primary_rawjson_schema.md]

ห้าม prompt ตัวไหนสร้างกฎของตัวเอง ถ้าเจอกรณีที่สเปคไม่ครอบคลุม → แก้สเปคก่อน (พร้อม log ตาม `rule_of_tune.md` ข้อ 7) แล้วค่อยให้ prompt อ้างถึง ไม่ใช่เขียนกฎใหม่ลงใน prompt เงียบๆ

ทำไม 2 pass ไม่ใช่ 5

production ปัจจุบัน (`js/drawing/drawing-index.js` → `qt_runRead()`) ใช้ 5 pass และ **ไม่ตรงกับ training data เลย** — dataset ของ t01/t02 คือ "1 หน้า → JSON เต็มของหน้านั้น" ซึ่งไม่ใช่ shape ของ pass ไหนเป๊ะๆ ผลคือโมเดลที่ทูนแล้วเสียบแทน production ไม่ได้จริง

2-pass ใหม่ออกแบบให้ **training shape = production shape** ตั้งแต่ต้น:

```
Pass 0  1 หน้า (ภาพ)                → wrapper §2 + รายการ view/pattern (§3)
Pass 1  1 หน้า (ภาพ) + pattern       → เนื้อหาตาม pattern นั้น (elements[] / grid{} / categories[])
merge   ไม่ใช้ AI                   → คำนวณ span จาก grid (§4), spec join (§7), merge point elements (§4)
```

ขั้น merge **ไม่มี prompt** — เป็นโค้ดล้วน จงใจ เพราะ §4 เขียนไว้ตรงๆ ว่า *"Span: calculated by code from the grid only — never let the model estimate distance"* (ข้อนี้คือสิ่งที่ production Pass 3 ปัจจุบันละเมิดอยู่ — สั่งโมเดลให้กะระยะเป็นเมตรจากภาพ)

Pass 0 — จำแนกหน้า + wrapper

output ที่ต้องได้ (§2 + §3)

§2 บังคับ 10 field บน wrapper: `png, doc_page, discipline, sheet_code, sheet_name, pattern, source_image, confidence_score, confidence_flags, warnings`

ในนั้น **3 ตัวไม่ต้องใช้ AI** — เป็นข้อเท็จจริงของไฟล์ ไม่ใช่สิ่งที่อ่านจากภาพ:

field	ที่มา
`png`	ชื่อไฟล์
`doc_page`	ลำดับหน้า
`source_image`	path เต็ม

AI ต้องตอบแค่: `discipline`, `sheet_code`, `sheet_name`, `pattern`, `confidence_score`, `confidence_flags`, `warnings`

⚠️ ประเด็นออกแบบข้อ 1 — 1 หน้าอาจมีหลาย pattern (§3)

§3 เขียนไว้ว่า *"A page may contain multiple views/patterns — inventory every view first with `views[]` (prevents losing one), then write each out as a separate file per view"*

แปลว่า Pass 0 **ห้ามตอบ pattern เดียวต่อหน้า** ต้องตอบเป็นรายการ:

```
{
  "sheet_code": "S-04",
  "sheet_name": "แปลนฐานรากและแปลนคาน",
  "discipline": "structural",
  "views": [
    { "view_no": 1, "view_title": "แปลนฐานราก", "pattern": "plan", "confidence_score": 0.95 },
    { "view_no": 2, "view_title": "แปลนคาน",   "pattern": "plan", "confidence_score": 0.95 }
  ],
  "confidence_flags": [],
  "warnings": []
}
```

นี่คือบทเรียนข้อ 7 ใน `No_touch_box/CLAUDE.md` โดยตรง — *"เคยพลาดเพราะบังคับเลือก `pattern` เดียว ต่อหน้า ทำให้ view ที่ 2 หายไปเงียบๆ ไม่มี warning"*

⚠️ ประเด็นออกแบบข้อ 2 — `pattern` มี 16 ค่า แต่ต้องเลือกจาก §1 เท่านั้น

§0.9: *"Never coin a new one"* — prompt ต้องพิมพ์รายการ 16 ค่าออกมาให้ครบใน instruction และ บังคับให้เลือกจากในนั้น พร้อมเกณฑ์แยกที่ §0.9 ระบุไว้:

- detail sheet → `section` (ไม่ใช่ `detail`)
- elevation / schematic diagram → `side_profile` (ไม่ใช่ `elevation`)
- site plan → `site_plan` (ไม่ใช่ `plan`)
- ตารางสรุป member → `schedule` · ตารางตัดเหล็กรายเส้น → `bbs_schedule`
- รายงานเจาะสำรวจดิน → `soil_boring_log` (ไม่ใช่ `schedule`/`notes`)

บ้าน 07 เคยประดิษฐ์ `detail`/`diagram`/`elevation` ขึ้นมาเองและต้อง remap 21 ไฟล์ — prompt ต้องกันเรื่องนี้ตั้งแต่ต้น

ไฟล์

```
pass0/
└── prompt.md          ← ตัวเดียว ไม่แยกตาม pattern (ยังไม่รู้ pattern ตอนนี้)
```

Pass 1 — สกัดเนื้อหา แยก prompt ต่อ pattern

input

```
ภาพหน้านั้น (1 ใบ) + pattern ที่ Pass 0 ตอบมา + view_title (ถ้ามีหลาย view)
```

output

เนื้อหาตาม container ที่ §0.1 กำหนด — **3 แบบเท่านั้น ห้ามประดิษฐ์ชื่อ array ใหม่**

pattern	container (§0.1)
`gridline`	`grid{ x_lines[], y_lines[] }`
`material_list`	`categories[].items[]`
ที่เหลือทั้งหมด	`elements[]`

กฎที่ทุก prompt ต้องมีเหมือนกัน (ใส่เป็น shared block ไม่ copy 15 รอบ)

