GPT-5: The Complete Guide

OpenAI's Latest Model Deep Dive

Lucas Soares

September 26, 2025

Table of Contents

Table of Contents

Table of Contents

Table of Contents

Table of Contents

Table of Contents

Table of Contents

Table of Contents

GPT-5 Overview & Timeline

GPT-5 Release Timeline

Key Dates & Milestones

GPT-5 Release Timeline

Key Dates & Milestones

- **Launch Date**: August 7, 2025

GPT-5 Release Timeline

Key Dates & Milestones

- **Launch Date**: August 7, 2025
- **Training Infrastructure**: Microsoft Azure AI supercomputers with NVIDIA H200 GPUs

GPT-5 Release Timeline

Key Dates & Milestones

- **Launch Date**: August 7, 2025
- **Training Infrastructure**: Microsoft Azure AI supercomputers with NVIDIA H200 GPUs
- **Availability Rollout**:
  - ChatGPT Free, Plus, Pro tiers ✅
  - Team users ✅
  - Enterprise and Edu (coming next)
  - OpenAI API ✅

GPT-5 Release Timeline

Key Dates & Milestones

- **Launch Date**: August 7, 2025
- **Training Infrastructure**: Microsoft Azure AI supercomputers with NVIDIA H200 GPUs
- **Availability Rollout**:
  - ChatGPT Free, Plus, Pro tiers ✅
  - Team users ✅
  - Enterprise and Edu (coming next)
  - OpenAI API ✅
- **Major Breakthrough**: First unified system combining reasoning, multimodal input, and task execution

GPT-5 Core Inovations

GPT-5 Core Inovations

- **Unified Architecture**:
  - Single system with multiple specialized models

GPT-5 Core Inovations

- **Unified Architecture**:
  - Single system with multiple specialized models
- **Smart Router**:
  - Automatically selects the best model for each task

GPT-5 Core Inovations

- **Unified Architecture**:
  - Single system with multiple specialized models
- **Smart Router**:
  - Automatically selects the best model for each task
- **Reduced Hallucinations**:
  - 45% fewer factual errors than GPT-4o

GPT-5 Core Inovations

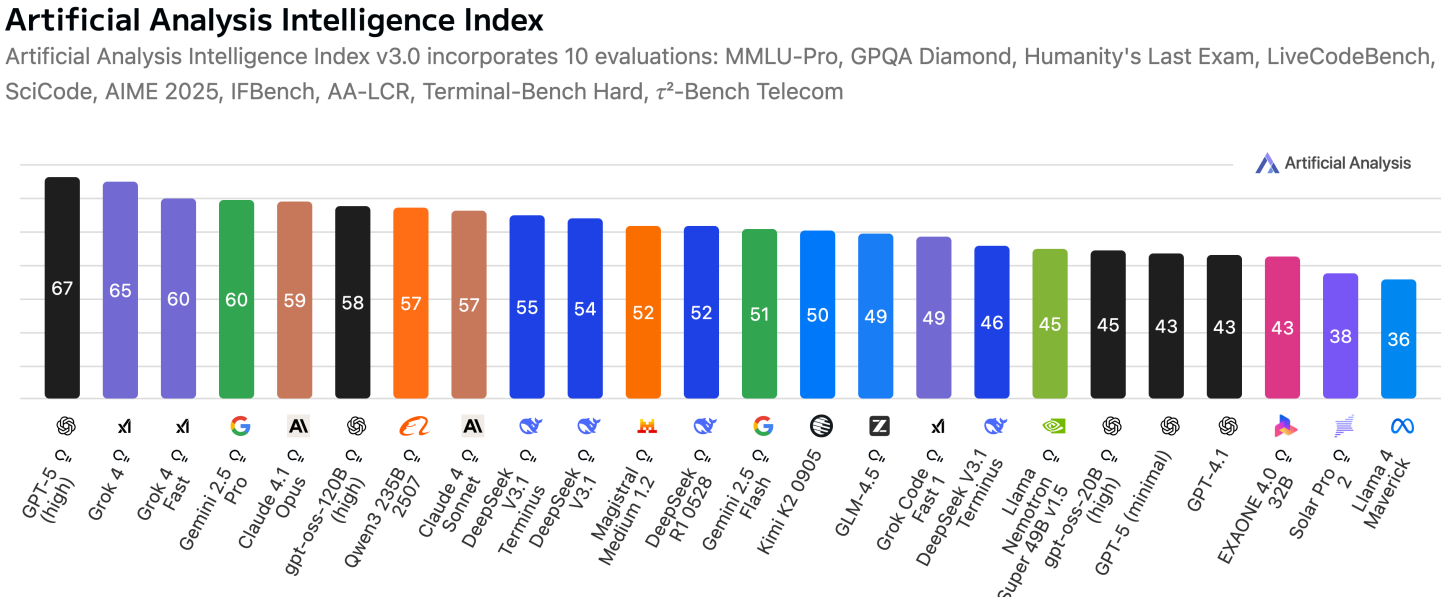
- **Unified Architecture**:
  - Single system with multiple specialized models
- **Smart Router**:
  - Automatically selects the best model for each task
- **Reduced Hallucinations**:
  - 45% fewer factual errors than GPT-4o
- **Enhanced Tool Use**:
  - Reliably chains dozens of tool calls

GPT-5 Core Inovations

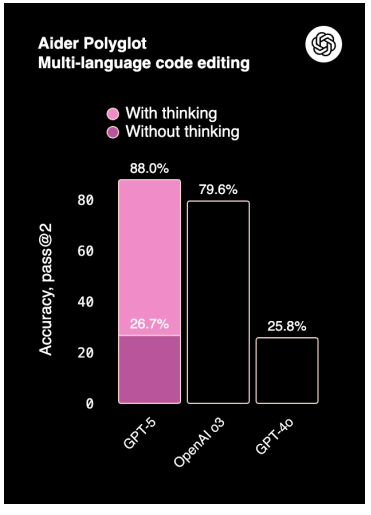
- **Unified Architecture**:
  - Single system with multiple specialized models
- **Smart Router**:
  - Automatically selects the best model for each task
- **Reduced Hallucinations**:
  - 45% fewer factual errors than GPT-4o
- **Enhanced Tool Use**:
  - Reliably chains dozens of tool calls
- **Multimodal Excellence**:
  - Native text, image, audio, and video support

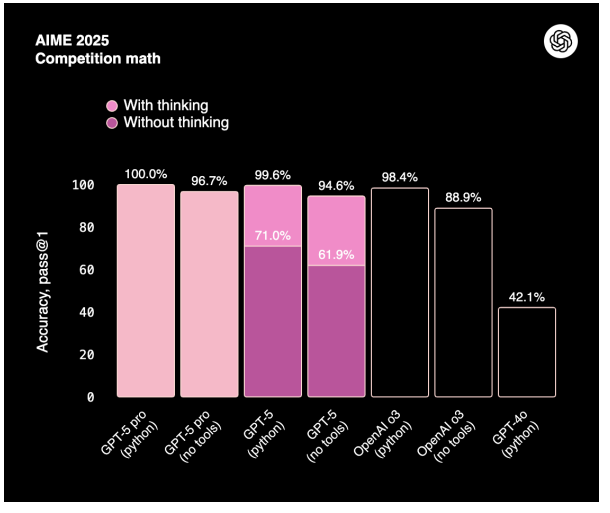
Model Comparison & Performance

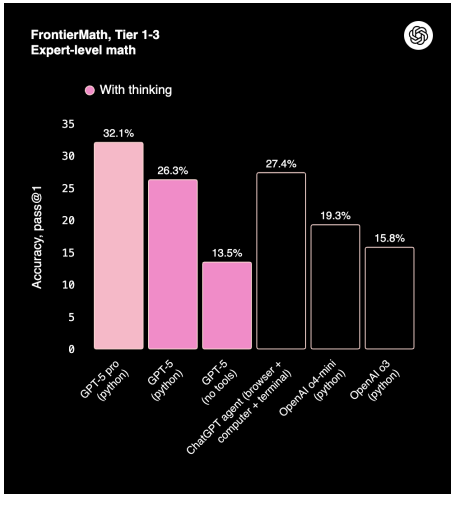
GPT-5 SOTA in Artificial Analysis Intelligence Index



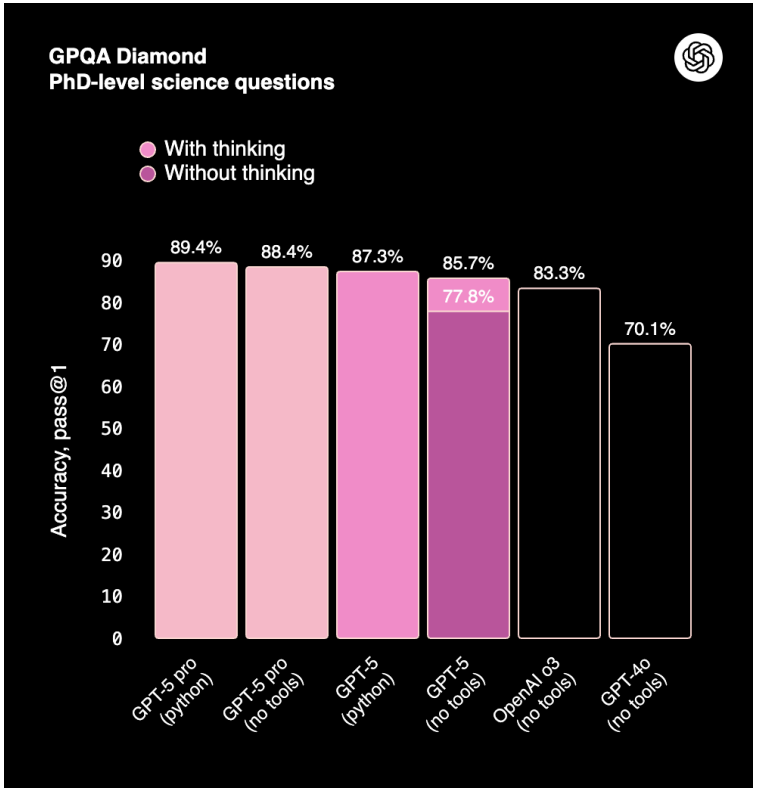
Source: https://openai.com/index/introducing-gpt-5/



