Claude Certified Developer – Professional

Exam Guide

Version 1.0 · Effective November 2026 · Exam code: CCDV-P · This guide is subject to change without notice.

1. About This Certification

The Claude Certified Developer – Professional certification validates that an individual can operate Claude-powered applications, agents, and workflows in production at scale. Building on the scope of the Claude Certified Developer – Foundations certification, it is intended for experienced technical professionals who are accountable for the reliability, quality, cost, and safety of live Claude systems serving real users, and who increasingly direct fleets of coding agents that carry out engineering work under their supervision.

This guide is the authoritative reference for candidates preparing to sit the exam. It describes the credential's purpose, the intended audience, the exam format and content blueprint, scoring, policies, and the rules of conduct that all candidates agree to. Read it in full before scheduling your exam.

2. Purpose and Value of the Credential

The primary purpose of the Claude Certified Developer – Professional certification is to provide an independent assessment of the advanced knowledge, skills, and abilities required to run Claude-based systems in production competently. Earning the credential signals to employers, clients, and teams that the holder can own the operation of deployed Claude applications and agents: instrumenting them, diagnosing complex failures, gating releases with evaluation, controlling cost and latency at fleet scale, and hardening tool-using agents against adversarial input.

Candidates who earn the credential can demonstrate that they are able to effectively apply knowledge in the following areas:

- Instrumenting Claude systems with production-grade observability (tracing, metrics, logging) and monitoring quality, cost, and drift
- Diagnosing complex failure modes across multi-turn and multi-agent systems, and leading incident response
- Engineering evaluation systems, including calibrated LLM-as-judge pipelines, regression suites, and CI/CD evaluation gates

- Designing resilient, high-throughput integration layers with progressive delivery of prompt and model changes
- Designing context, memory, and retrieval architectures that keep long-running and knowledge-grounded agents accurate and affordable
- Optimizing cost and latency at fleet scale through model routing, caching strategy, and latency engineering
- Implementing layered runtime guardrails and defense-in-depth security for tool-using agents
- Orchestrating durable multi-agent systems with checkpointing, session persistence, and failure containment
- Operating fleets of coding agents for software delivery, including verification, AI code review, and unattended runs

3. Intended Audience

The certification is intended for technical professionals who own Claude-powered systems in production. This audience primarily comprises senior and staff software engineers, platform and site reliability engineers extending their practice to LLM systems, and solutions engineers responsible for the reliability, quality, cost, and safety of deployed Claude applications.

Professional candidates go beyond building and shipping: they operate. They are accountable for uptime and quality service level objectives (SLOs), respond to incidents, control spend across applications and tenants, and harden agentic systems against adversarial input. Increasingly, they also direct fleets of coding agents that carry out engineering work under their supervision. They typically work in environments such as healthcare, financial services, government, and enterprise SaaS, where production reliability and security expectations are high.

The intended candidate works daily with the Claude APIs and SDKs, including the Messages API, Agent SDK, and Claude Managed Agents, and with Claude Code and MCP, and understands how first-party API deployments differ from cloud-platform deployments. They are at home in observability stacks, CI/CD pipelines with automated evaluation gates, and multi-tenant enterprise environments with quota, isolation, and cost-allocation requirements, and they have built and deployed at least one tool-using agent in production rather than only an LLM-backed chatbot or summarizer.

4. Exam Details at a Glance

Credential	Claude Certified Developer – Professional
Exam code	CCDV-P
Number of items	70, of which 60 are scored and 10 are unscored items. Unscored items are not identified and do not affect your score.
Item format	Scenario-based multiple-choice and multiple-response items; each item states how many responses to select
Time limit	120 minutes
Delivery	Proctored: online proctored and/or test center, per program policy
Prerequisites	None. The Claude Certified Developer – Foundations certification is recommended, but not required, preparation. The credential is awarded based on exam performance alone.
Passing score	Scaled score of 720 on a scale of 100–1,000
Exam fee	$175 USD
Validity period	12 months from the date the credential is awarded
Result reporting	Pass/fail with scaled score (100–1,000), plus percent-correct by domain on the score report

5. Exam Content Outline (Blueprint)

The exam blueprint defines the content domains and skills measured and the approximate weight of each domain on the exam. Weights reflect the relative importance of each domain to competent performance as determined through the job task analysis and content validation surveys. The percentages indicate the approximate proportion of scored items drawn from each domain.