§	กฎ
§0.2	ทุก element มี `element_id`, `element_type`, `confidence_score`, `confidence_flags` · `confidence_score: null` ได้ **ห้ามมั่วตัวเลข**
§0.3	`element_id` = มาร์คที่พิมพ์บนแบบเท่านั้น ห้ามต่อท้ายตำแหน่ง/ชั้น/section
§0.4	`element_type` เลือกจาก 40 ค่าที่มี ห้ามประดิษฐ์ (เคยบานเป็น 359 ค่า)
§0.5	ขนาด member = mm จำนวนเต็ม (`width_mm`) · ระดับ/ระยะ/พิกัด = เมตร (`level_m`, `span_length_m`, `pos_m`) เป็นตัวเลข ไม่ใช่ string
§0.7	ทุกค่าที่แปลงจากที่พิมพ์บนแบบ เก็บต้นฉบับไว้ใน `*_printed_as`
§0.8	`grid_ref`: จุด = ไม่มีขีด (`C1`) · ช่วง = มีขีด (`D-C`) · 2 แกน = แนวตั้งก่อน (`D-C x 1-2`) · ห้ามมีคำว่า `grid`/`dummy`
§2a	ห้ามใส่ `phase_note` (ใช้เฉพาะ staged `op2` เท่านั้น)

กฎเฉพาะแต่ละ pattern

#	pattern	§ ที่บังคับ	จุดยากที่ prompt ต้องเน้น
1	`plan`	§4, §5, §10, §10a	**beam-endpoint rule** (ปลายคานไม่อยู่บนกริด = ต้องมี dummy grid ไม่ใช่ทั้งคาน) · atomic ต่อ segment สำหรับคาน แต่ merge เป็นหนึ่งเดียวสำหรับ footing/column · ลำดับ element (บนลงล่าง ซ้ายไปขวา แนวตั้งก่อนแนวนอน) · `SI` vs `S1`
2	`section`	§6, §6a, §7	`main_bar` **แยก top/middle/bottom เสมอ** แต่ **column ใช้** `count` **เดี่ยว ห้ามแยก** (แยกแล้วนับเหล็กเป็น 2 เท่า) · `Ø` = RB เสมอ ห้ามเดาจากขนาด · `additional_bars` ต้องดู leader line จริง ไม่เชื่อ label
3	`schedule`	§6, §6a, §7, §8	`level` เป็น field แยก ห้ามยัดลง `element_id` · ตารางหลายชั้นข้ามาร์คได้ (ข้อยกเว้นของกฎ merge §4)
4	`notes`	§0.1 ข้อยกเว้น	`sections[{heading, items[]}]` = หัวข้อโน้ต **ไม่ใช่** drawing element ห้ามพับเข้า `elements[]`
5	`index`	§0.1 ข้อยกเว้น	`sections[{title, sheet_range}]` = สารบัญเอกสาร ไม่ใช่ element
6	`material_list`	§11	1 PNG อาจเป็น 2 แผ่นจริง (หมุน 90° แล้วผ่าซ้าย/ขวา) · แถวต่อเนื่องที่ไม่มี item_no ต้องเป็นแถวแยก ห้ามรวมกับแถวบน · `columns[]` = หัวตาราง ไม่ใช่ element
7	`site_plan`	§13	⚠️ `element_type` **ยังไม่ standardize** (10 ค่าข้าม 5 บ้าน ซ้ำความหมายกัน) — prompt ต้องล็อกชื่อให้เหลือชุดเดียวก่อน ไม่งั้นจะ drift ต่อ
8	`side_profile`	§1 #8	รูปด้าน/รูปตัดอาคาร ไม่มีเหล็ก ไม่ใช่ข้อมูลภูมิประเทศ
9	`gridline`	§4, §11a	**ยากสุดทั้งชุด** — dummy grid ตั้งชื่อตามกริดบน/ซ้าย (ไม่ใช่ใกล้สุด) · prime ordering · `pos_m` ติดลบได้ถ้าอยู่ก่อน origin · `pos_m` อ่านจากเส้นบอกระยะที่พิมพ์จริง **ห้ามเดา** · หลายอาคาร = grid master แยกไฟล์
10	`title`	§1 #10 *(draft)*	ยังไม่มี field-set ยืนยัน
11	`symbol`	§1 #11 *(draft)*	`fixture_symbol_legend[]` เป็นตาราง reference ไม่พับเข้า `elements[]`
12	`roof_plan`	§1 #12 *(draft)*	สันหลังคา/ตะเข้/ชายคา
13	`misc`	§1 #13	ตารางราคาซีรีส์ = `series_price_table[]` รูปเดียวเท่านั้น (เคยเก็บ 5 แบบใน 5 บ้าน)
14	`bbs_schedule`	§1 #14 *(draft)*	**ยังไม่มีข้อมูลเทรนเลย 0 ไฟล์**
15	`soil_boring_log`	§1 #15 *(draft)*	**ยังไม่มีข้อมูลเทรนเลย 0 ไฟล์**
—	`unknown`	—	ไม่มี prompt โดยนิยาม

ไฟล์

```
pass1/
├── _common/
│   ├── notation.md        ← สัญกรณ์แบบก่อสร้างไทย (DB/RB/Ø/ค1/w1/@0.20/fc'/SD40)
│   ├── confidence.md      ← §0.2 ห้ามเดา null + flag ดีกว่าเลขมั่ว
│   └── format_lock.md     ← §0.2-§0.8 กฎที่ทุก pattern ใช้ร่วมกัน
├── plan.md
├── section.md
├── schedule.md
├── notes.md
├── index.md
├── material_list.md
├── site_plan.md
├── side_profile.md
├── gridline.md
├── title.md
├── symbol.md
├── roof_plan.md
├── misc.md
├── bbs_schedule.md
└── soil_boring_log.md
```

15 ไฟล์ ไม่ใช่ 13 — สเปคเพิ่ม `bbs_schedule` + `soil_boring_log` เมื่อ 2026-08-04 (16 pattern ลบ `unknown` = 15) เลข 13 คือของเดิมก่อนวันนั้น

⚠️ ปัญหาใหญ่สุดของดีไซน์นี้ — ข้อมูลเทรนต่อ pattern ไม่เท่ากันอย่างรุนแรง

นับจริงจาก `rawjson_ยังไม่ได้แก้ไขโดนคน/` ทั้ง **12 บ้าน** (2026-08-04, รวม **1,240 ไฟล์**):

pattern	ไฟล์	บ้าน	เทรนแยกไหวไหม
`material_list`	435	10	✅
`section`	243	12	✅
`plan`	192	12	✅
`side_profile`	94	12	✅
`schedule`	71	12	✅
`notes`	58	12	🟡 บาง
`index`	42	12	🟡 บาง
`symbol`	27	9	🟡 บาง
`misc`	25	9	🟡 บาง
`roof_plan`	23	12	🟡 บาง
`gridline`	**13**	12	🔴 **น้อยเกิน**
`site_plan`	12	12	🔴 น้อยเกิน
`title`	5	5	🔴 น้อยเกิน
`bbs_schedule`	**0**	0	⛔ ไม่มีเลย
`soil_boring_log`	**0**	0	⛔ ไม่มีเลย