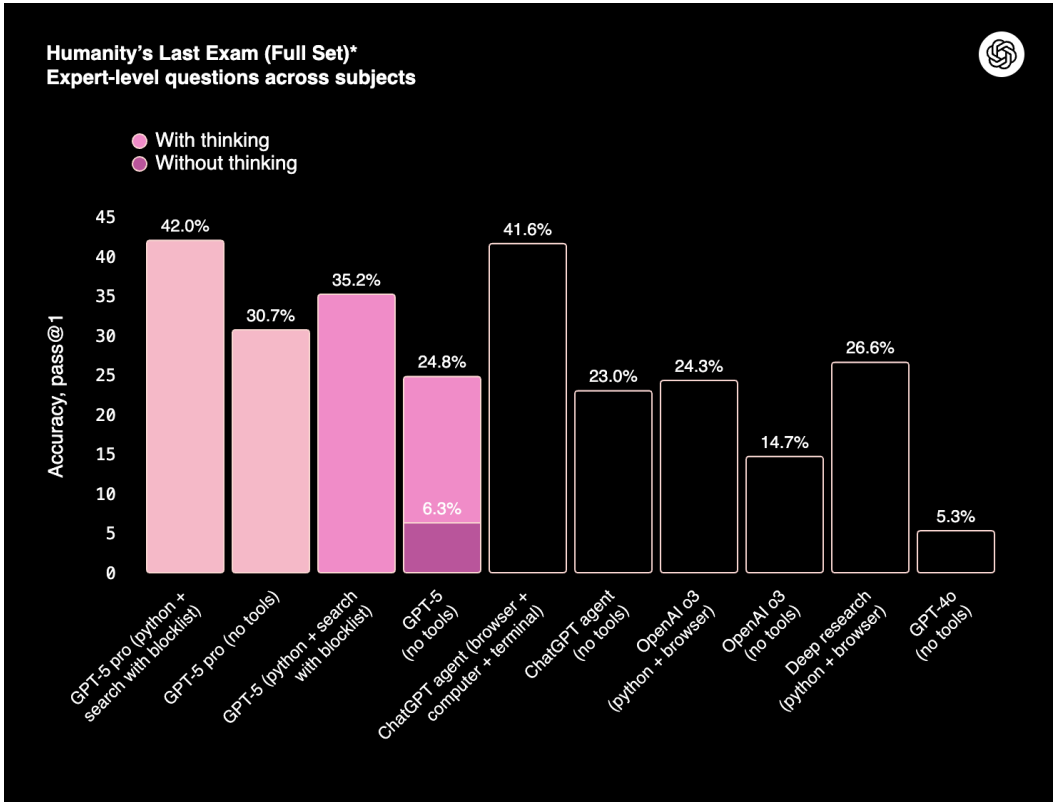




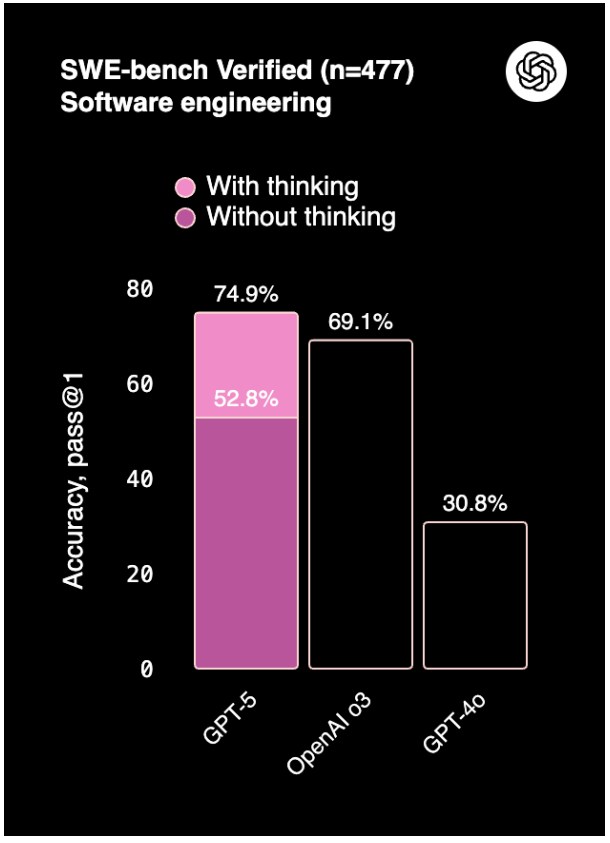
Source: https://openai.com/index/introducing-gpt-5/



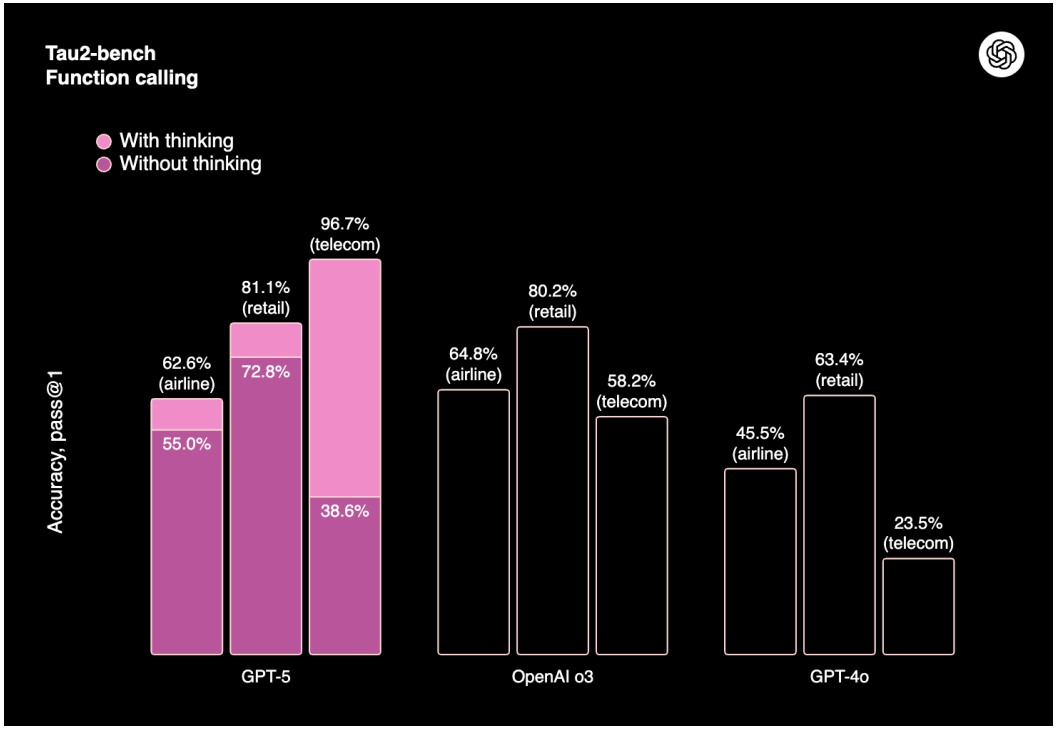
Source: https://openai.com/index/introducing-gpt-5/



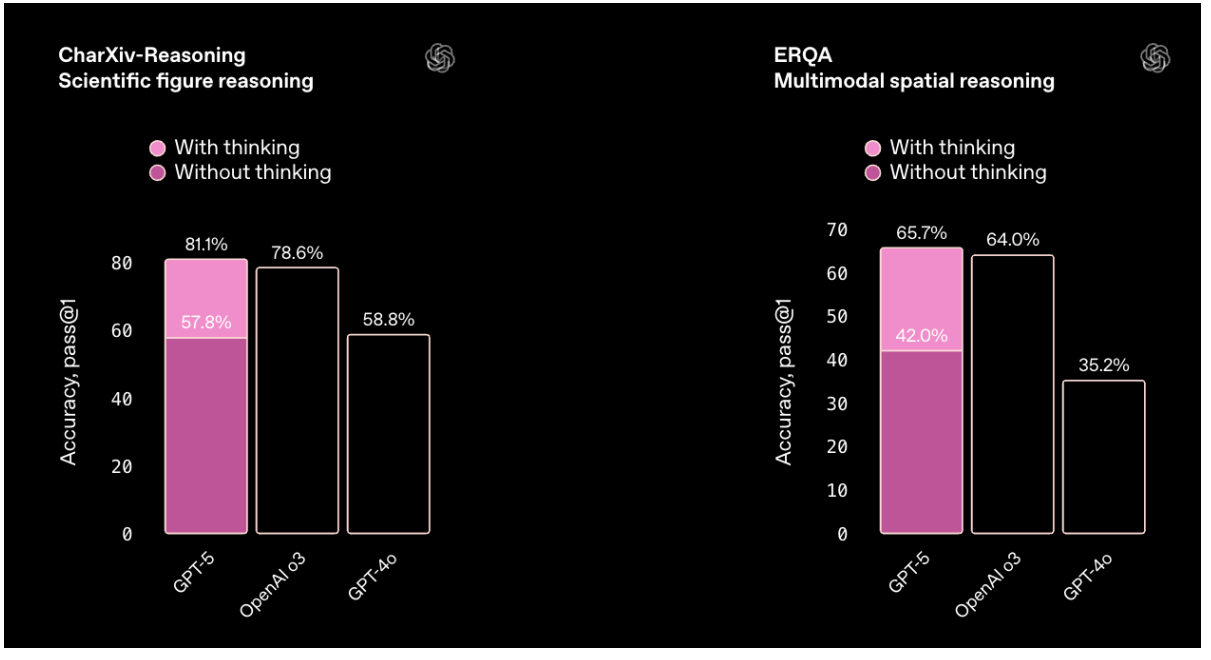
Source: https://openai.com/index/introducing-gpt-5/



