

Which work deserves *your best model?*

A discovery prompt that inspects your real work and hands you ready-to-run briefs for the few tasks worth escalating to Claude Fable 5. Plus 9 prompts I am running the moment it is back.

Why this exists

Most people point their most powerful model at whatever is in front of them. That is backwards. The patient, expensive models earn their keep on one kind of work: big, spread across several tools, heavy on judgment, with a clear finish line. Short and obvious work belongs on a faster model or with you.

The skill is not prompting better. It is knowing which work to hand it. This prompt does the sorting. Do not run the work yet. First, find it.

The discovery prompt

```
You are helping me decide which parts of my work are worth escalating to Claude Fable 5. Do not execute any of the work yet. Inspect my context, find strong candidates, and turn the best ones into ready-to-run Fable briefs.
```

```
My role and current goals: [your role, priorities, goals, constraints]  
Use this context: [repos, project folders, docs, Notion, Slack, Linear/Jira, email, calendar, analytics, task lists]  
Constraints: [deadlines, privacy boundaries, tools off-limits, budget, work that must stay human]
```

```
Inspect the available context. Do not brainstorm generic examples. Build an inventory of: active projects, repeated workflows, stalled decisions, messy backlogs, work that spans several tools, and work where better planning, judgment, verification, or follow-through would materially improve the result.
```

```
Score each candidate 1 to 5 on: multi-source context, delegation fit, judgment required, clear finish line, leverage, Fable fit. Recommend Fable only when the task is large enough to justify a slower, more expensive run. Down-rank work that is short, obvious, highly interactive, hard to verify, or better handled by a faster model or a human.
```

```
Return: the top 10 ranked use cases, evidence for each, why each is or is not worth Fable, the deliverable, the verification method, the context/tools/permissions needed, and the risks or human decisions. Then a ready-to-run brief for the top three (problem, outcome, sources, constraints, workflow, human checkpoints, evidence required). Stop after the list and the three briefs. Do not start any task until I choose one.
```

How to use it

1. Paste it into Claude.
2. Fill the three brackets: your role, sources, constraints.
3. Run it. You get a ranked shortlist and three briefs.
4. Pick one and hand it over.

Bonus: 9 prompts I am loading for the moment Fable is back

Fable is about to return. I am preparing every prompt I will send it the moment it comes back. I recommend you do the same. Steal these.

1. Review all the code written after the date Fable was banned. Look for optimizations and improvements you can make.
2. Walk through every major user path in the app you are building using browser control. Write a report on where users get confused and what to improve.
3. Make a checklist of every task you do from now until tonight. Feed it to Fable and ask what it could automate.

4. Feed it all your goals, ambitions, interests, skillsets, and assets. Ask what simple businesses it could help build over the next weeks so you can make your first dollar online.
5. Connect to the X MCP and find 5 helpful use cases other people are using Fable for that are relevant to your workflows.
6. Connect it to the X MCP. Have it read your last 100 posts and come up with 5 SaaS ideas you could build.
7. /loop it every 24 hours to run a security check on all the API endpoints in your existing apps.
8. Use the Unreal 5.8 MCP to build incredible, in-depth 3D games.
9. Go to Sonnet 5 and ask, based on what it knows about you, what incredible prompts you could give Fable tonight.

Keep going. For more operator playbooks like this, subscribe to The Blink at chelseahulin.com.