`gridline` คือปัญหาที่ต้องแก้ก่อนอย่างอื่น: - มีแค่ **13 ไฟล์** (1 ต่อบ้าน) เพราะโดยนิยามมันเป็น master 1 ตัวต่อบ้าน - แต่มันคือ pattern ที่**ยากที่สุด** และ**ทุกอย่างขึ้นกับมัน** — README ของ `op2` เขียนไว้ตรงๆ: *"Move one grid line and every `span_length_m` in every `plan` file is wrong"* - t01/t02 เจอปัญหาเดียวกันมาแล้ว: gridmaster 5 ตัว จาก 403 ตัวอย่าง และเป็นตัวที่บังคับให้ ต้องดัน `MAX_LENGTH` จาก 9,216 → 24,576 เพราะมัดภาพ 2-4 ใบต่อตัวอย่าง

ตัดสินใจแล้ว (มะขาม, 2026-08-04) — 4 ข้อ

1. `gridline` — ใช้หน้า 00 ของทุกบ้านตรงๆ ✅ ตัดสินใจแล้ว

ไม่รวมกับ `plan` — `gridline` แยก prompt ของตัวเองต่อไป dataset = ไฟล์ `<house>_หน้า00_ gridline.json` ของทุกบ้าน (13 ไฟล์ = 13 บ้าน ณ ตอนนับ, 1 ไฟล์ต่อบ้านโดยนิยาม) ยอมรับว่าจำนวน ตัวอย่างน้อยเพราะเป็นธรรมชาติของ pattern นี้ (master ไฟล์เดียวต่อบ้าน ไม่ใช่ต่อหน้า) ไม่ใช่ปัญหา ที่ต้องแก้ด้วยการรวม pattern — ถ้าจะเพิ่มตัวอย่างต้องมาจากการเพิ่มจำนวนบ้าน ไม่ใช่เปลี่ยนนิยาม

2. `title` / `site_plan` / `bbs_schedule` / `soil_boring_log` — แก้ที่ dataset ตรงๆ ✅ ตัดสินใจแล้ว

ไม่ตัดออกจาก training set — แนวทางคือ**ขยาย dataset จริง** (เพิ่มบ้าน/เพิ่มหน้าที่มี pattern พวกนี้จริงในแบบ) ไม่ใช่เขียน prompt ทิ้งไว้เฉยๆ แล้วข้ามการเทรน `bbs_schedule`/`soil_boring_log` ยังมี **0 ตัวอย่าง** อยู่ — ต้องหาว่าบ้านไหนมีหน้าประเภทนี้จริงในแบบต้นฉบับก่อนถึงจะสร้าง label ได้ (ยังไม่ได้ทำ — ดูหัวข้อ "ค้าง" ท้ายไฟล์)

3. Pass 0 — เขียนออกมาก่อน แม้ยังไม่จบ ✅ เขียนแล้ว (draft v1)

มติ: **priority หลักคือแก้ dataset** (โดยเฉพาะคานหายบ้าน 06-12) ส่วน Pass 0 "เดี๋ยวมาต่อ" — ยังเขียน prompt ออกมาก่อนตามที่สั่ง → `pass0/prompt.md` เขียนแล้ว (16 pattern เต็ม, wrapper + `views[]` ตาม §2/§3) **แต่ยังเป็น draft v1** — 3 อย่างยังไม่ทำ: (1) ยังไม่มีสคริปต์ แปลง raw JSON → training pair จริง (ต้อง group ไฟล์ `source_image` เดียวกัน), (2) ยังไม่เลือก โมเดล/hyperparameter, (3) ยังไม่ทดสอบกับภาพจริงสักหน้า

4. Token budget — ยังไม่ทำ รอทุกอย่างนิ่งก่อน ✅ ตัดสินใจแล้ว

ไม่คำนวณตอนนี้ — รอจน dataset (โดยเฉพาะ Pass 1 ต่อ pattern) นิ่งก่อน แล้วคำนวณทั้งชุดทีเดียว ครั้งเดียว ไม่ทยอยคำนวณทีละ pattern (กัน error สะสมแบบ t02 บั๊ก 1)

ที่ยังไม่ได้ทำ (สถานะจริง หลังมติ 2026-08-04)

Priority ตอนนี้ = แก้ dataset ก่อน (มติข้อ 3) — ไม่ใช่เขียน prompt ให้ครบ 15 ไฟล์ก่อน

- ⛔ **บล็อกอันดับ 1: คานหายในบ้าน 06-12 ยังไม่เคยตรวจ** (บ้าน 01-05 หายทุกหลังตอนตรวจ: 20→27, 22→36, 30→38, 5→21) — เทรนด้วย label ที่คานหาย = สอนให้ under-predict ตรงๆ
- ⛔ **`bbs_schedule` / `soil_boring_log` มี 0 ตัวอย่าง** — ต้องหาว่าบ้านไหนมีหน้าประเภทนี้จริงก่อน (มติข้อ 2: แก้ด้วยการขยาย dataset ไม่ใช่ตัดออกจากแผนเทรน)
- ✅ `pass0/prompt.md` เขียนแล้ว (draft v1) — ยังไม่มีสคริปต์แปลงข้อมูล, ยังไม่ทดสอบจริง
- ✅ `pass1/IO_SPEC.md` เขียนแล้ว — input/output contract ละเอียดต่อ pattern ครบ 15 ตัว (ภาพที่ต้องการ, auxiliary input เช่น grid master, container, field list ทุกตัว อ้าง § ของสคีมา) **แต่ยังไม่ใช่ prompt text จริง** — เป็นสเปคเตรียมก่อนเขียน prompt แต่ละไฟล์
- ⬜ `pass1/plan.md`, `section.md`, ... (prompt text จริง 15 ไฟล์) — ยังไม่เขียนสักไฟล์ รอ dataset นิ่งตามลำดับ priority ข้างบน
- ⬜ ยังไม่ได้เขียนสคริปต์แปลง raw JSON → training pair (ทั้ง Pass 0 และ Pass 1)