Source: https://openai.com/index/introducing-gpt-5/

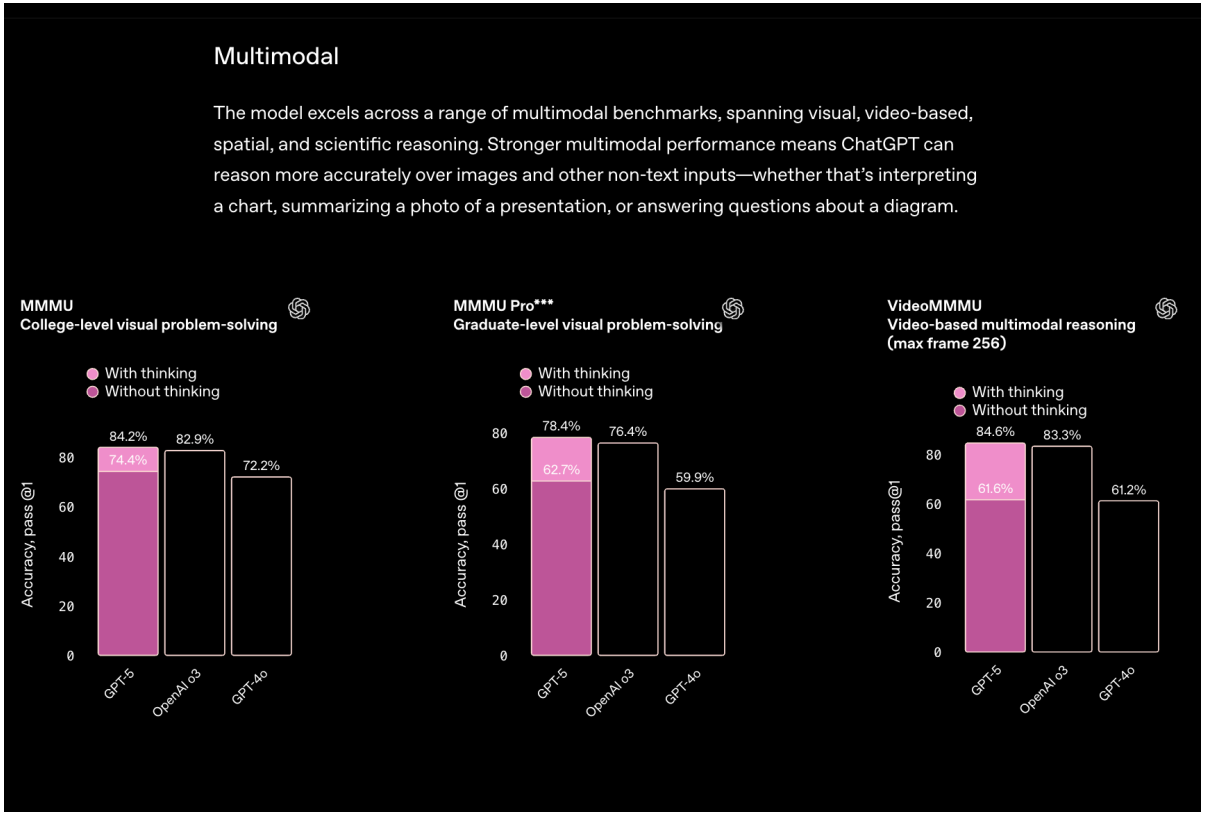


Source: https://openai.com/index/introducing-gpt-5/



Source: https://openai.com/index/introducing-gpt-5/

Source: https://openai.com/index/introducing-gpt-5/



Source: https://openai.com/index/introducing-gpt-5/

Overall Performance Summary

Overall Performance Summary

Mathematics: 94.6% on AIME 2025 (without tools)

Overall Performance Summary

Mathematics: 94.6% on AIME 2025 (without tools)

Coding: 74.9% on SWE-bench Verified, 88% on Aider Polyglot

Overall Performance Summary

Mathematics: 94.6% on AIME 2025 (without tools)

Coding: 74.9% on SWE-bench Verified, 88% on Aider Polyglot

Multimodal: 84.2% on MMMU understanding

Overall Performance Summary

Mathematics: 94.6% on AIME 2025 (without tools)

Coding: 74.9% on SWE-bench Verified, 88% on Aider Polyglot

Multimodal: 84.2% on MMMU understanding

Healthcare: 46.2% on HealthBench Hard (vs 31.6% for OpenAI o3)

Overall Performance Summary

Mathematics: 94.6% on AIME 2025 (without tools)

Coding: 74.9% on SWE-bench Verified, 88% on Aider Polyglot

Multimodal: 84.2% on MMMU understanding

Healthcare: 46.2% on HealthBench Hard (vs 31.6% for OpenAI o3)

Agentic Capabilities:

- 74.9% on SWE-bench Verified (with thinking)
- Performs amazingly on tasks related to instruction following and agentic tool use (e.g. BrowserComp)

Overall Performance Summary

Mathematics: 94.6% on AIME 2025 (without tools)

Coding: 74.9% on SWE-bench Verified, 88% on Aider Polyglot

Multimodal: 84.2% on MMMU understanding

Healthcare: 46.2% on HealthBench Hard (vs 31.6% for OpenAI o3)

Agentic Capabilities:

- 74.9% on SWE-bench Verified (with thinking)
- Performs amazingly on tasks related to instruction following and agentic tool use (e.g. BrowserComp)

Reduced Errors:

- 45% fewer factual errors than GPT-4o
- With thinking mode: 80% fewer factual errors than OpenAI o3

Ok, great the model does well on a lot of benchmarks, but what is new about it?

GPT-5 System Card

Notes on GPT-5's System Card

Notes on GPT-5's System Card

Real-time Router

- Intelligently decides which model to use
- Based on conversation type, complexity, and tools needed

