FREE 90-DAY ROADMAP · 2026 EDITION

# The 90-Day “Become an AI Agent Developer” Roadmap

A free week-by-week plan covering exactly what to learn, build and certify — for total beginners and switchers. No fluff: just the order that works.

### What’s inside

- A day-by-day plan for ai agent development from scratch
- The exact order: how to build ai agents → frameworks → deploy
- Weekly build goals that become your portfolio
- A portfolio checklist recruiters look for
- Career, salary and 2026 hiring-trend context
- Where the certification fits each phase

Published by the Global Skill Development Council (GSDC) — a vendor-neutral certification body trusted by 2,50,000+ professionals across 100+ countries. Plan assumes ~1–2 focused hours a day; move faster or slower to fit your schedule.

00 · BEFORE YOU START

# Who this is for & what you need

This roadmap assumes you are starting close to zero. If you can install Python and follow a tutorial, you can finish it.

## You'll need

- ✔ A laptop, Python 3.11+, and a code editor (VS Code)
- ✔ An LLM API key (or a local model) for testing
- ✔ A GitHub account to host your portfolio
- ✔ ~1–2 focused hours a day for 12 weeks

## How to use it

Work through the weeks in order — each one builds on the last. Ship the weekly build before moving on; a finished small project beats a half-finished big one.

01 · THE BIG PICTURE

# The exact order that works

Most beginners stall by jumping to frameworks too early. This sequence avoids that:

### 1 · FOUNDATIONS (WEEKS 1–4)

- Python + LLM basics, then build a first agent by hand
- Understand function calling and the ReAct loop before any framework

### 2 · FRAMEWORKS (WEEKS 5–8)

- LangChain, LangGraph, AutoGen, CrewAI; add RAG & memory
- Build your first capstone

### 3 · DEPLOY & CERTIFY (WEEKS 9–12)

- Test, deploy, polish the portfolio
- Practice exams, then certify

PHASE 1 · WEEKS 1–4

# Foundations: build before you frameworks

The goal of the first month is understanding, not tools. By the end you'll have written an agent loop yourself — so frameworks make sense instead of feeling like magic.

- ✔ **Week 1** — Python refresh + how LLMs and prompting work
- ✔ **Week 2** — Your first agent: function calling + ReAct
- ✔ **Week 3** — Tools and memory
- ✔ **Week 4** — Mini-project: one genuinely useful agent

**End-of-phase build:** a single tool-using agent that solves a real task for you (e.g. summarises your inbox or answers from your notes).

PHASE 1 · WEEK 1

# Python + LLM basics

- ✔ **Days 1–2:** Python essentials — functions, dicts, virtual envs, pip
- ✔ **Days 3–4:** Call an LLM API; understand tokens, temperature, system prompts
- ✔ **Days 5–6:** Prompting patterns: zero/few-shot, structured output (JSON)
- ✔ **Day 7:** Review & commit a small script to GitHub

**Build:** a CLI script that takes a question and returns a structured JSON answer from an LLM.

PHASE 1 · WEEK 2

# Your first agent: function calling + ReAct

- ✔ **Days 1–2:** Define tools; let the LLM choose one via function calling
- ✔ **Days 3–4:** Execute the tool, feed the result back, get a final answer
- ✔ **Days 5–6:** Build the ReAct loop by hand (reason → act → observe)
- ✔ **Day 7:** Add a second tool and handle a multi-step task

**Build:** a no-framework agent that can use a calculator and a web/search tool to answer multi-step questions.

PHASE 1 · WEEK 3

# Tools & memory

- ✔ **Days 1–2:** Design good tools: clear names, typed args, error handling
- ✔ **Days 3–4:** Short-term memory: managing the conversation context window
- ✔ **Days 5–6:** Long-term memory: embeddings + a vector store (start with Chroma)
- ✔ **Day 7:** Combine tools + memory in one agent

**Build:** an agent that remembers past chats by retrieving from a local vector store.

PHASE 1 · WEEK 4

# Mini-project: one useful agent

- ✔ **Days 1–2:** Pick a real problem you have; scope it small
- ✔ **Days 3–5:** Build it end-to-end with tools + memory
- ✔ **Days 6–7:** Write a README; record a 60-second demo; push to GitHub

**Portfolio piece #1:** a documented, demoable single agent. This is the foundation recruiters will look at first.

PHASE 2 · WEEKS 5–8

# Frameworks & multi-agent systems

Now that you understand the loop, frameworks let you move faster and build bigger. You'll learn the major ones and ship a real capstone.

