

USER MANUAL

PinchFin MCP for Claude Code

GETTING STARTED

PinchFin runs inside Claude Code as an MCP (Model Context Protocol) integration. You interact with PinchFin by talking to Claude in plain English. Claude recognizes when a PinchFin tool is relevant and calls it automatically — or you can request specific tools directly.

Prerequisites

Claude Code
Python
PinchFin Package
API Key

Pro or Max subscription (claude.ai/download)
3.10 or newer (python.org/downloads)
`pip install git+https://github.com/Pinchfin/pinchfin-mcp.git`
Provided in your beta invite

Configuration

Add PinchFin to your Claude Code config file (`~/.claude.json` on Mac/Linux, `C:\Users\YOUR_NAME\.claude.json` on Windows). Add the following inside the "mcpServers" section:

CLAUDE.JSON CONFIGURATION

```
"pinchfin": {
  "type": "stdio",
  "command": "python",
  "args": ["-m", "pinchfin_mcp.server"],
  "env": {
    "PINCHFIN_API_URL": "https://pinchfin-api-production.up.railway.app/v1",
    "PINCHFIN_API_KEY": "YOUR_API_KEY_HERE"
  }
}
```

Windows users: change "python" to "py" if python is not in your PATH. After saving, restart Claude Code completely.

HOW TO USE PINCHFIN

PinchFin is conversational. You describe what you need in plain English, and Claude determines which PinchFin tools to call. You can also request specific tools directly.

Natural Language (Recommended)

Just describe the deal and what you want to know. Claude will select the right tool, show you the cost estimate, and ask for confirmation before running.

What You Type	What Happens
"Screen this deal: 200 Main St, Austin TX, 150 units, \$30M loan"	Claude calls <code>pinchfin_deal_screen</code> with your parameters
"How much can I borrow on a \$60M asset with \$3.2M NOI?"	Claude calls <code>pinchfin_loan_sizer</code> or <code>pinchfin_debt_sizing_preview</code>
"Run sponsor diligence on Greystar Real Estate Partners"	Claude calls <code>pinchfin_sponsor_analysis</code>
"What does the multifamily market look like in Dallas?"	Claude calls <code>pinchfin_market_research</code>
"Compare these two term sheets" (paste term sheet details)	Claude calls <code>pinchfin_term_sheet_comparator</code>
"Give me a full screening on this deal" (provide deal details)	Claude calls <code>pinchfin_quick_review</code> (orchestrated workflow)
"Size a pref equity position on this deal — \$16.5M senior, \$24.5M cost..."	Claude calls <code>pinchfin_pref_equity_qr</code>

Direct Tool Requests

You can also request specific tools by name if you know exactly what you want.

What You Type	Tool Called
"Run <code>pinchfin_deal_screen</code> on 500 Oak Ave..."	<code>pinchfin_deal_screen</code>
"Check my PinchFin balance"	<code>pinchfin_account</code>
"What PinchFin tools are available?"	<code>pinchfin_help</code>
"How much would a sponsor analysis cost?"	<code>pinchfin_estimate_cost</code>

RECOMMENDED WORKFLOWS

1. Quick Screening (1–2 minutes, ~\$0.15)

Start every deal here. Fast, cheap, tells you whether to spend more time.

EXAMPLE PROMPT

Screen this deal: 5700 Clemson Dr, Dallas TX 75214. 165-unit Class B multifamily, 1985 vintage. Sponsor: ABC Realty Group. Loan request: \$42M acquisition.

You get: pass/fail against 6 credit constraints, binding constraint identified, property context data, risk flags, and a narrative summary.

2. Targeted Deep Dive (5–10 minutes, ~\$1–3)

When you need depth on a specific dimension — run individual tools.

Scenario	Tool to Use
New sponsor, need background	pinchfin_sponsor_analysis
Unfamiliar market	pinchfin_market_research
Need rent/sale comps	pinchfin_comp_analysis
Value-add feasibility check	pinchfin_business_plan_analysis
Comparing lender offers	pinchfin_term_sheet_comparator
Complex deal features (LIHTC, ground lease)	pinchfin_nuance_gate
Preferred equity sizing	pinchfin_pref_equity_qr

2A. Preferred Equity Sizing (~\$3–8)

Size a preferred equity position behind an existing or proposed senior loan. Validates the senior debt independently, then sizes pref against 9 constraints with PIK accrual modeling and exit scenario analysis.

EXAMPLE PROMPT

Size a pref equity position on this deal: 2800 Telegraph Ave, Oakland CA. 165-unit new construction, \$24.5M total cost, \$16.5M senior loan at 7.50%. Stabilized NOI \$1.5M, exit cap 5.25%. Sponsor requesting \$2.5M pref at 7% current / 7% PIK. 24 months construction, 9 months lease-up.