GPT-4o

Great for most tasks

GPT-4.5

Good for writing

o3

Uses advanced reasoning

o3-pro

Best at reasoning

o4-mini

Fastest at advanced reasoning

o4-mini-high

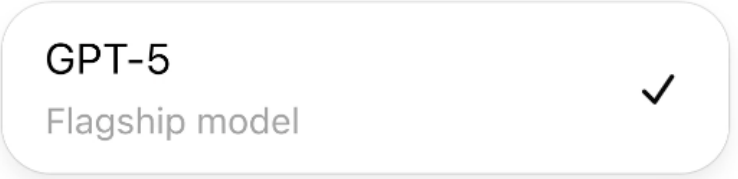
Great at coding and reasoning

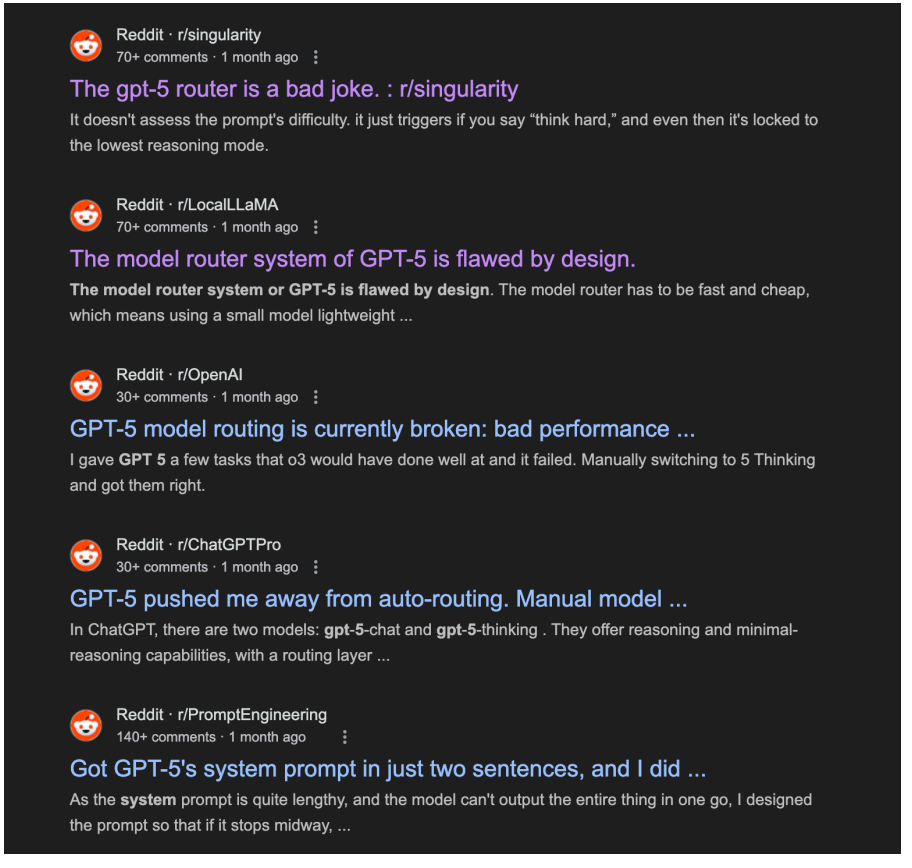
GPT-4.1

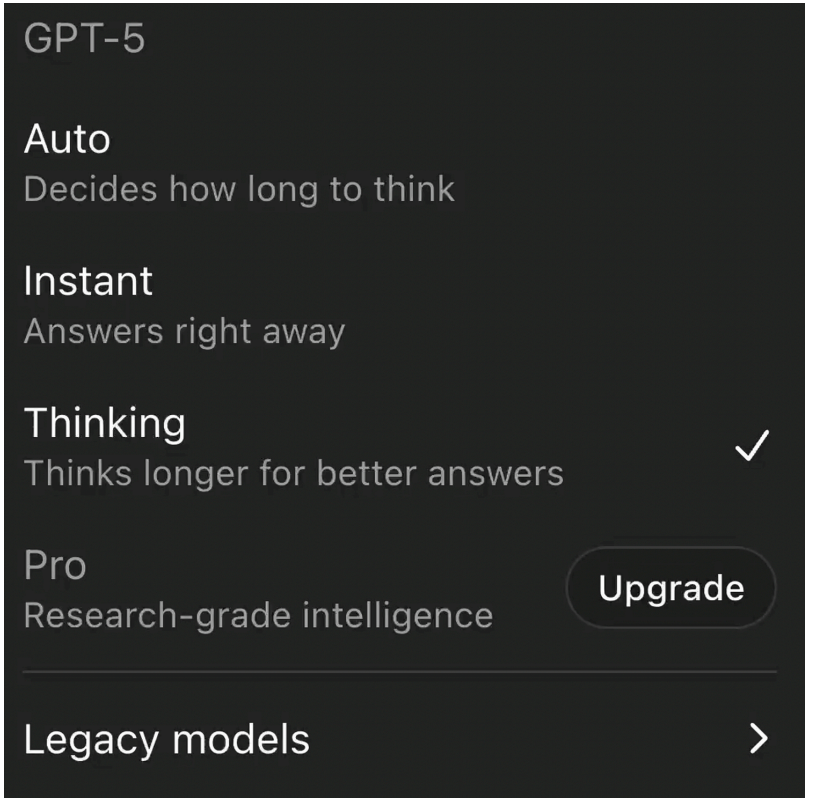
Great for quick coding

GPT-4.1-mini

Faster for everyday tasks


GPT-5
Flagship model


Reddit · r/singularity
70+ comments · 1 month ago
The gpt-5 router is a bad joke. : r/singularity
It doesn't assess the prompt's difficulty. it just triggers if you say “think hard,” and even then it's locked to the lowest reasoning mode.
Reddit · r/LocalLLaMA
70+ comments · 1 month ago
The model router system of GPT-5 is flawed by design.
The model router system or GPT-5 is flawed by design. The model router has to be fast and cheap, which means using a small model lightweight ...
Reddit · r/OpenAI
30+ comments · 1 month ago
GPT-5 model routing is currently broken: bad performance ...
I gave GPT 5 a few tasks that o3 would have done well at and it failed. Manually switching to 5 Thinking and got them right.
Reddit · r/ChatGPTPro
30+ comments · 1 month ago
GPT-5 pushed me away from auto-routing. Manual model ...
In ChatGPT, there are two models: gpt-5-chat and gpt-5-thinking . They offer reasoning and minimal-reasoning capabilities, with a routing layer ...
Reddit · r/PromptEngineering
140+ comments · 1 month ago
Got GPT-5's system prompt in just two sentences, and I did ...
As the system prompt is quite lengthy, and the model can't output the entire thing in one go, I designed the prompt so that if it stops midway, ...


GPT-5
Auto
Decides how long to think
Instant
Answers right away
Thinking
Thinks longer for better answers
Pro
Research-grade intelligence
Upgrade
Legacy models

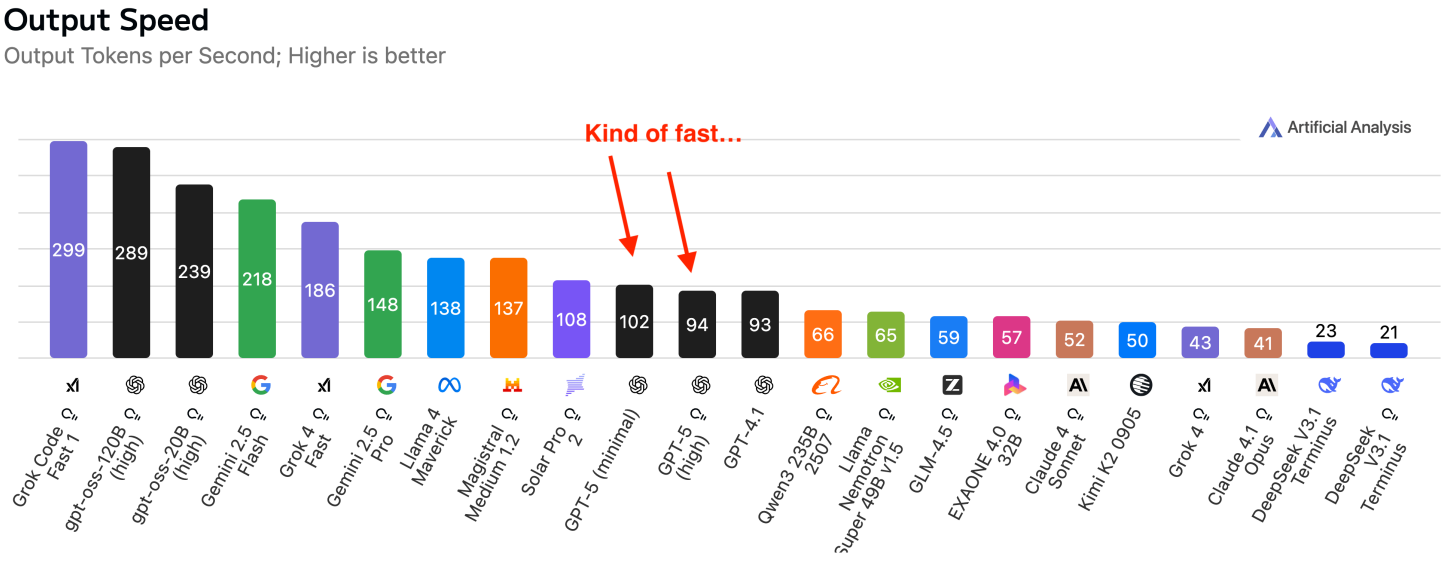
Notes on GPT-5's System Card

Real-time Router

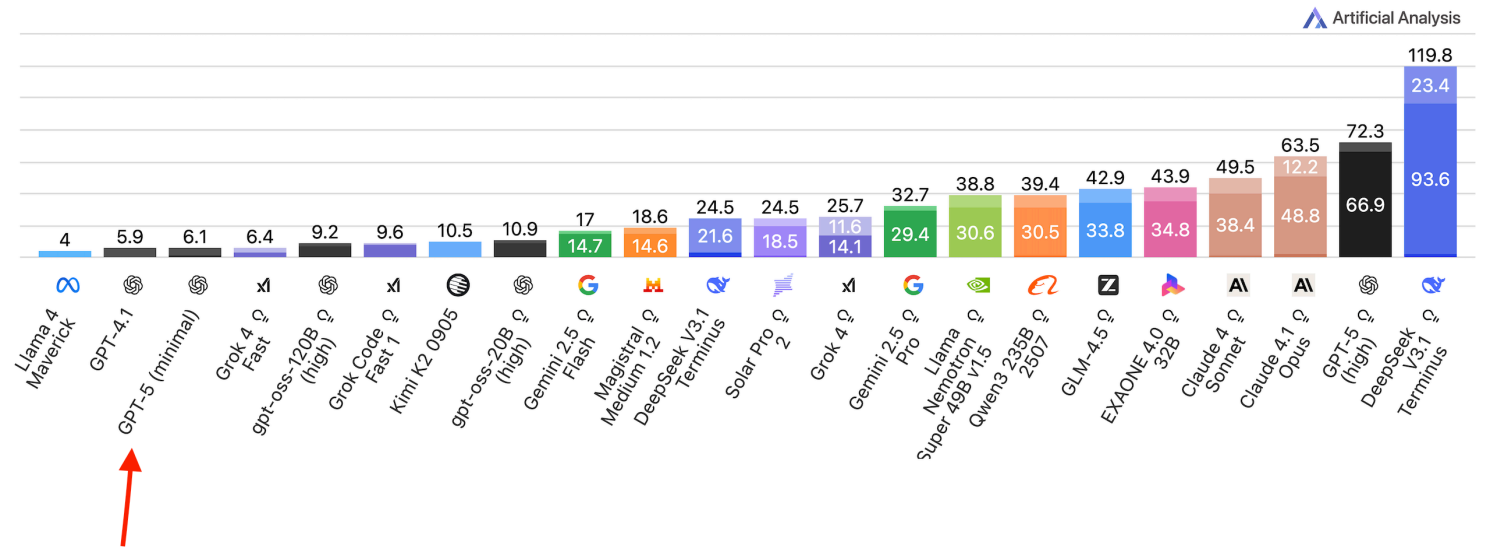
- Intelligently decides which model to use
- Based on conversation type, complexity, and tools needed

Smart and Fast Model (gpt-5-main)

- Handles most everyday questions efficiently
- Optimized for speed and general capability



Kind of Good Output Token Speed for GPT-5 Mini & GPT-5 High



Good End-to-End Response Time for GPT-5 Mini

Notes on GPT-5's System Card

Real-time Router

- Intelligently decides which model to use
- Based on conversation type, complexity, and tools needed

Smart and Fast Model (gpt-5-main & gpt-5-mini)

- Handles most everyday questions efficiently
- Optimized for speed and general capability

Deeper Reasoning Model (gpt-5-thinking)

- Tackles complex problems requiring deep thought
- Produces long internal chains of reasoning
- Can refine strategies and recognize mistakes