Domain	Content Domain	Weight
1	Advanced Agent Architecture and Orchestration	17%
2	Production Integration, Reliability, and Knowledge Grounding	20%
3	Observability and Production Monitoring	12%
4	Advanced Debugging and Incident Response	10%
5	Evaluation Engineering and Quality Assurance	12%
6	Cost and Latency Optimization at Scale	8%
7	Runtime Guardrails and Production Security	11%
8	Agent Fleet Operations for Software Delivery	10%
Total		100%

Detailed objectives by domain

Each domain below lists the skills against which exam items are written. Skill descriptions summarize the knowledge and competencies measured.

Domain 1: Advanced Agent Architecture and Orchestration (17%)

Multi-Agent Orchestration

Evaluating orchestration patterns for production multi-agent designs and inter-agent failures, including supervisor/worker hierarchies, parallel subagent dispatch, delegation and context-isolation boundaries, inter-agent communication, and human-in-the-loop and approval flows.

Durable Execution and State Management

Applying session persistence, checkpointing, resumability, and memory architecture patterns to long-running agents that must survive interruption, restart, and model version change.

Architecture Selection at Scale

Evaluating production cost, reliability, and quality data to decide whether to evolve a running system between workflow and agentic patterns, and selecting the agent harness, framework, and reusable-pattern standards a team should adopt.

Context and Memory Architecture

Evaluating context-policy options for long-running agents approaching context limits or cost ceilings, including context compaction and editing, persistent memory, cache-safe instruction changes mid-conversation, and tool-use techniques that keep large tool catalogs out of the context window.

Domain 2: Production Integration, Reliability, and Knowledge Grounding (20%)

Reliability Engineering for Claude Services

Evaluating and applying resilience patterns for production Claude integrations, including retry and backoff design, circuit breakers, graceful degradation, and cross-model, cross-region, and cross-platform fallbacks that distinguish rate limits from safety refusals from context-window stops while respecting per-platform feature parity.

Scale and Throughput Management

Analyzing rate-limit strategy, queueing and backpressure, batch pipeline design, multi-tenant quota and isolation, and capacity planning for high-volume Claude workloads, and identifying whether rate limits, token throughput, or batch economics is the binding constraint.

Production MCP and Change Management

Applying production MCP operation and change-management practice to MCP server deployments, model version migrations, and prompt updates, including remote MCP deployment and authentication, tool and schema versioning, migration testing against breaking behavior changes, platform-specific configuration parity and gateway/routing architectures, prompt versioning, and progressive rollout with rollback.

Retrieval and Knowledge Grounding at Scale

Evaluating retrieval architecture for knowledge-grounded production applications, including agentic search versus RAG tradeoffs, chunking and embedding strategy, hybrid search, and reranking, and designing grounding practices such as source citation, uncertainty declaration, and confidence signals, and diagnosing retrieval-quality failures by stage.

Domain 3: Observability and Production Monitoring (12%)

Telemetry and Instrumentation

Applying instrumentation practice to production Claude systems under debuggability and privacy requirements, including distributed traces for agent loops and tool calls, token and cost metrics, cache-performance diagnostics, grounding and reasoning traceability, structured log events, and content-capture configuration under privacy constraints such as zero-data-retention.

Production Monitoring and Alerting

Analyzing production signals from LLM-backed services to define service level indicators (SLIs) and SLOs, design alerting and dashboards (including token and cost telemetry), and detect quality drift, behavioral regressions, and anomalous usage, including drift after model updates.