Pass 0 — Page Classifier Prompt (draft v1)

> ⚠️ อ่าน `rule_of_tune.md` ก่อนแก้ prompt นี้ — เข้าข่ายกฎข้อ 2 (การแก้ script/prompt ที่กระทบข้อมูลทุนต้องเตือนก่อน)

Pass 0 — Page Classifier (t03)

อำนาจสูงสุด = `primary_rawjson_schema.md` §1 (pattern taxonomy), §2 (wrapper fields), §3 (multi-view). ห้าม prompt นี้สร้างกฎ pattern ของ ตัวเอง — ถ้าพบกรณีที่สเปคไม่ครอบคลุม ให้แก้สเปคก่อน (พร้อม log ตาม rule 7) ไม่ใช่ยัดกฎใหม่ไว้ในนี้

ต่างจาก production Pass 0 ตรงไหน

Production (`QT_PROMPT_PAGE_IDENTIFY`, `js/ai/prompts.js`) ยัดทุกหน้าของ PDF เข้า call เดียว ตอบแค่ 2 หมวด (`section_pages`/`layout_pages`) — ใช้สำหรับ **routing ระหว่าง pass** ไม่ใช่ label ที่ train ได้ Pass 0 ตัวนี้**คนละงาน**: รับภาพ **1 หน้า** ตอบ **wrapper เต็ม + pattern ที่ถูกต้อง เต็ม 16 ค่า** เพื่อให้ output จับคู่กับ raw JSON ground truth ได้ตรงๆ (1 หน้า → 1 label)

หลักการทำงาน

```
input:  ภาพ 1 หน้า (PNG, ตามที่ op1 อ่านจริง — ไม่ย่อ thumbnail แบบ production Pass 0)
output: sheet-level metadata + รายการ view บนหน้านั้น (อาจมีมากกว่า 1 view — ดู §3)
```

`png`, `doc_page`, `source_image` **ไม่ต้องให้ AI ตอบ** — เป็นข้อเท็จจริงของไฟล์ ไม่ใช่สิ่งที่อ่าน จากภาพ ใส่เข้า wrapper ทีหลังด้วยโค้ด

System Prompt

```
You are a structural engineer indexing pages from a Thai RC construction drawing set.
Your job is NOT to extract element data yet — only to identify what KIND of page this is
and how many distinct views/sections it contains.

CRITICAL — do not skip this: many pages carry more than one drawing on them (e.g. a footing
plan and a beam plan side by side, or a detail box inset into a larger sheet). You must find
EVERY view before answering. A page you record as having only 1 view when it actually has 2
is a silent data-loss bug — the second view disappears with no warning anywhere downstream.

Work this way, in order:
1. Read the title block: find the sheet code (e.g. "S-04", "A-11") and the sheet name/title
   as printed (e.g. "แปลนฐานรากและแปลนคาน").
2. Scan the WHOLE page for every underlined or bold heading/caption — each one is a candidate
   view. Count them before classifying anything.
3. For each view found, classify it into exactly one of the 16 patterns below.
4. Never invent a 17th pattern. If nothing fits, use "unknown" and say why in confidence_flags.
```

Pattern taxonomy — the 16 categories (verbatim from schema §1)

```
1.  plan            floor plan / layout, has grid_ref
2.  section         detail section — rebar spec/dimensions for beam, column, footing
3.  schedule        summary table of a MEMBER type (column, beam, door, window, fence, ...)
4.  notes           project-level requirements/specs
5.  index           drawing set table of contents
6.  material_list   bill of quantities (BOQ)
7.  site_plan       site layout
8.  side_profile    non-top-down view — elevation/building section (NOT terrain, no rebar)
9.  gridline        grid reference file (per-page companion + หน้า00 master)
10. title           cover page
11. symbol          symbol/legend page
12. roof_plan       roof plan — ridge/hip lines, eave overhangs
13. misc            whole-series catalog/promotional page, not about THIS house's construction
14. bbs_schedule    bar bending schedule — ONE ROW = ONE CUT BAR (bar_mark, shape_code, len_A/B/C,
                    qty, grade). NOT the same as "schedule" — a schedule row describes a MEMBER.
15. soil_boring_log soil investigation / borehole log — SPT blow counts, stratum table, lab
                    results, groundwater level. No grid_ref, no element marks, no rebar.
16. unknown         doesn't fit any of the 15 above, or page is unreadable
```

Disambiguation rules — read before answering (schema §0.9, §1)

- A **detail sheet** → `section`, never a made-up `detail`

- An **elevation or schematic diagram** → `side_profile`, never `elevation`/`diagram`
- A **site layout** → `site_plan`, never plain `plan`
- A table of **members** (1 row = 1 column/beam/door/window) → `schedule`
- A table of **cut bars** (1 row = 1 bar with a bend shape and length) → `bbs_schedule`
- A **soil/borehole report** (SPT, stratum table, groundwater) → `soil_boring_log`, never `schedule` or `notes`
- A **whole-series price table or design-collage cover** (covers all houses in a series, not just this one) → `misc`, never `title`
- This house's own cover page only → `title`

User Prompt Template

```
This is one page from a Thai RC structural drawing set.

Read the title block first: sheet code, sheet name, discipline (structural/architectural/
sanitary/electrical — infer from the sheet code prefix and content, e.g. S-xx=structural,
A-xx=architectural).

Then find every distinct view on this page (§ above) and classify each one.

Return ONLY valid JSON, no markdown fences, no commentary:

{
  "sheet_code": "S-04",
  "sheet_name": "แปลนฐานรากและแปลนคาน",
  "discipline": "structural",
  "views": [
    {
      "view_no": 1,
      "view_title": "แปลนฐานราก",
      "pattern": "plan",
      "confidence_score": 0.95,
      "confidence_flags": []
    },
    {
      "view_no": 2,
      "view_title": "แปลนคาน",
      "pattern": "plan",
      "confidence_score": 0.9,
      "confidence_flags": []
    }
  ],
  "warnings": []
}

If the page has only one view, "views" still has exactly one entry — never omit the array.
If a sheet_code is genuinely absent (e.g. a cropped or borderless image), set it to null and
add "sheet_code_not_visible" to confidence_flags — do NOT invent one.
```

Confidence discipline (per schema §0.2 + Constistant's `QT_CONFIDENCE_POLICY`)

- `confidence_score: null` is allowed when the sheet genuinely gives nothing to judge by — **never invent a number to fill the field.**
- Reserve `confidence_score ≥ 0.9` for pages where the sheet code AND the pattern-defining content (rebar dots for `section`, grid_ref lines for `plan`, SPT table for `soil_boring_log`) are both unambiguous.
- If classifying by sheet code alone (content unclear/blurry) → cap confidence at 0.5 and add a flag naming what was unclear.
- A page you cannot decide between two patterns for → pick the more specific one and flag the runner-up in `confidence_flags` (e.g. `"could_be_side_profile_instead_of_roof_plan"`) — never silently pick one with no trace of the ambiguity.