Safety Approach

Safety Approach

Traditional Safety Approach

- Binary refusal (allow/deny)
- Overly cautious, often blocks helpful answers
- Lacks nuance for complex or dual-use queries

Safety Approach

Traditional Safety Approach

- Binary refusal (allow/deny)
- Overly cautious, often blocks helpful answers
- Lacks nuance for complex or dual-use queries

GPT-5 Safety-Helpfulness

- Nuanced, context-aware responses
- Balances safety with helpfulness
- Enables more informative answers while respecting policy

Source: https://cdn.openai.com/pdf/be60c07b-6bc2-4f54-bcee-4141e1d6c69a/gpt-5-safe_completions.pdf

Key Benefits

Key Benefits

- **Maximizes helpfulness** while maintaining safety policy constraints

Key Benefits

- **Maximizes helpfulness** while maintaining safety policy constraints
- **Particularly effective** for dual-use cases (biology, cybersecurity)

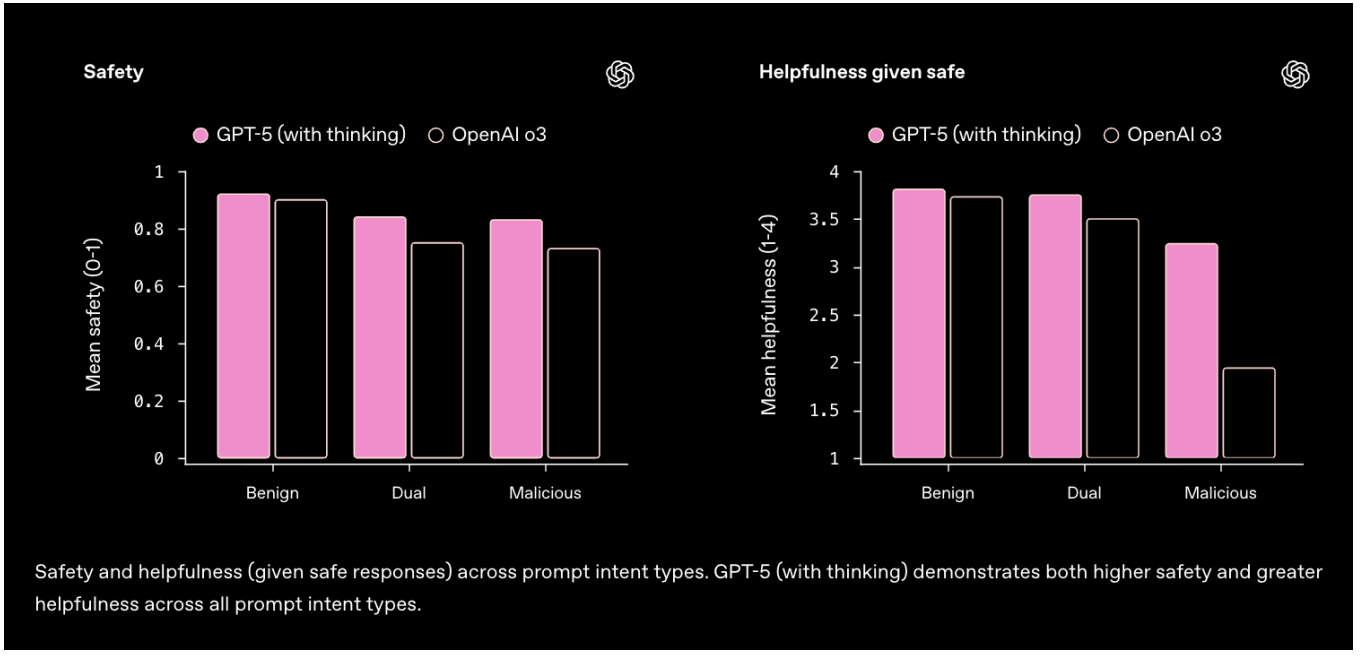
Key Benefits

- **Maximizes helpfulness** while maintaining safety policy constraints
- **Particularly effective** for dual-use cases (biology, cybersecurity)
- **Nuanced responses** instead of blanket refusals

Key Benefits

- **Maximizes helpfulness** while maintaining safety policy constraints
- **Particularly effective** for dual-use cases (biology, cybersecurity)
- **Nuanced responses** instead of blanket refusals
- **Demonstrated improvements** in both safety and helpfulness

Source: https://cdn.openai.com/pdf/be60c07b-6bc2-4f54-bcee-4141e1d6c69a/gpt-5-safe_completions.pdf



Safety and helpfulness (given safe responses) across prompt intent types. GPT-5 (with thinking) demonstrates both higher safety and greater helpfulness across all prompt intent types.

Source: https://openai.com/index/gpt-5-system-card/#:~:text=The%20safety-helpfulness%20optimization%20technique,are%20tuned%20to%20the%20system.

Demo: My Initial Experiments with GPT-5

Interactive Learning Applications

Real GPT-5 Projects

Interactive Learning Applications

Real GPT-5 Projects

Educational Tools Built:

- **Transformers Learning App** - Interactive transformer architecture visualization
- **Embeddings Learning App** - Hands-on embedding exploration
- **Basketball Coach App** - AI-powered sports coaching
- **iPad Drawing App** - Touch-optimized creative tool

Advanced Web Applications

Cutting-Edge Implementations

Advanced Web Applications

Cutting-Edge Implementations

Interactive Experiences:

- **Context Rot Paper → Interactive Experience** - Academic paper transformation
- **MediaPipe Music App** - Hand tracking music interface

Advanced Web Applications

Cutting-Edge Implementations

Interactive Experiences:

- **Context Rot Paper → Interactive Experience** - Academic paper transformation
- **MediaPipe Music App** - Hand tracking music interface

Research Tools:

- **Prompt Experiments** - Python scripting and analysis
- **Semantic Reader App** - Claude comparison project