You get: max supportable pref equity, binding constraint, quarterly PIK schedule, 5 exit scenarios (base, $\pm 10\%$, -20%, breakeven), sponsor MOIC/IRR with and without pref, return sensitivity by exit timing, and a 22-section DOCX memo.

3. Full Quick Review (10 minutes, ~\$3–6)

The flagship workflow. Internally runs sponsor + market + comps + cash flow + sizing and synthesizes everything into a screening recommendation with 4 DOCX reports.

EXAMPLE PROMPT

Run a full Quick Review on 123 Main St, Dallas TX 75201. 165-unit multifamily, Class B, 1985 vintage. \$42M loan request for acquisition. Sponsor is XYZ Realty Group. Current NOI around \$2.5M, pro forma rents ~\$1,800/unit.

You get: screening recommendation (PASS/CONDITIONAL/FAIL), sponsor tier, market rating, comp-supported rent conclusions, constraint waterfall, risk assessment, and due diligence questions.

4. Iterative Analysis

PinchFin tools work within your Claude Code conversation. You can chain analyses naturally:

CONVERSATION FLOW

You: Screen this deal — 200 units, \$25M, Austin TX

Claude: [runs deal screen] Deal passes. Binding constraint: DSCR at 1.08x.

You: The sponsor is new to us. Run a background check.

Claude: [runs sponsor analysis] Sponsor rated B+. 12 prior deals...

You: Looks good. How does the Austin market look?

Claude: [runs market research] Austin multifamily: 4.2% vacancy, 3.1% rent growth...

You: Size the loan at our constraints.
Claude: [runs loan sizer] Max loan \$23.8M, binding: DSCR at 1.05x...

UNDERSTANDING YOUR RESULTS

JSON Results

Every tool returns structured JSON directly in your Claude Code conversation. Claude will interpret and summarize the results for you. Key fields to look for:

Field	What It Means
status: pass/fail	Whether the deal meets credit box criteria
binding_constraint	The most restrictive constraint (governs max loan)
sponsor_tier: A/B/C/D	Sponsor quality classification
market_rating: strong/moderate/weak	Overall market assessment
screen_result: PASS/CONDITIONAL/FAIL	Quick Review recommendation
go_no_go: GO/NO-GO/CONDITIONAL	Nuance gate or sponsor determination

DOCX Reports

When analysis runs in real mode, branded PinchFin DOCX reports are automatically generated and available for download. Reports use institutional formatting with the PinchFin v9 design system — ready to share with investment committees, credit officers, or capital partners.

Credit Box Constraints

PinchFin uses a 6-constraint waterfall for all loan sizing. The most restrictive constraint governs the maximum loan amount.

Constraint	Threshold Type
LTV As-Is	75.0% Hard ceiling
LTV Stabilized	75.0% Hard ceiling
DSCR Stabilized	1.05x Hard floor (sizes the loan)
Debt Yield Stabilized	7.00% Hard floor
LTC	80.0% Hard ceiling
Max Loan	\$100M Program cap

MANAGING YOUR ACCOUNT

Action	What to Say
Check balance	"Check my PinchFin balance" or "How much credit do I have?"
See transaction history	"Show my PinchFin transactions"
Get a cost estimate	"How much would a sponsor analysis cost for a 200-unit deal?"
List your deals	"List my PinchFin deals"
Check deal progress	"What's the status of deal PF-2026-001?"

TIPS FOR BEST RESULTS

1. Start with a screen. At \$0.10–\$0.50, it's the cheapest way to filter deals before committing to deeper analysis.
2. Provide complete deal details. The more context you give (address, units, loan amount, sponsor name, NOI, business plan), the better the analysis.
3. Use the Quick Review for pipeline screening. At \$3–6, it replaces hours of manual work with a structured recommendation in 10 minutes.
4. Re-runs are 35% cheaper. If you need to re-analyze a deal with updated assumptions, the system applies a discount for cached context.
5. Chain tools in conversation. Run a screen first, then drill into sponsor or market as needed. Claude maintains context across the conversation.
6. Every billable tool shows cost before running. You always confirm. No surprises.

TROUBLESHOOTING

Issue	Fix
PinchFin tools not showing up	Verify ~/.claude.json has valid JSON with pinchfin in mcpServers. Restart Claude Code completely (close and reopen).
Authentication failed	Double-check your API key — no extra spaces or quotes. Make sure you pasted the full key starting with pf_
Insufficient funds	Ask "check my PinchFin balance" to see remaining credits. Contact your PinchFin admin for additional credits during beta.
"python" not found (Windows)	Change "command": "python" to "command": "py" in your claude.json config.
Tool returns mock/simulated data	This means the server is running in fallback mode. Contact support — the analysis engine may need to be restarted.

SUPPORT

For questions, issues, or feedback during the beta, contact your PinchFin admin directly. We want to hear everything — what worked, what broke, what you wish it did.