ยังไม่ตัดสินใจ / ยังไม่ทำ (สถานะจริง ไม่ใช่ของที่ทำเสร็จแล้ว)

- **โมเดล/hyperparameter สำหรับ Pass 0 ยังไม่เลือก** — ตาม decision log (2026-08-04): ใช้ dataset fix ก่อน ค่อยกลับมาปิด Pass 0 ให้จบ
- **ยังไม่มีสคริปต์แปลง raw JSON → training pair ของ Pass 0** — ต้อง group ไฟล์ที่ `source_image` เดียวกันเข้าด้วยกันก่อน (ตอนนี้ 1 ไฟล์ = 1 view แต่ Pass 0 ต้องตอบ 1 หน้า = หลาย view)
- **ยังไม่คำนวณ token budget** — รอจน dataset (Pass 1) นิ่งก่อนแล้วคำนวณพร้อมกันทีเดียว (มติ 2026-08-04 ข้อ 4: "ยังไม่ต้องทำ รอแก้เสร็จทุกอย่างทีเดียว")
- **ยังไม่ทดสอบกับภาพจริงสักหน้า** — นี่คือ draft v1 เขียนจาก schema ล้วนๆ ยังไม่ผ่าน dry-run

Pass 1 — Input/Output Contract, per Pattern

> ⚠️ อ่าน `rule_of_tune.md` ก่อนแก้ไฟล์นี้ — เข้าข่ายกฎข้อ 2 เหมือนไฟล์อื่นใน `prompt/`

Pass 1 — Input/Output contract, per pattern

15 pattern (16 ลบ `unknown` ซึ่งไม่มี Pass 1 โดยนิยาม) — แต่ละ pattern มี prompt ของตัวเอง เอกสารนี้กำหนด **input จริง / output จริง** ของแต่ละตัว ก่อนไปเขียน prompt text จริง

Input ที่ทุก pattern ใช้ร่วมกัน (ไม่ต้องเขียนซ้ำ 15 รอบ)

ส่วนประกอบ	มาจากไหน	หมายเหตุ
ภาพ	1 หน้า PNG เต็มหน้า (ความละเอียดเดียวกับที่ `op1` อ่านจริง)	**ไม่ crop เฉพาะ view** — ไม่มีพิกัด bounding box เก็บไว้ในสคีมาเลย (Pass 0 ให้แค่ `view_title`/`pattern` ไม่ให้พิกัด) ถ้าหน้านั้นมีหลาย view ต้องบอกโมเดลด้วยข้อความว่า **"สกัดเฉพาะ view ที่ชื่อ X เท่านั้น"** ให้โมเดลเล็งเอง
`sheet_code`, `sheet_name`, `discipline`	ผลจาก Pass 0 (รู้อยู่แล้ว ไม่ต้องให้โมเดลตอบซ้ำ)	ใส่เข้า wrapper ตอน merge ไม่ใช่ Pass 1
`view_title` (ถ้ามีหลาย view)	ผลจาก Pass 0	บอกโมเดลว่ากำลังสกัด view ไหนของหน้านั้น
`png`, `doc_page`, `source_image`	ข้อเท็จจริงของไฟล์	ไม่ต้องให้ AI ตอบ (เหมือน Pass 0)
Confidence discipline	§0.2	`confidence_score: null` ดีกว่าเลขมั่ว ทุก pattern ใช้กฎเดียวกัน
Thai notation guide	เหมือนที่ Constistant ใช้ (`QT_THAI_NOTATION_GUIDE`)	DB/RB/Ø/ค/w/@/fc'/SD40 — ใช้ร่วมทุก pattern ที่มีเหล็ก

Auxiliary input พิเศษ (ไม่ใช่ทุก pattern ได้):

pattern	ต้องการ grid master เพิ่มไหม	เหตุผล
`plan`, `roof_plan`	✅ ต้องการ — แนบ `หน้า00_gridline.json` ของบ้านนั้น (หรืออย่างน้อยรายชื่อ dummy grid ทั้งหมด)	ต้องเขียน `grid_ref` ให้ตรงกับชื่อ dummy grid ที่ประกาศไว้แล้ว (`1'`, `A''`) ไม่งั้นโมเดลจะประดิษฐ์ชื่อใหม่/เขียน `"?"` ทั้งที่จริงมีชื่อแล้ว
`section`, `schedule`, `notes`, `index`, `material_list`, `site_plan`, `side_profile`, `title`, `symbol`, `misc`, `bbs_schedule`, `soil_boring_log`	❌ ไม่ต้องการ	ไม่มี `grid_ref` ในเนื้อหา (spec/ตาราง/ข้อความล้วน)
`gridline`	— (เป็นตัว grid master เอง)	ดูหัวข้อเฉพาะด้านล่าง — อาจต้องการ**หลายภาพ** ไม่ใช่ auxiliary text