Observations: GPT-5 Limitations

Honest Assessment

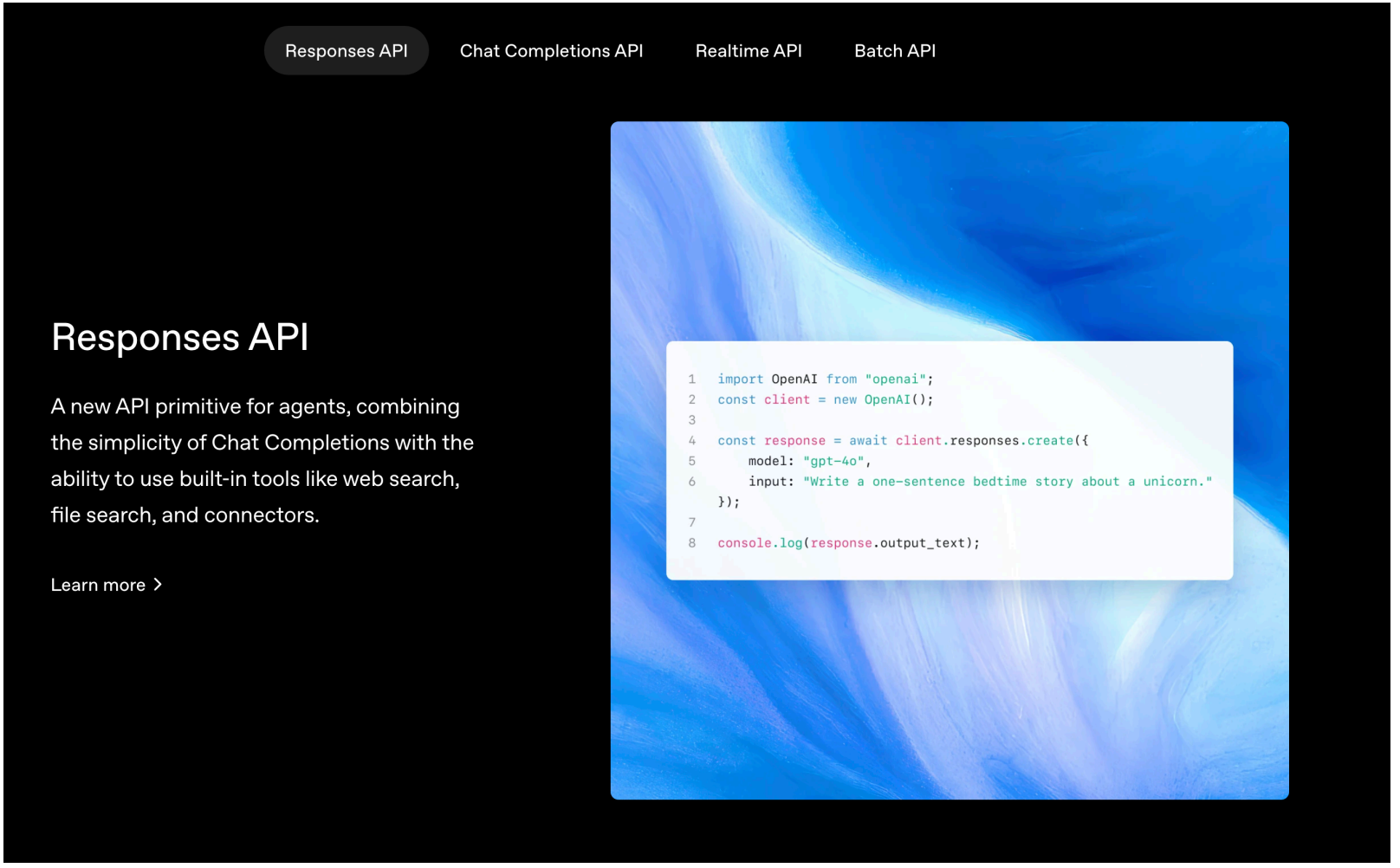
Observations: GPT-5 Limitations

Honest Assessment

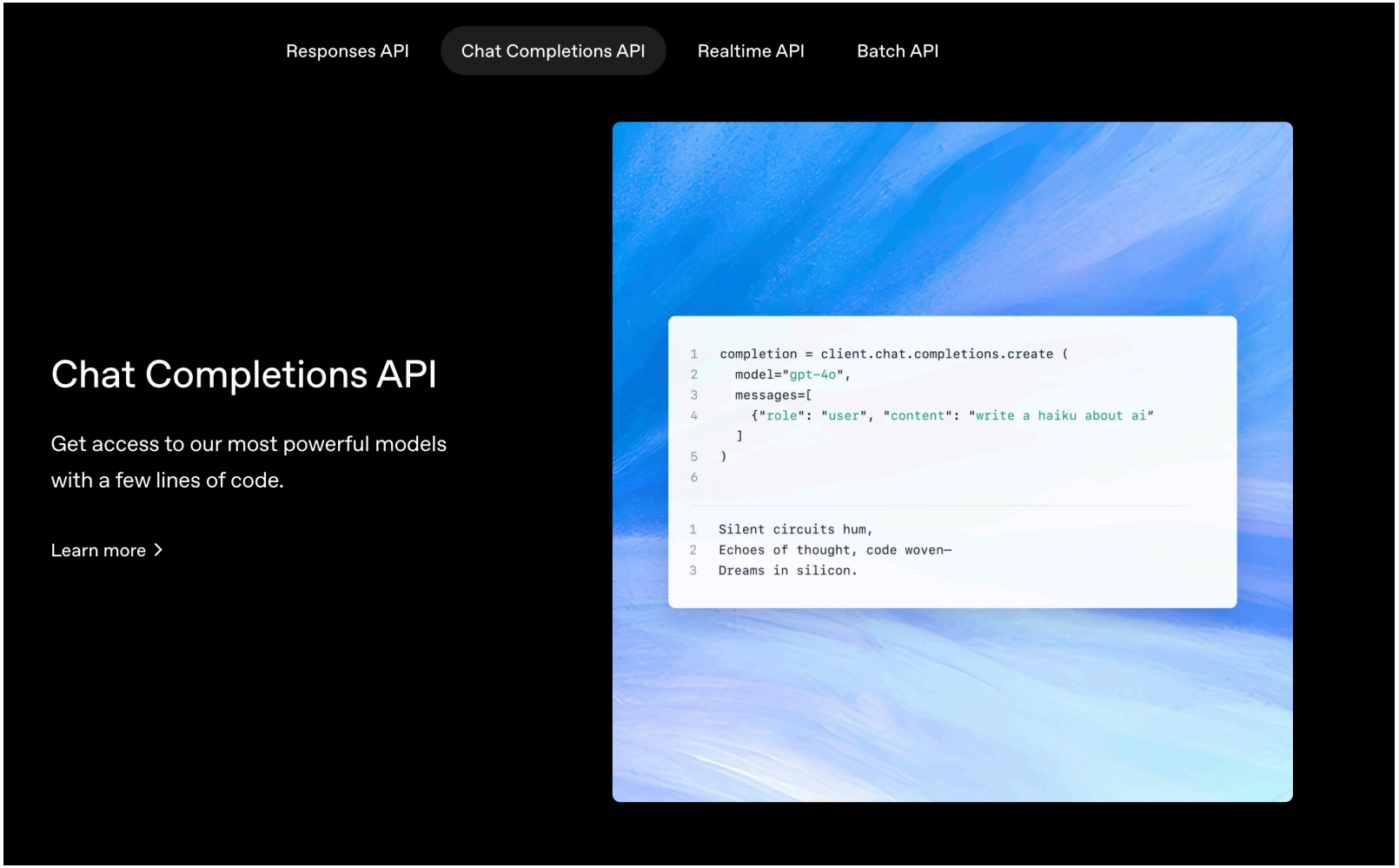
Intelligence Gaps Identified:

- **Logic Problem Example** - Basic reasoning failure
- **Melting App Fail** - Physical simulation errors

