

Claude Code vs. Google Antigravity: Which AI Coding Tool Should You Learn?

This guide breaks down the key differences, live test results, and final verdict on the two leading AI coding tools so you can decide which one to invest your time in.

By: [Nate Herk](#)

I. What Are These Tools?

Both Claude Code and Google Antigravity are agentic coding platforms. They can take a large mission, break it into a plan, spin up sub-agents to work in parallel, manage your files and folders, run terminal commands, and execute across your entire codebase.

Claude Code is powered by Anthropic's Claude models (default: Opus 4.6).

Antigravity defaults to Google's Gemini models (default: Gemini 3 Pro).

You can swap out the model on both tools, but the default model largely determines the quality ceiling you will experience. Both tools are free to use on their own. What you pay for is the model usage (the tokens) powering the AI behind the scenes.

II. Where Do You Build?

Claude Code is terminal-first. It is a CLI tool that plugs into whatever environment you are already working in. Your editor, key bindings, and extensions all stay the same. The agentic power (sub-agents, parallel execution, orchestration) is surfaced through your terminal and configured by you. You build the workflow on top of the tool.

Claude Code offers multiple entry points:

- **Terminal CLI** - The original and most powerful mode. Install with npm and type `claude` in any terminal.
- **VS Code Extension** - Inline diffs, plan review, file visibility, app mentions, all inside your editor.
- **Desktop App** - A more user-friendly standalone option.
- **Web Version** - Run straight from your browser with no install.

The terminal CLI is technically the most feature-complete mode, but the VS Code extension covers the vast majority of what you need.

Antigravity is its own standalone IDE. It is thought to be a fork of VS Code, so extensions and shortcuts can carry over, but you are opening a different application entirely. It includes a manager view for watching multiple agents work in parallel and a built-in browser agent that can navigate real web pages. There is no "run Antigravity inside VS Code" option.

The simplest way to think about it: Claude Code gives you the building blocks and lets you work the way you already work. Antigravity packages the whole agentic workflow into a purpose-built environment that you move into.

III. Output Quality

The model you use inside the tool matters enormously. Running Claude Code with Haiku is a completely different experience than running it with Opus. The same applies to Antigravity with Gemini Flash vs. Gemini Pro. The harness shapes how the model works, but the model determines the ceiling.

Planning

Both tools have a dedicated planning mode where the AI maps out multi-file work before touching any code (read-only, no code changes, just strategy and clarification questions). Claude Code's planning mode is notably stronger. You can control how deep it reasons, all the way up to what Anthropic calls "ultra-think" for complex problems.

Project Understanding

Claude Code excels at understanding your existing project and file structure. It reads everything, picks up on your patterns and naming conventions, and builds on top of what you already have. Generated code fits your codebase rather than feeling bolted on.

Design and Frontend

Antigravity's major strength is building full apps and frontends from scratch. In an independent 21-day test across 12 real projects, 94% of the code it generated was clean, passing linting with no errors or style issues. It completed about 73% of tasks without any human intervention and delivered a 60-70% reduction in development time. Antigravity tends to have better "taste" when it comes to UI/UX, making things look and feel more polished out of the box.

The Trade-off

On longer projects, Antigravity can start to drift away from the rules you set at the beginning. This is a documented pattern (Google's own developer forums have threads about it). It might build something perfectly to spec on day one and gradually start ignoring your guidelines by day three. This can happen with any coding agent, but it is more pronounced with Antigravity.

Benchmarks

The SWE-bench Verified test (real GitHub issues from real open-source projects) shows the top-tier models are close:

Tool + Model	SWE-bench Score
Claude Opus 4.6 (Claude Code)	80.9%
Gemini 3 Pro (Antigravity)	76.2%

Google has not published exact testing methodology the way Anthropic has, so this is not a perfect comparison, but directionally the tools are remarkably close at the top tier.

Real-World Productivity

- **Claude Code:** Anthropic's engineering team reported a 50% productivity gain and 67% more pull requests merged per engineer per day.
- **Antigravity:** The independent 21-day test showed 60-70% faster development.

Neither is a controlled scientific study, but both indicate significant speed improvements.

IV. Speed, Reliability, and Token Management

Token Costs

Both tools can burn through your usage limits faster than you expect if you are not careful. Claude Code had a caching bug in March 2026 that inflated token costs by 10-20x, draining Max plan sessions in under two hours. Antigravity has its own quota issues, with Pro users getting locked out for a full week after hitting limits and a credit-based pricing system that does not clearly explain what a credit buys you.

Key takeaway: Learning how to manage your context and tokens is one of the most important skills you can build right now. It translates to whatever coding agent you end up using.

Task Speed

An independent test found Claude Code completed a representative task in about 4 minutes vs. Antigravity's 8 minutes. However, Antigravity can feel faster in practice during real-world building sessions.

Maturity

Claude Code is production-released with multiple updates shipping per week. Antigravity is still in public preview (as of April 2026) and carries the instability that comes with that stage, including login issues, occasional Windows bugs, and agents getting stuck in loops.

Context Loss

Context loss is real for both tools. Even with Claude Code's 1 million token context window, issues arise when you are loading real codebases, running lots of tool calls, and going back and forth for 40+ prompts. Antigravity's own docs recommend starting a new chat once the agent takes more than 10 seconds to respond. Claude Code has a `/compact` command that compresses your conversation history.