Trace Analysis at Scale

Analyzing high-volume trace data using sampling strategies, aggregation, and failure-cohort analysis to surface and quantify systemic issues that are not visible in individual traces.

Domain 4: Advanced Debugging and Incident Response (10%)

Complex Failure Diagnosis

Analyzing evidence from complex production failures, including multi-turn, multi-agent, and non-deterministic failures, to isolate the problem origin among model output, agent harness, tools, and data, including model refusals as a failure class and retrieval-grounding failures.

Incident Response and Learning Loops

Evaluating incident-response options during an LLM-system outage or quality regression, including triage, mitigation, rollback, and communication, and converting the incident into regression evaluations, monitoring coverage, and postmortem findings.

Domain 5: Evaluation Engineering and Quality Assurance (12%)

Evaluation System Design

Evaluating and designing offline evaluation systems for Claude applications, including golden sets, regression suites, behavioral checks, CI/CD evaluation gates, retrieval precision and groundedness metrics, and release criteria for prompt, model, and agent changes, and maintaining suite validity over time against contamination, saturation, and coverage gaps.

LLM-as-Judge Engineering

Analyzing LLM-as-judge evaluator quality against human-labeled data, including calibration, rubric design, bias and position effects, inter-rater reliability, and judge drift across model versions.

Online Quality Measurement

Analyzing quality on live production traffic using online evaluation techniques, including live-traffic sampling, A/B testing, feedback loops, and metric selection matched to task type (extraction, generation, agentic, retrieval-grounded).

Domain 6: Cost and Latency Optimization at Scale (8%)

Cost Architecture at Scale

Evaluating fleet-level optimization strategies using usage and cost data, including cost modeling and forecasting, model routing and cascades across the Claude model family, caching strategy and cache diagnostics, batch-versus-realtime economics, and per-tenant budget enforcement.

Latency Engineering

Analyzing the latency profile of a Claude application against a latency target, distinguishing time-to-first-token from total latency, and applying streaming, parallel tool execution, cache checkpointing, and effort-level tuning.

Domain 7: Runtime Guardrails and Production Security (11%)

Runtime Guardrail Systems

Evaluating layered guardrail architectures against a production agent's risk profile, including input/output filtering, PII redactors, hook-based policy enforcement, destructive-action prevention, sandboxing that grants tool and web access without exposing production systems, approval gates, refusal handling and safety fallbacks, deterministic enforcement for irreversible actions, and false-positive/false-negative tuning.

Threat Defense and Adversarial Testing for Agentic Systems

Analyzing the attack surface of tool-using agents that process untrusted content, including prompt injection, tool and memory poisoning, data exfiltration paths, unsafe-autonomy conditions, and the high-risk combination of private data access, untrusted input, and external communication, and applying defense-in-depth mitigations, red-team exercise design, and secure MCP configuration, drawing on the OWASP Top 10 for LLM Applications and the NIST AI Risk Management Framework and its Generative AI Profile.

Identity, Secrets, and Audit at Scale

Applying enterprise identity and secrets practice to production Claude deployments, including agent identity, workload identity federation replacing static API keys, credential and secrets management, least-privilege tool permissioning, and audit logging of security-relevant agent actions.

Domain 8: Agent Fleet Operations for Software Delivery (10%)

Parallel Agent Orchestration for Engineering Work

Evaluating parallel fan-out patterns for large engineering tasks, including parallel agents on isolated worktrees, task decomposition, and merge and conflict strategy, and diagnosing fleet-level failure modes from fleet telemetry.

Verification and AI Code Review Operations

Analyzing verification operations in delivery pipelines where generation outpaces review, including AI code review introduced in shadow mode, calibration against human review before merge gating, and human oversight design.

Unattended and Event-Driven Agent Operations

Applying unattended-operation patterns to goal- or schedule-driven engineering tasks, including agents running toward goals, on schedules, or as event-driven responders such as CI/CD incident first response, with checkpointing, notification, and escalation design.