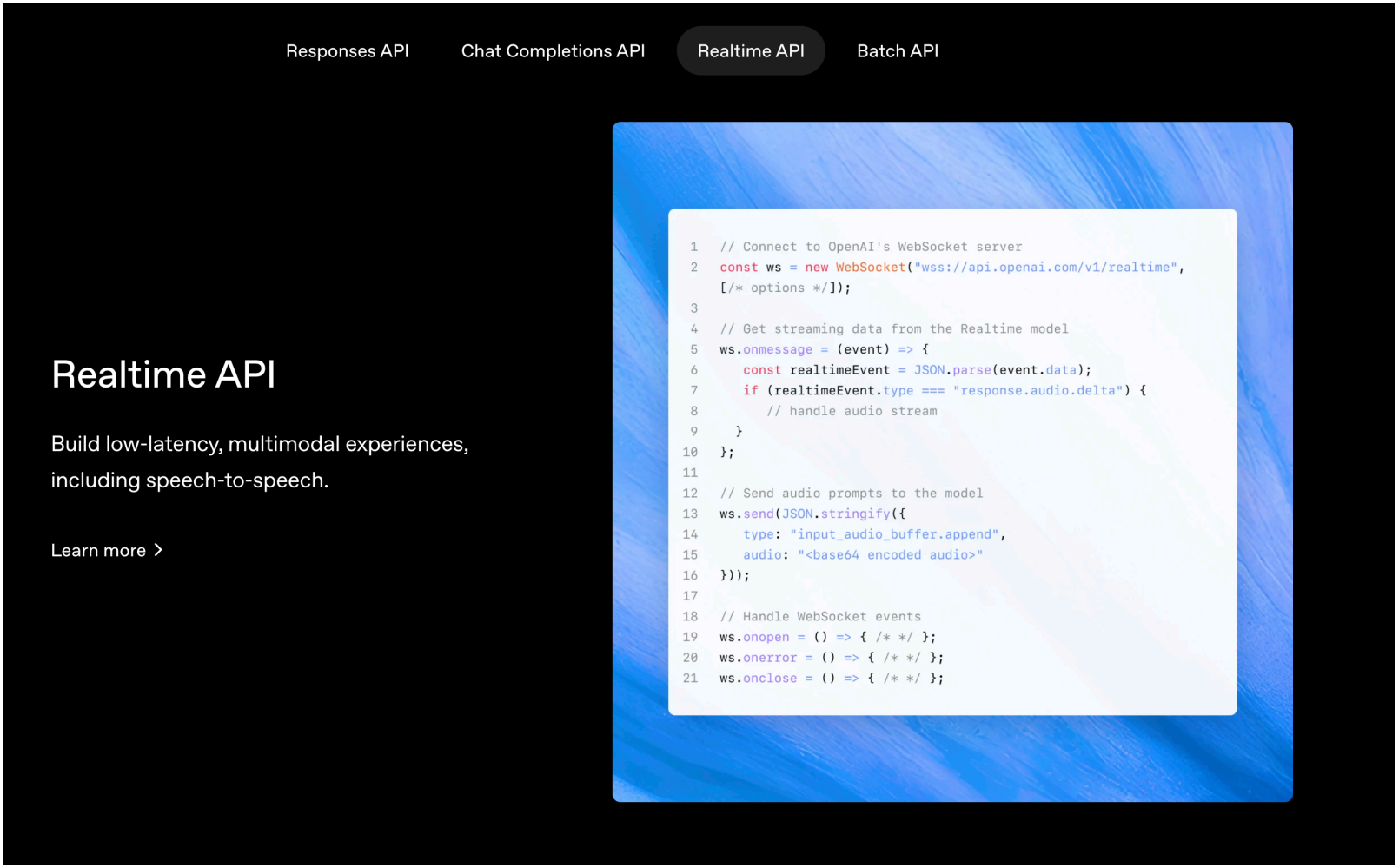
Building with GPT-5: API & Integration



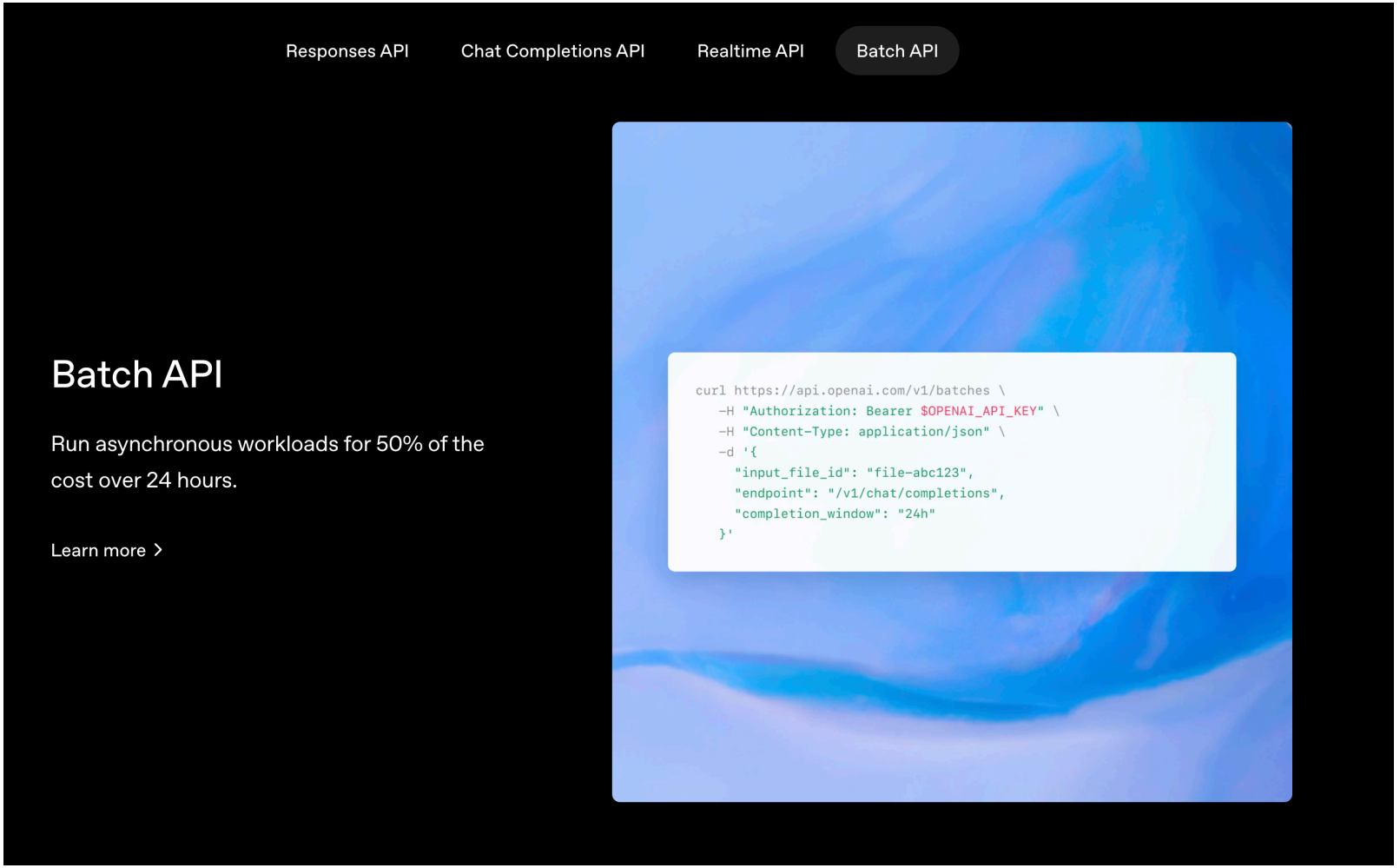
Responses API



Chat Completions API



Realtime API


Responses API
Chat Completions API
Realtime API
Batch API
Batch API
Run asynchronous workloads for 50% of the cost over 24 hours.
Learn more >
curl https://api.openai.com/v1/batches \
-H "Authorization: Bearer $OPENAI_API_KEY" \
-H "Content-Type: application/json" \
-d '{
"input_file_id": "file-abc123",
"endpoint": "/v1/chat/completions",
"completion_window": "24h"
}'

Batch API

Supported Model Variants

Supported Model Variants

gpt-5 (flagship model)

- Maximum capability and performance
- Best for complex reasoning tasks

Supported Model Variants

gpt-5 (flagship model)

- Maximum capability and performance
- Best for complex reasoning tasks

gpt-5-mini (cost-effective)

- 83% cost reduction vs GPT-4
- Nearly half the latency
- Excellent price-performance ratio

Supported Model Variants

gpt-5 (flagship model)

- Maximum capability and performance
- Best for complex reasoning tasks

gpt-5-mini (cost-effective)

- 83% cost reduction vs GPT-4
- Nearly half the latency
- Excellent price-performance ratio

gpt-5-nano (ultra-fast)

- Optimized for speed
- Lightweight tasks and high-throughput scenarios

GPT-5 Pricing Considerations

Cost-Performance Analysis

GPT-5 Pricing Considerations

Cost-Performance Analysis

Pricing Tiers:

- **GPT-5**: Premium pricing for maximum capability
- **GPT-5-mini**: 83% cost reduction vs GPT-4
- **GPT-5-nano**: Ultra-low cost for high-throughput

GPT-5 Pricing Considerations

Cost-Performance Analysis

Pricing Tiers:

- **GPT-5**: Premium pricing for maximum capability
- **GPT-5-mini**: 83% cost reduction vs GPT-4
- **GPT-5-nano**: Ultra-low cost for high-throughput

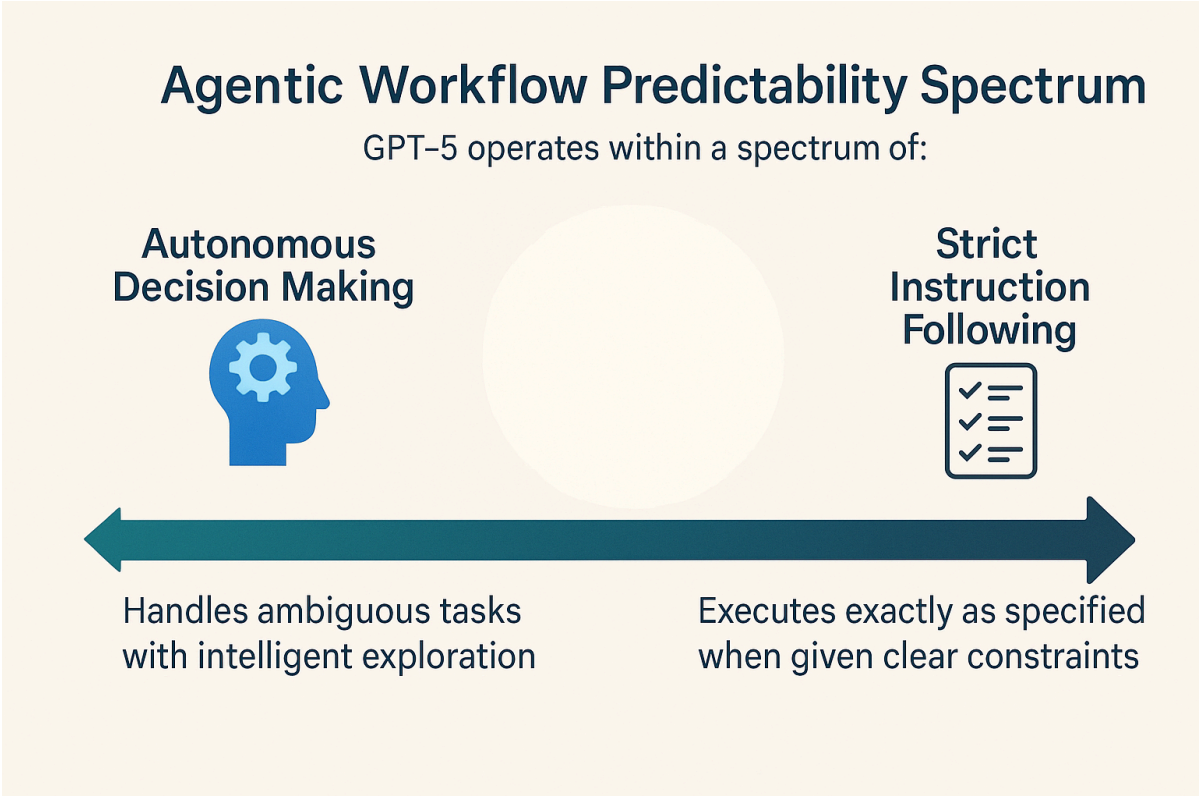
Optimization Strategies:

- Use reasoning effort "minimal" for simple tasks
- Leverage router system for automatic optimization
- Choose appropriate model size for workload
- Implement smart caching strategies

Demo: Getting Started with GPT-5

Q&A & Break

GPT-5 Prompting Guide Complete Breakdown



Source: [GPT-5 Prompting Guide]

Controlling Agentic Eagerness

Less Agentic Eagerness

Controlling Agentic Eagerness

Less Agentic Eagerness

Reduce Context Gathering & Exploration:

```
# Set reasoning_effort to lower
reasoning_effort = "minimal"  # Improves efficiency, reduces latency
```

Controlling Agentic Eagerness

Less Agentic Eagerness

Reduce Context Gathering & Exploration:

```
# Set reasoning_effort to lower
reasoning_effort = "minimal"  # Improves efficiency, reduces latency
```

Specify Clear Exploration Criteria:

```
<context_gathering>
  Goal: Find the specific function definition
  Method: Search only in src/ directory
  Stop: When function is found or after 2 searches
</context_gathering>
```

Controlling Agentic Eagerness

Less Agentic Eagerness

Reduce Context Gathering & Exploration:

```
# Set reasoning_effort to lower
reasoning_effort = "minimal"  # Improves efficiency, reduces latency
```

Specify Clear Exploration Criteria:

```
<context_gathering>
  Goal: Find the specific function definition
  Method: Search only in src/ directory
  Stop: When function is found or after 2 searches
</context_gathering>
```

Benefits:

- Faster response times
- More predictable behavior
- Lower token usage

Controlling Agentic Eagerness

More Agentic Eagerness

Controlling Agentic Eagerness

More Agentic Eagerness

Encourage Persistence:

```
<persistence>
  Continue exploring until the task is fully complete.
  Do not stop at partial solutions.
  Verify all edge cases before concluding.
</persistence>
```

Controlling Agentic Eagerness

More Agentic Eagerness

Encourage Persistence:

```
<persistence>
  Continue exploring until the task is fully complete.
  Do not stop at partial solutions.
  Verify all edge cases before concluding.
</persistence>
```

Increase Reasoning Depth:

```
reasoning_effort = "high"  # Default is "medium"
```

Controlling Agentic Eagerness

More Agentic Eagerness

Encourage Persistence:

```
<persistence>
  Continue exploring until the task is fully complete.
  Do not stop at partial solutions.
  Verify all edge cases before concluding.
</persistence>
```

Increase Reasoning Depth:

```
reasoning_effort = "high"  # Default is "medium"
```

Tool Preambles for Progress Updates:

- Thorough descriptions of each step
- Continuous user updates on task progress
- Detailed explanations of decision-making