---

ตารางสรุป 15 pattern

#	pattern	container (§0.1)	ต้องการ grid master	ตัวอย่างเทรนจริง	สถานะ
1	`plan`	`elements[]`	✅	192	พร้อม
2	`section`	`elements[]`	❌	243	พร้อม
3	`schedule`	`elements[]`	❌	71	พร้อม
4	`notes`	`elements[]` หรือ `sections[]` (ข้อยกเว้น §0.1)	❌	58	พร้อม (บาง)
5	`index`	`sections[]` (ข้อยกเว้น §0.1)	❌	42	พร้อม (บาง)
6	`material_list`	`categories[].items[]`	❌	435	พร้อม
7	`site_plan`	`elements[]`	❌	12	⚠️ น้อย + §13 element_type ยังไม่ standardize
8	`side_profile`	`elements[]`	❌	94	พร้อม
9	`gridline`	`grid{x_lines[],y_lines[]}`	(เป็นตัวเอง)	13	⚠️ น้อยโดยธรรมชาติ (มติ 08-04: ไม่รวมกับ plan)
10	`title`	`elements[]` (draft)	❌	5	🔴 น้อยเกิน + draft
11	`symbol`	`fixture_symbol_legend[]`-style (draft)	❌	27	⚠️ draft
12	`roof_plan`	`elements[]` (draft)	✅	23	⚠️ draft + บาง
13	`misc`	`series_price_table[]` หรือ `elements[]`	❌	25	⚠️ บาง
14	`bbs_schedule`	`elements[]` (draft)	❌	**0**	⛔ ไม่มีตัวอย่างเลย
15	`soil_boring_log`	`elements[]` + wrapper พิเศษ (draft)	❌	**0**	⛔ ไม่มีตัวอย่างเลย

---

1. `plan`

Input: ภาพหน้า (เต็มหน้า) + **grid master ของบ้านนั้น** (รายชื่อ `x_lines[]`/`y_lines[]` ทั้งหมด รวม dummy grid — ไม่ต้องส่งทั้งไฟล์ JSON เต็ม แค่รายชื่อ+`pos_m` พอ) + `view_title` ถ้ามีหลาย view บนหน้าเดียว

Output (`elements[]`, §4-§5, §10, §10a):

field	type	บังคับ	หมายเหตุ
`element_id`	string	✅	มาร์คที่พิมพ์เท่านั้น (§0.3)
`element_type`	string	✅	จากรายการ 40 ค่า (§0.4)
`grid_ref_start` / `grid_ref_end`	string	span element (คาน)	จุด ไม่มีขีด (§0.8)
`grid_refs[]`	array of string	point element (footing/column)	array ไม่ใช่ comma-string (§4) — **merge เป็น 1 entry ต่อมาร์ค**
`span_length_m`	number|null	span element	**โมเดลกรอกได้ แต่โค้ดจะทับใน merge เสมอ** (§4) — ห้ามพึ่งค่านี้เป็นค่าสุดท้าย
`span_source`	enum	span element	`grid_table`/`local_dimension`/`unresolved`/`n/a`
`count`	number	point element	หลัง merge (§4)
`confidence_score`/`confidence_flags[]`	—	✅	§0.2

กฎพิเศษที่ prompt ต้องเน้น: beam-endpoint rule (§4 "How to FIND dummy grids") — ปลายคานที่ไม่อยู่บนกริดที่มี **ห้ามทิ้งคาน ห้ามเขียนบรรยายแทน grid_ref** ต้องขอ dummy grid เพิ่ม (แต่ Pass 1 **แก้ grid master เองไม่ได้** — ทำได้แค่ flag `confidence_flags: ["needs_dummy_grid_at_<pos>"]` ให้ merge/มนุษย์จัดการ เพราะ gridline เป็นไฟล์แยก แก้พร้อมกันไม่ได้ในการเรียกเดียว)

ลำดับ elements[]: บนลงล่าง ซ้ายไปขวา แนวตั้งก่อนแนวนอนถ้าจุดเริ่มตรงกัน (§4 "Element ordering")

---

2. `section`

Input: ภาพหน้า (เต็มหน้า) เท่านั้น — ไม่มี auxiliary

Output (`elements[]`, §6, §6a, §7):

field	type	หมายเหตุ
`element_id`	string	มาร์ค (§0.3)
`element_type`	string	§0.4
`width_mm` / `height_mm` / `thickness_mm` / `depth_mm`	number	จำนวนเต็ม (§0.5) ห้าม packed string
`main_bar`	object	**แยก `top` / `bottom` เสมอแม้เท่ากัน** (§6) — column **ห้าม**แยก ใช้ `count` เดี่ยว
`main_bar.middle`	object	optional — มีเฉพาะเมื่อเห็นแถวเหล็กกลางจริง (§6)
`additional_bars[]`	array	เฉพาะเหล็กที่ไม่อยู่หน้าไหนเลย — **เช็ค leader line จริง ห้ามเชื่อ label เฉยๆ** (§7)
`stirrup`	object	`{count?, dia_mm, type, spacing_mm}` — ชื่อ `stirrup` เท่านั้น ห้าม `tie`/`tie_bar` (§0.6)
`steel_section`	object	ถ้าเป็นเหล็กรูปพรรณ แทน `main_bar`/`stirrup` (§6a) — **มีอย่างใดอย่างหนึ่งเท่านั้น ห้ามมีทั้งคู่**
`concrete_grade`/`steel_grade`	string	§7
`confidence_score`/`confidence_flags[]`	—	§0.2

กฎพิเศษ: `Ø` = RB เสมอ ห้ามเดาจากขนาด (§6 ท้าย) · column ใช้ `count` เดี่ยว **ห้ามแยก top/bottom** (แยกแล้วนับเหล็กเป็น 2 เท่า) · ถ้ามาร์คเดียวกันปรากฏหลายจุดในหน้าเดียวและสเปคเหมือนกันทุกตัวอักษร — เขียนแค่ครั้งเดียวพอ ปล่อยให้ merge จัดการ join เข้า `specs{}` (§7)

---

3. `schedule`

Input: ภาพหน้า เท่านั้น

Output (`elements[]`, §6, §6a, §7, §8):

โครงสร้าง field เหมือน `section` ทุกประการ (ใช้ prompt block เดียวกันสำหรับ rebar/spec ได้) **ต่างแค่บริบท** — 1 แถว = 1 member ในตาราง ไม่ใช่หน้าตัดขยาย

field เพิ่มเฉพาะ pattern นี้:

field	type	หมายเหตุ
`level`	string	multi-level schedule (§8) — **ห้ามยัดลง `element_id`** ต้องแยก field เสมอ เช่น `C1` @ `"roof frame"` vs `C1` @ `"ground floor, pedestal, footing"`

กฎพิเศษ: มาร์คเดียวกันซ้ำได้ถ้า `level` ต่างกัน (ข้อยกเว้นเดียวของกฎ merge §4 — ตารางหลายชั้นห้าม merge เข้าด้วยกัน)

---

4. `notes`

Input: ภาพหน้า เท่านั้น

Output: สอง container ที่ยอมรับได้ (§0.1 ข้อยกเว้น) — เลือกตามรูปแบบจริงของหน้า:

รูปแบบหน้า	container
ข้อความยาวไม่มีหัวข้อข้อขัด	`elements[]` (`element_type: "note"`)
แบ่งเป็นหัวข้อลำดับเลข ("1. ข้อกำหนดทั่วไป", "2. ...")	`sections[{heading, items[]}]` — **นี่ไม่ใช่ drawing element ห้ามพับเข้า `elements[]`**

กฎพิเศษ: เนื้อหาระดับโปรเจกต์ (fc', fy, cover, มาตรฐานอ้างอิง) — คัดลอกข้อความจริง ไม่สรุปย่อ (จะเสียความแม่นยำ)

---

5. `index`

Input: ภาพหน้า เท่านั้น

Output (`sections[]`, ข้อยกเว้น §0.1):

field	type	หมายเหตุ
`title`	string	ชื่อหมวด เช่น `"แบบสถาปัตยกรรม"`
`sheet_range`	string	เช่น `"A-01 ถึง A-15"`

กฎพิเศษ: นี่คือสารบัญเอกสาร — **ไม่ใช่ element ของแบบ** ห้ามพับเข้า `elements[]` แม้จะมี field คล้าย element

---

6. `material_list`

Input: ภาพหน้า — ⚠️ **ก่อนส่งเข้า Pass 1 ต้องเช็คก่อนว่า 1 PNG เป็น 2 แผ่นจริงซ้อนกันไหม** (§11: หมุน 90° แล้วแยกซ้าย/ขวาเป็นคนละภาพ) — **นี่คือ pre-processing ก่อน Pass 1 ไม่ใช่สิ่งที่ขอให้โมเดลทำเอง** ถ้าตรวจพบว่าเป็น 2 แผ่น ต้องแยกไฟล์ก่อนแล้วเรียก Pass 1 สองครั้ง

Output (`categories[].items[]`):

field	type	หมายเหตุ
`category`	string	ชื่อหมวดงาน
`items[].item_no`	string	เลขลำดับ
`items[].description`	string	รายการ
`items[].quantity`	number	—
`items[].unit`	string	หน่วยไทย (ลบ.ม./ตร.ม./กก. ฯลฯ)

กฎพิเศษ: แถวต่อเนื่องที่ไม่มี item_no/qty ของตัวเอง **ต้องเป็นแถวแยก ห้ามรวมกับแถวบน** (§11) · `columns[]` ในหน้าตารางนี้ = หัวตาราง ไม่ใช่ element (§0.1)

---

7. `site_plan`

Input: ภาพหน้า เท่านั้น

Output (`elements[]`):

field พื้นฐานตาม §0.2/§0.5 ทั่วไป

⚠️ **กฎพิเศษที่สำคัญที่สุดของ pattern นี้ (§13):** `element_type` **ยังไม่ standardize** — พบ 10 ค่าซ้ำความหมายกันข้าม 5 บ้านเดิม (`building_footprint`/`building_outline`, `boundary_line`/`lot_boundary`, `grading_note`/`grading_note_or_slab`) **prompt ต้องล็อกชื่อให้เหลือชุดเดียวตั้งแต่ตอนเขียน** ไม่ปล่อยให้โมเดลเลือกเองเหมือนที่ผ่านมา ไม่งั้นจะ drift ต่อเข้า dataset ใหม่อีก — **ต้องตัดสินใจชื่อ canonical ก่อนเขียน prompt จริง** (ยังไม่ได้ตัดสินใจ ณ ตอนเขียนไฟล์นี้)

---

8. `side_profile`

Input: ภาพหน้า เท่านั้น

Output (`elements[]`):

รูปด้านหรือรูปตัดอาคาร (ไม่ใช่ terrain/site info) — field หลักคือ `level_m` (ระดับ), มิติความสูง, ป้ายกำกับชั้น ไม่มีเหล็ก ไม่มี `grid_ref`

กฎพิเศษ: ห้ามใช้ pattern นี้กับข้อมูลที่ตั้ง/ภูมิประเทศ (ชื่อเดิม `site_profile` เคยทำให้เข้าใจผิด — เปลี่ยนชื่อแล้วตั้งแต่ 07-08)

---

9. `gridline`

Input: อาจมากกว่า 1 ภาพ — ต่างจากทุก pattern อื่น เพราะ dummy grid บางเส้นต้อง cross-reference จากหลายหน้า (เช่น plan sheet ยืนยันตำแหน่งหนึ่ง, beam plan ยืนยันอีกตำแหน่ง) ตรงกับที่ §2 ให้ใช้ `source_pages[]` แทน `source_image` เดี่ยว

Output (`grid{x_lines[], y_lines[]}`):

```
{
  "id": "1'",
  "pos_m": 7.6,
  "type": "dummy",
  "confidence_score": 1,
  "confidence_flags": []
}
```

กฎพิเศษ (ยากสุดทั้งชุด, §4): - ตั้งชื่อ dummy grid ตามกริดที่**บน/ซ้าย** เสมอ (ไม่ใช่ใกล้สุด) - ลำดับ prime ตามทิศการอ่าน (x: ซ้าย→ขวา, y: บน→ล่าง) — เจอตัวแรก 1 prime เจอตัวสอง 2 prime - จุดกำเนิด (0,0) ต้องเป็นกริดหลักซ้ายสุด/บนสุดเสมอ — dummy grid ห้ามแย่งตำแหน่งนี้ ถ้าอยู่ก่อน origin ใช้ `pos_m` **ติดลบ** - `pos_m` อ่านจากเส้นบอกระยะที่พิมพ์จริงเท่านั้น **ห้ามเดา** - ต้องระบุด้วยว่าแบบชุดนี้ใช้ **drawing break line** หรือไม่ (สัญลักษณ์ซิกแซกกลางเส้น มักถูกมองข้าม — บทเรียนบ้าน 06 ที่ทำให้ grid master ต้องพลิกจาก 10 เส้นเป็น 11 เส้น) - multi-building set (§11a): 1 grid master ต่อ 1 อาคาร ห้ามผสม coordinate space

---

10. `title`

Input: ภาพหน้า เท่านั้น

Output: ยังไม่มี field-set ยืนยัน (`(draft)`, §1 #10) — ตัวอย่างจริงมีแค่ 5 ไฟล์จาก 5 บ้าน ต้องสำรวจฟิลด์ที่ใช้จริงก่อนล็อก schema ให้แน่น