Organization-Wide Coding Agent Configuration

Applying enterprise configuration for organizations adopting coding agents, including managed settings and policies, MCP allowlists, telemetry, gateway routing, and governance with a quality and safety bar for sharing prompts, Skills, and MCP servers across teams.

6. How to Prepare

There is no single required course. Anthropic does not guarantee that any particular resource ensures a passing result. Candidates are encouraged to combine hands-on production experience with the resources below:

- Study the exam blueprint in Section 5 and self-assess against each objective
- Review the Claude Certified Developer – Foundations exam guide; the scope of that certification is assumed knowledge for this exam

- Review official Anthropic documentation and engineering guidance for the Claude API, Agent SDK, Claude Managed Agents, Claude Code (including hooks, Skills, plugins, and headless operation), and MCP, as well as the differences between first-party API and cloud-platform deployments
- Operate at least one tool-using Claude agent in production, with distributed tracing and cost telemetry, SLIs and SLOs, an evaluation suite that gates releases, layered guardrails, and an incident-response and postmortem practice
- Practice the Professional competencies: diagnosing multi-agent and non-deterministic failures from traces, calibrating an LLM-as-judge evaluator against human labels, designing fallbacks and backpressure for Claude workloads, choosing context and memory policies for long-running agents, modeling fleet-level cost and latency, red-teaming a tool-using agent, and running parallel and unattended coding agents
- Complete the sample questions in Section 7 to familiarize yourself with item style

7. Sample Questions

These illustrative items show the style and cognitive level of the exam. They are not drawn from the live item bank. Correct answers and rationale appear after the questions.

Sample 1 · Domain 2 — Production Integration, Reliability, and Knowledge Grounding

A customer-facing Claude service runs with a fallback policy that retries any failed request against a second model with exponential backoff. After a model update, the team sees an unexpected cost spike and a cluster of fallback responses that are shorter than expected. Traces show three distinct stop conditions mixed in the same retry path: rate-limit errors, responses that stopped at the maximum output length, and responses in which the model declined the request. Which change best improves reliability?

A. Increase the retry count and widen the backoff window so that more requests eventually succeed regardless of why they failed.

B. Branch on the stop condition: back off and retry rate limits, raise the output budget on length stops, and route declined requests to a safety-handling path.

C. Route all traffic to the fallback model permanently, since it is answering the retried requests and the primary model changed with the update.

D. Disable the fallback entirely and return the first error to the caller so that the cost spike stops and failures become visible.

Sample 2 · Domain 5 — Evaluation Engineering and Quality Assurance

A team gates prompt releases with an LLM-as-judge evaluator that compares the new response to the current production response and picks the better one. Over several weeks, every candidate prompt passes, yet user-reported quality has not improved. A sample of judge verdicts shows the judge agrees with human raters on only slightly more than half of the pairs, and it selects whichever response appears first far more often than the humans do. Which action should the team take first?

A. Replace pairwise comparison with absolute scoring on a fixed rubric so the judge never sees two responses side by side.

B. Switch the judge to a larger, more capable model, since a stronger model will agree with human raters more often.

C. Randomize response order, re-measure agreement against the human labels, and withhold gating authority until agreement is acceptable.

D. Lower the pass threshold so the gate becomes more sensitive to the small improvements each candidate prompt introduces.

Sample 3 · Domain 7 — Runtime Guardrails and Production Security

An internal support agent reads customer tickets, has access to a CRM containing account details, and can send email through a tool. A red-team exercise shows that a ticket containing hidden instructions can cause the agent to look up another customer’s record and email it to an external address. The team has already added a system prompt instruction telling the model to ignore instructions found in tickets. Which additional control most directly closes the exploit path?

A. Enforce recipient allowlisting or human approval on the email tool, and scope CRM access to the ticket’s own account.

B. Add a stronger system prompt warning that exfiltrating customer data is prohibited and that ticket content must never be treated as instructions.