Best practice for both tools: Keep your sessions focused. One task per session. Start fresh often.

Development Pace

Claude Code shipped six major platform-level features in Q1 2026 alone, with multiple releases per week (at one point, three releases in 5 days). Antigravity went from version 1.11 to 1.21 over roughly four to five months, with most updates being minor improvements and bug fixes. When choosing a tool to invest your time in, the pace of development matters.

V. Integrations (MCP and Beyond)

Both tools support MCP (Model Context Protocol), the open standard that lets AI tools plug into external services like GitHub, databases, Supabase, Playwright, Firecrawl, design tools, monitoring platforms, and more. Over 1,500 MCP servers exist and the number is growing.

Claude Code's MCP setup is CLI-driven. You add servers with a single command, prompt Claude Code to do it, or edit a JSON config file scoped globally, per project, or per user.

Antigravity's MCP setup uses a visual panel built into the IDE where you can browse a marketplace and click to install. It is more beginner-friendly to start, but for advanced setups you still end up editing a config file.

Both tools have direct terminal access, meaning they can use any CLI tool: Git, npm, Docker, Playwright, Google Workspace. If it has a CLI and runs in a terminal, either tool can use it. You can even run Claude Code's CLI inside Antigravity's terminal if you want both tools in one environment.

VI. Pricing Comparison

Before looking at the numbers, consider the framing: even at \$200/month, the level of output you get from either tool is something you could never hire a human to do for the same price. No matter which tool you choose, you are getting an insane amount of leverage for the money.

Claude Code Pricing

Plan	Cost	Details
Pro	\$20/month	Entry-level access. Limit burns fast with active building.
Max (Standard)	\$100/month	5x more usage than Pro.
Max (High)	\$200/month	20x more usage than Pro. Most consistent experience.

You can also use your API key (may be more expensive than a subscription) or plug in a cheaper/open-source model (but you lose the value of Opus).

Antigravity Pricing

Plan	Cost	Details
Free	\$0	Basic access with weekly rate limits. Runs out fast with real work.
Google AI Pro	\$20/month	Higher rate limits (refresh every 5 hours), 2TB Google storage, \$10 in Google Cloud credits.
Google AI Ultra	\$250/month	Highest rate limits.

Important note: Even on Pro, rate limits for premium models like Claude Opus inside Antigravity are very restrictive. Some users report getting locked out after a single session.

Bottom Line on Cost

- **Just getting started?** Antigravity's free tier lets you experiment at zero cost.

- **Reliable daily production use?** Claude Code on Max gives the most consistent experience right now.
- **At the highest tiers** (\$200 for Claude Code, \$250 for Antigravity), they are in a similar range and both deliver far more output than hiring a developer.

VII. Live Test Results

Three head-to-head tests were run with Claude Code (Opus 4.6) vs. Antigravity (Gemini 3.1 Pro), using the same prompt for each.

Test 1: One-Shot Habit Tracker App (No Planning)

Prompt: Build a full-stack habit tracker app with a dashboard that shows streaks, a calendar heat map, and the ability to add, edit, and delete habits.

Category	Claude Code	Antigravity
Speed	Slower	Faster (finished first)
Logic/Functionality	Better user journey and more complete feature set	Functional but felt more limited
Design	Looked more "vibe-coded"	Noticeably better visual design
Initial Run	Worked on first load	Blank white page (required a fix)

Test 2: AI Trends PDF Report (Plan Mode)

Prompt: Research AI trends in the SMB market and create a 3-page PDF report with a holistic breakdown.

Category	Claude Code	Antigravity
Planning	Created a detailed plan before executing	Skipped planning entirely and went straight to output
Clarification	Did not ask questions (but neither did Antigravity)	Did not ask questions

Output Quality	Better styling, headers, colors, tables	Basic formatting, no styling
Sources	Cited sources at the end	Did not cite any sources
Winner	Claude Code	-

Test 3: Landing Page Design (No Planning, No Questions)

Prompt: Design a landing page for a premium AI automation course targeted at agency owners. Include a hero section, social proof, a pricing table, and a CTA.

Category	Claude Code	Antigravity
Copy/Structure	Better written copy and page structure	Good but not as strong
Design/Feel	Decent but clearly vibe-coded	Stronger animations, depth, floating elements, better colors
Winner (Design)	-	Antigravity

VIII. The Final Verdict

A clear pattern emerged across all three tests:

Claude Code thinks better. It plans more thoroughly, writes better copy, structures projects more logically, and cites sources. It is the stronger tool for building automations, skills, products, and anything where code quality and reliability matter.

Antigravity makes things look better. It consistently produces more polished UI/UX, better visual design, and smoother animations right out of the box. It is the stronger tool for web app design, websites, and frontend-heavy work. Gemini is also noticeably faster.

Rule of Thumb

Use Antigravity if: Design and speed are your priority, or you want to test something out for free before you commit.

Use Claude Code if: You care about real code quality, configurability, and production reliability.

The Bigger Picture

It is not always about which tool is "better" overall. The real question is: for this specific job, which tool does the job best? Or which combination of tools can you use for different pieces of the job?

With Claude Code's custom skills and community instructions, you can close the design gap over time. And the pace at which Claude Code is shipping updates makes it increasingly hard to beat.

Want to connect with others building and monetizing AI automation?

[Become an AIS Plus Member](#)