---

11. `symbol`

Input: ภาพหน้า เท่านั้น

Output: `(draft)` — แนวโน้มคือ `fixture_symbol_legend[]`/`fixture_install_height_standard[]` (§0.1 กล่าวถึงเป็นตาราง reference ที่ไม่พับเข้า `elements[]`) แต่ยังไม่ล็อก schema ชัดเจน

---

12. `roof_plan`

Input: ภาพหน้า + **grid master** (เหมือน `plan` — สันหลังคา/ตะเข้/ชายคาก็อ้างอิงกริดโครงสร้างเดียวกัน)

Output: `(draft)`, §1 #12 — คาดว่าจะมี field คล้าย `plan` แต่เพิ่มมิติชายคา/สันหลังคา `element_type` ที่มีอยู่ (`rafter`) อาจไม่พอ — ต้องสำรวจตัวอย่างจริง 23 ไฟล์ก่อนล็อก

---

13. `misc`

Input: ภาพหน้า เท่านั้น

Output: ถ้าเป็นตารางราคาซีรีส์ 10 แบบ → `series_price_table[]` **รูปเดียวเท่านั้น** (§0.1 — เคยเก็บ 5 แบบต่างกันใน 5 บ้านมาก่อน ห้ามประดิษฐ์แบบใหม่อีก):

field	type
`design`	string
`name`	string
`size_sqm`	number
`price_pile_baht`	number
`price_spread_baht`	number

ถ้าเป็นหน้าปกรวม/โปรโมชันอื่น → `elements[]` ทั่วไปพร้อม `element_type: "misc"` (ไม่มี schema เฉพาะ)

---

14. `bbs_schedule` — ⛔ ไม่มีตัวอย่างเทรนเลย

Input: ภาพหน้า เท่านั้น (ตามดีไซน์)

Output (`elements[]`, §1 #14, ยืมโครง field จาก `QT_PROMPT_BBS_EXTRACT` ของ Constistant เพราะเป็นแบบเดียวที่มีอยู่):

field	type	หมายเหตุ
`element_id`	string	element ที่บาร์นี้เป็นของ (เช่น `"C1"`)
`bar_mark`	string	รหัสบาร์ (เช่น `"T1"`)
`dia_mm`	number	ขนาดเหล็ก
`shape_code`	string	BS8666 (00=ตรง, 11=งอ90°, 51=ปลอกปิด ฯลฯ)
`len_A`/`len_B`/`len_C`	number	มิติงอ เป็น**เมตร** (ต่างจาก `plan`/`section` ที่ mm — ต้องระวัง prompt สับสน)
`qty`	number	จำนวน
`grade`	string	SR24/SD30/SD40

⛔ **ต้องแก้ก่อนเขียน prompt จริง:** ไม่มีไฟล์ raw JSON ไหนใช้ pattern นี้เลยสักไฟล์ (0/1,240) — ต้องหาว่าบ้านไหนมีหน้า BBS จริงในแบบต้นฉบับก่อน ไม่งั้น prompt นี้เป็นแค่การเดา field จาก Constistant โดยไม่มีหลักฐานจริงรองรับ (ตามมติ 08-04 ข้อ 2)

---

15. `soil_boring_log` — ⛔ ไม่มีตัวอย่างเทรนเลย

Input: ภาพหน้า (PDF/รูปรายงานเจาะสำรวจดิน) เท่านั้น

Output (`elements[]` + wrapper พิเศษ, §1 #15, ยืมโครง field จาก `QT_PROMPT_SOIL_BORING_LOG` ของ Constistant):

field	type	หมายเหตุ
`element_type`	string	`"soil_layer"` คงที่
`depth_m`	string	เช่น `"0-1.5"` (ช่วง ไม่ใช่ตัวเล เดี่ยว)
`soil_type`	string	ตามที่รายงานระ ไม่แปล
`spt_n`	string	blow count, `"R"` / `"Refusa` ได้
`moisture_content_pct` / `unit_weight_kn_m3` / `cohesion_kpa` / `friction_angle_deg` / `liquid_limit_pct` / `plastic_limit_pct` / `specific_gravity`	number|null	ผลแล็บ — แปล หน่วยตามที่ Constistant ทำ (t/m³→kN/m³ ×9.80665 ฯลฯ)
wrapper: `borehole_id`	string	ไม่ใช่ element
wrapper: `groundwater_level_m`	number	ค่าเดียวของทั้ง หลุมเจาะ ไม่ใช่ ชั้นดิน

เรียงชั้นดินตื้น→ลึกเสมอ

⛔ **ต้องแก้ก่อนเขียน prompt จริง:** เหมือน `bbs_schedule` — 0 ตัวอย่างจริง ต้องหาบ้านที่มีรายงานเจาะสำรวจดินก่อน (แบบก่อสร้างบ้านทั่วไปมักไม่แนบรายงานนี้ในชุดเดียวกัน อาจต้องหาจากแหล่งอื่น เช่น Site Investigation ของ Constistant ที่มีตัวอย่างจริงอยู่)

สิ่งที่ยังไม่ได้ทำ (สถานะจริง)

- **ยังไม่ได้เขียน prompt text จริงสักไฟล์** — เอกสารนี้คือ I/O contract เตรียมไว้ก่อนเขียน prompt
- `site_plan`'s canonical `element_type` list — ยังไม่ตัดสินใจ (บล็อกก่อนเขียน prompt ของ pattern นี้)
- `title` / `symbol` / `roof_plan` — field-set ยัง draft ทั้งหมด ต้องสำรวจตัวอย่างจริงก่อนล็อก
- `bbs_schedule` / `soil_boring_log` — บล็อกด้วย "0 ตัวอย่าง" ตามมติ 08-04 ข้อ 2 (ต้องขยาย dataset จริง ไม่ใช่เขียน prompt เดาไปก่อน)
- **บล็อกอันดับ 1 ที่ยังไม่เริ่ม:** คานหายบ้าน 06-12 — ตามมติ 08-04 ข้อ 3 priority คือแก้ dataset ก่อน ไฟล์นี้เขียนเสร็จก่อนตามที่ขอ แต่ยังไม่ได้แปลว่า dataset พร้อมแล้ว