C. Reduce the model’s effort level so that it is less likely to follow multi-step instructions embedded in ticket content.

D. Log every email the agent sends, including recipient and attachments, so that any exfiltration can be investigated after the fact.

Answer key and rationale

Sample 1: B. The three stop conditions are different failure classes and require different handling. Retrying a rate-limited request with backoff is correct; retrying a length-stopped response reproduces the truncation and wastes tokens; retrying a declined request repeats the refusal at additional cost and may be inappropriate. A single undifferentiated retry path is the root cause. Increasing retries (A) amplifies the cost spike; permanent rerouting (C) and disabling the fallback (D) trade one failure for another without addressing the classification problem.

Sample 2: C. The evidence describes a position bias and poor agreement with human labels, so the judge's verdicts are not yet trustworthy. The first step is to control the bias (randomized or swapped ordering), re-calibrate against the human-labeled set, and withhold gating authority until agreement is acceptable. Changing the scoring format (A) or the judge model (B) may help later but does not establish calibration; lowering the threshold (D) makes an uncalibrated gate more permissive.

Sample 3: A. The agent combines private data access, untrusted input, and an external communication channel, so prompt-level instructions alone cannot be relied on to prevent exfiltration. A deterministic control enforced outside the model, such as recipient allowlisting, approval gates for external sends, and least-privilege data scoping, closes the demonstrated path regardless of what the injected text says. A stronger prompt (B) is not an enforceable boundary; reducing effort (C) does not remove the capability; logging (D) detects rather than prevents.

8. How the Exam Is Scored

The Claude Certified Developer – Professional exam is a criterion-referenced assessment: each candidate is measured against a fixed performance standard, not against other candidates. You pass by demonstrating the knowledge and skills defined in the blueprint, not by outperforming a percentage of peers.

Passing standard. The passing score is established through a formal standard-setting study in which trained subject matter experts judge the level of performance expected of a minimally qualified candidate. The score is reported on a scaled range of 100–1,000, and the cut score is 720.

Unscored items. Each exam form includes a small number of unscored items that are being evaluated for use on future forms. They are not identified, are mixed with scored items, and do not count toward your score. Allow time for every item.

Result reporting. Your result is reported as a pass or fail status with a scaled score from 100 to 1,000. Your score report also shows the percentage of items you answered correctly within each content domain. Section-level percentages are provided to help you understand your performance and are not used to determine your pass or fail result, which is based on your total scaled score.

9. Registration and Scheduling

Registration and scheduling are handled through the Anthropic Partner Academy and Pearson:

1. Go to your certification page on the Anthropic Partner Academy and review the exam details.
2. Download the Exam Guide and review the Certification Terms and Conditions and the Certification Exam Policy.
3. Click Register, confirm your exam and exam profile, then select Continue to Pearson.
4. At Pearson, choose online proctoring or a test center, select a date and time, and complete checkout. The price reflects any partner discount.
5. If you have already registered, log in to the Anthropic Partner Academy and select Go to my Pearson account to schedule, take your exam, or view your results and receipts.
6. OnVUE (online proctored) exams can be canceled or rescheduled up to one hour before your appointment, and test center exams up to 48 hours before. Later changes forfeit the exam fee. Rescheduling keeps your price, including any discount. Canceling starts an automatic refund of anything you paid at Pearson's checkout, which takes 7 to 10 business days. If you paid through the Anthropic Partner Academy before 8 a.m. PT on November 12, 2026, Anthropic refunds your payment after you cancel, which can take a few extra days. If your refund has not arrived after 10 business days, email certifications-support@anthropic.com.

10. Exam Policies

Identification

On exam day you must present a valid, unexpired, government-issued photo identification. The name on your ID must match the name on your registration exactly. If you need to correct the name on your registration, you can update this information in your exam profile within your Partner Academy profile before scheduling your exam. For OnVUE (online proctored) exams, updates sync in near real time. For test center appointments, updates sync only once per day — update your name at least 24 hours in advance, or the update may not be reflected when you check in. **NOTE**: You can update your name yourself once; it locks when you confirm your profile. For any later change, contact certifications-support@anthropic.com.