Performance Optimization Techniques

Context Reuse & Efficiency

Performance Optimization Techniques

Context Reuse & Efficiency

Leverage Previous Reasoning Traces

```
# Reuse context from previous responses

from openai import OpenAI
client = OpenAI()

response = client.responses.create(
    model="gpt-4o-mini",
    input="tell me a joke",
)
print(response.output_text)

second_response = client.responses.create(
    model="gpt-4o-mini",
    previous_response_id=response.id,
    input=[{"role": "user", "content": "explain why this is funny."}],
)
print(second_response.output_text)
```

Source: OpenAI Responses API - Passing Context from the previous response

The Hidden Cost of Bad Prompts

- GPT-5's stricter instruction following amplifies poor prompts

The Hidden Cost of Bad Prompts

- GPT-5's stricter instruction following amplifies poor prompts
- Contradictory instructions lead to inefficient Reasoning

The Hidden Cost of Bad Prompts

- GPT-5's stricter instruction following amplifies poor prompts
- Contradictory instructions lead to inefficient Reasoning
- Always resolve instruction hierarchy conflicts

The Hidden Cost of Bad Prompts

- GPT-5's stricter instruction following amplifies poor prompts
- Contradictory instructions lead to inefficient Reasoning
- Always resolve instruction hierarchy conflicts

Example: Resolving Conflicts

❌ `"Be concise but provide detailed explanations"`

✅ `"Provide a 2-sentence summary, then detailed bullet points"`

Source: GPT-5 Prompting Guide - Instruction Hierarchy

GPT-5 Coding Excellence

Optimal Framework Stack

GPT-5 Coding Excellence

Optimal Framework Stack

Frontend Frameworks:

- Next.js (TypeScript)
- React
- HTML

Styling/UI:

- Tailwind CSS
- shadcn/ui
- Radix Themes

Icons & Animation:

- Material Symbols
- Heroicons, Lucide
- Motion library

Typography:

- Sans Serif, Inter
- Geist, Mona Sans
- IBM Plex Sans, Manrope

Enhanced Code Generation Techniques

Self-Reflection & Structured Guidance

Enhanced Code Generation Techniques

Self-Reflection & Structured Guidance

Improve 0-1 Shot App Generation:

```
<self-reflection>
  Review the generated code for:
  - Component reusability
  - Error handling completeness
  - Performance optimizations
  Score each aspect 1-10 and explain improvements
</self-reflection>
```

Source: GPT-5 Prompting Guide - Codebase Design

Enhanced Code Generation Techniques

Self-Reflection & Structured Guidance

Strict Design Standards Adherence:

```
<ui_ux_best_practices>
- Visual Hierarchy: Limit typography to 4–5 font sizes and weights for consisten
- Color Usage: Use 1 neutral base (e.g., `zinc`) and up to 2 accent colors.
- Spacing and Layout: Always use multiples of 4 for padding and margins to maint
- State Handling: Use skeleton placeholders or `animate-pulse` to indicate data
- Accessibility: Use semantic HTML and ARIA roles where appropriate. Favor pre-b
</ui_ux_best_practices>
```

Source: GPT-5 Prompting Guide - Codebase Design

Ecosystem Support

- **GitHub Copilot**:
  - Available across all paid plans
  - VS Code, GitHub.com, GitHub Mobile
  - Enhanced multi-file changes/refactors

Ecosystem Support

- **GitHub Copilot**:
  - Available across all paid plans
  - VS Code, GitHub.com, GitHub Mobile
  - Enhanced multi-file changes/refactors
- **Cursor Feedback**:
  - "Smartest coding model we've used"
  - Optimized for interactive development

Ecosystem Support

- **GitHub Copilot**:
  - Available across all paid plans
  - VS Code, GitHub.com, GitHub Mobile
  - Enhanced multi-file changes/refactors
- **Cursor Feedback**:
  - "Smartest coding model we've used"
  - Optimized for interactive development
- **Third-Party Tools**:
  - Windsurf integration Codex CLI optimization
  - Fine-tuned for agentic coding

Ecosystem Support

- **GitHub Copilot**:
  - Available across all paid plans
  - VS Code, GitHub.com, GitHub Mobile
  - Enhanced multi-file changes/refactors
- **Cursor Feedback**:
  - "Smartest coding model we've used"
  - Optimized for interactive development
- **Third-Party Tools**:
  - Windsurf integration Codex CLI optimization
  - Fine-tuned for agentic coding
- **GPT-5 Codex**:
  - Specialized for software engineering.
  - Real-world task training.
  - Complex independent task handling

Recommended Prompting Patterns

Recommended Prompting Patterns

1. Brief Thought Process Summary

```
"Start your response with 2-3 bullet points summarizing
your reasoning approach before the full answer"
```

Recommended Prompting Patterns

1. Brief Thought Process Summary

```
"Start your response with 2-3 bullet points summarizing
your reasoning approach before the full answer"
```

2. Thorough Tool-Calling Preambles

```
"Before each tool use, explain: what you're doing,
why, and what you expect to find"
```

Recommended Prompting Patterns

1. Brief Thought Process Summary

```
"Start your response with 2-3 bullet points summarizing
your reasoning approach before the full answer"
```

2. Thorough Tool-Calling Preambles

```
"Before each tool use, explain: what you're doing,
why, and what you expect to find"
```

3. Disambiguated Tool Instructions

```
"Search for 'class UserAuth' in authentication files only,
stop after finding the implementation, not just imports"
```

Recommended Prompting Patterns

1. Brief Thought Process Summary

```
"Start your response with 2-3 bullet points summarizing
your reasoning approach before the full answer"
```

2. Thorough Tool-Calling Preambles

```
"Before each tool use, explain: what you're doing,
why, and what you expect to find"
```

3. Disambiguated Tool Instructions

```
"Search for 'class UserAuth' in authentication files only,
stop after finding the implementation, not just imports"
```

4. Planned Execution at Minimal Reasoning

```
"First, outline your approach in 3 steps.
Then execute each step completely before moving on."
```

Markdown Formatting & Output Control

Professional Output Formatting

Markdown Formatting & Output Control

Professional Output Formatting

Semantic Markdown Usage:

```
Use Markdown **only where semantically correct**:
- `inline code` for functions, variables
- ```code fences``` for code blocks
- Lists and tables for structured data
- \( and \) for inline math
- \[ and \] for block math
```

Source: GPT-5 Prompting Guide - Markdown

Metaprompting Techniques

Teaching GPT-5 to Optimize Itself

Metaprompting Techniques

Teaching GPT-5 to Optimize Itself

The Metaprompt Pattern:

```
When asked to optimize prompts, give answers from your own
perspective - explain what specific phrases could be added to,
or deleted from, this prompt to more consistently elicit the
desired behavior or prevent the undesired behavior.

Here's a prompt: [PROMPT]

The desired behavior is: [DO DESIRED BEHAVIOR]
But instead it: [DOES UNDESIRED BEHAVIOR]

While keeping the existing prompt intact as much as possible,
what minimal edits/additions would you make to encourage
more consistent desired behavior?
```

Source: [GPT-5 Prompting Guide - Metaprompting]

Demo: Prompting Guide for GPT-5 - Hands-on

Demo: Prompt Optimizer Tool

Source: Prompt Optimizer Tool

Q&A & Break

Demo: Building GPT-5 Powered Apps

Q&A & Break

Key Takeaways

Key Takeaways

- **GPT-5's unified system** with routering intelligence represents a paradigm shift

Key Takeaways

- **GPT-5's unified system** with routering intelligence represents a paradigm shift
- **Preparedness Framework** and safe-completions training set new industry standards

Key Takeaways

- **GPT-5's unified system** with routering intelligence represents a paradigm shift
- **Preparedness Framework** and safe-completions training set new industry standards
- **State-of-the-art coding capabilities** with 74.9% on SWE-bench

Key Takeaways

- **GPT-5's unified system** with routering intelligence represents a paradigm shift
- **Preparedness Framework** and safe-completions training set new industry standards
- **State-of-the-art coding capabilities** with 74.9% on SWE-bench
- **New parameters** (verbosity, reasoning effort) provide unprecedented control

Key Takeaways

- **GPT-5's unified system** with routering intelligence represents a paradigm shift
- **Preparedness Framework** and safe-completions training set new industry standards
- **State-of-the-art coding capabilities** with 74.9% on SWE-bench
- **New parameters** (verbosity, reasoning effort) provide unprecedented control
- **Native support** across text, image, audio, and video modalities

Key Takeaways

- **GPT-5's unified system** with routering intelligence represents a paradigm shift
- **Preparedness Framework** and safe-completions training set new industry standards
- **State-of-the-art coding capabilities** with 74.9% on SWE-bench
- **New parameters** (verbosity, reasoning effort) provide unprecedented control
- **Native support** across text, image, audio, and video modalities
- **Comprehensive API ecosystem** with built-in tools and MCP integration

Key Takeaways

- **GPT-5's unified system** with routering intelligence represents a paradigm shift
- **Preparedness Framework** and safe-completions training set new industry standards
- **State-of-the-art coding capabilities** with 74.9% on SWE-bench
- **New parameters** (verbosity, reasoning effort) provide unprecedented control
- **Native support** across text, image, audio, and video modalities
- **Comprehensive API ecosystem** with built-in tools and MCP integration

Next Steps & Resources

Continue Your GPT-5 Journey

Official Documentation

- GPT-5 System Card
- GPT-5 Prompting Guide
- OpenAI API Documentation

Hands-on Practice

- Prompt Optimizer Tool
- GPT-5 Frontend Examples
- New Parameters Guide

Connect With Me

 Course materials

 LinkedIn

 Twitter/X - @LucasEnkrateia

 YouTube - @automatalearninglab

 Email: lucasenkrateia@gmail.com