Accommodations

Reasonable accommodations are available for candidates with documented disabilities or needs, in accordance with applicable law. Accommodations must be requested and approved by Pearson before you schedule your exam. Do not schedule your appointment until your request has been approved. Request accommodations at pearsonvue.com/us/en/test-takers/accommodations.

Retake policy

Candidates who do not pass may retake the exam after a required waiting period. Waiting periods increase with each failed attempt: 14 days after the first, 30 days after the second, and 90 days after the third. You may take an exam up to four times within a rolling twelve-month period. Limits apply per exam, so not passing one exam does not prevent you from registering for a different one. The exam fee applies to each attempt.

No-show and late arrival

Candidates who fail to appear for a scheduled exam, or who arrive after the permitted late-arrival window, forfeit the exam fee and must re-register. Cancellation and rescheduling deadlines are described in Section 9.

11. Exam-Day Experience and Rules of Conduct

The exam is administered in a standardized, secure, proctored environment. Whether you test online or at a Pearson test center, the following rules apply to protect the integrity of the credential for everyone who holds it.

During the exam you must:

- Remain within view of the proctor and webcam for the entire session, if testing online
- Keep your workspace clear of notes, books, phones, secondary monitors, and other materials
- Refrain from communicating with any other person during the exam
- Not capture, copy, photograph, or reproduce any exam content in any form

Prohibited items include mobile phones, smart watches, smart glasses, headphones, study materials, and any recording device. Permitted items, if any, such as scratch paper provided by the proctor, are specified by Pearson.

Consequences of misconduct. Cheating, attempting to access prohibited resources, or disclosing exam content may result in invalidation of your result, revocation of your credential(s), and/or a ban from future exams.

12. Confidentiality and Non-Disclosure Agreement

Before the exam begins you must accept a confidentiality and non-disclosure agreement. By accepting, you agree that all exam content, including questions, answer options, and scenarios, is the confidential and proprietary property of Anthropic, and that you will not disclose, reproduce, or distribute any portion of it. If you do not accept the agreement, the exam session ends and no refund is issued.

13. Credential Maintenance and Recertification

The Claude Certified Developer – Professional credential is valid for 12 months from the date it is awarded. Because the underlying technology evolves rapidly, the credential is time-limited so that holders maintain current knowledge.

To renew on time, you review what has changed since you certified and complete a free, non-proctored assessment on the Anthropic Partner Academy. There is no fee for on-time renewal. If your credential lapses, you must retake the full exam at the full fee to regain certified status.

If exam content changes significantly, Anthropic may require holders to retake the full exam to recertify rather than complete the renewal assessment.

Holders remain subject to the rules of conduct described in Section 11.

14. Candidate Support, Appeals, and Privacy

Support

To correct the name on your registration, update your exam profile in your Partner Academy profile before you confirm it. For trouble signing in or reaching Pearson, an exam missing from your Pearson account, a missing partner discount, or profile details you can't change, email certifications-support@anthropic.com. For exam-day problems, other checkout issues, scheduling, rescheduling, canceling, accommodations, and results, contact Pearson support at pearsonvue.com/us/en/anthropic.html.

Appeals and complaints

You may appeal a decision within 14 days of the date you are notified of it, or, for a concern about your exam result, within 14 days of your exam date. Submit your appeal to Pearson support at the link above. Appeals are reviewed under the program's appeals policy. The standard-setting outcome and the content of individual exam items are not subject to appeal.

Privacy

Personal data collected during registration and testing is handled in accordance with Anthropic's privacy policy, Pearson's privacy policy, and applicable data-protection law.

15. Document Control

Version	Summary of change	Date
1.0	Initial publication	November 2026