- ✔ **Week 5** — LangChain + LangGraph
- ✔ **Week 6** — AutoGen & CrewAI (multi-agent)
- ✔ **Week 7** — RAG & vector stores at scale
- ✔ **Week 8** — Capstone build #1

**End-of-phase build:** a multi-agent or RAG application built on a real framework.

PHASE 2 · WEEK 5

# LangChain + LangGraph

- ✔ **Days 1–2:** LangChain basics: LCEL chains, tools, prebuilt ReAct agents
- ✔ **Days 3–4:** Rebuild your week-2 agent in LangChain — far less code
- ✔ **Days 5–6:** LangGraph: model an agent as a stateful graph with loops
- ✔ **Day 7:** Add human-in-the-loop to a graph node

**Build:** a LangGraph agent with an explicit plan → act → review cycle.

PHASE 2 · WEEK 6

# AutoGen & CrewAI (multi-agent)

- ✔ **Days 1–2:** AutoGen: conversable agents; assistant + user-proxy with code execution
- ✔ **Days 3–4:** CrewAI: define roles, tasks and a crew
- ✔ **Days 5–6:** Multi-agent patterns: supervisor/worker, sequential pipeline
- ✔ **Day 7:** Compare frameworks; note when to use each

**Build:** a small crew (e.g. researcher + writer) that produces a finished deliverable.

PHASE 2 · WEEK 7

# RAG & vector stores

- ✔ **Days 1–2:** Chunking & embeddings; load a real document set
- ✔ **Days 3–4:** Vector stores: Chroma locally, then Pinecone or Redis at scale
- ✔ **Days 5–6:** Build a retrieve → answer pipeline with citations
- ✔ **Day 7:** Add a small eval set to measure answer quality

**Build:** a RAG agent that answers questions over your own documents, with sources.

PHASE 2 · WEEK 8

# Capstone build #1

- ✔ **Days 1–2:** Choose a capstone: a voice agent or a full-stack chatbot
- ✔ **Days 3–5:** Build the core flow on a framework you now know
- ✔ **Days 6–7:** Document it; add a demo; push to GitHub

**Portfolio piece #2:** a substantial, framework-based application — the centrepiece of your portfolio.

PHASE 3 · WEEKS 9–12

# Deploy, polish & certify

The final month is what separates hobby projects from hireable work: reliability, deployment and proof.

- ✔ **Week 9** — Testing, debugging & observability
- ✔ **Week 10** — Deploy to production
- ✔ **Week 11** — Capstone #2 + portfolio polish
- ✔ **Week 12** — Practice exams & certify

**End-of-phase outcome:** a deployed agent, a polished portfolio and a recognised certification.

PHASE 3 · WEEK 9

# Testing, debugging & observability

- ✔ **Days 1–2:** Unit-test your tools in isolation
- ✔ **Days 3–4:** Simulate conversations; assert expected tool calls
- ✔ **Days 5–6:** Trace decision paths, tokens and cost; catch loops
- ✔ **Day 7:** Add logging you can replay to reproduce failures

**Build:** a test + tracing harness around your capstone agent.

PHASE 3 · WEEK 10

# Deploy to production

- ✔ **Days 1–2:** Wrap your agent in an API; add a simple frontend
- ✔ **Days 3–4:** Containerise with Docker; manage secrets safely
- ✔ **Days 5–6:** Add rate limits, retries and circuit breakers
- ✔ **Day 7:** Deploy to a cloud host; add basic monitoring

**Build:** a live, deployed agent with a public URL you can show.

PHASE 3 · WEEK 11

# Capstone #2 + portfolio polish

- ✔ **Days 1–3:** Build a second, different capstone (show range)
- ✔ **Days 4–5:** Polish READMEs, demos and your GitHub profile
- ✔ **Days 6–7:** Update your resume and LinkedIn with the new builds

**Portfolio piece #3:** a second capstone that proves you can apply the skills to more than one problem.

PHASE 3 · WEEK 12

# Practice exams & certify

- ✔ **Days 1–3:** Review the syllabus; close any knowledge gaps
- ✔ **Days 4–5:** Take the practice exams; review every miss
- ✔ **Days 6–7:** Sit the certification exam (40 MCQ, 90 min, 65% to pass)

**Outcome:** a recognised credential plus three portfolio pieces — ready to apply for agentic AI roles.

12 · PORTFOLIO & CAREER

# The portfolio checklist recruiters look for

- ✔ 3+ documented builds, each with a README and a short demo
- ✔ At least one multi-agent or RAG app on a named framework
- ✔ One deployed agent with a live URL
- ✔ Clean, readable code and clear commit history
- ✔ Frameworks named explicitly (LangChain, LangGraph, AutoGen, CrewAI)
- ✔ A credential that verifies the skill set

| $147K | $115–191K | $239K |
|---|---|---|
| median agentic AI pay (US, Glassdoor 2026) | typical range | top earners |

2026 trend: portfolios of real builds beat buzzwords, and governance/safety skills are increasingly paid for.

YOUR NEXT STEP

# Finish the 90 days certified

This roadmap is the plan; the certification is the structure, mentoring and proof that make it stick — 12 modules and 34 hands-on builds that line up with every week above, plus the two capstones.

- ✔ 12 modules · 34 hands-on builds · capstones
- ✔ Daily live sessions & 1-on-1 mentoring
- ✔ Job support · globally recognised · 7-day money-back guarantee

Offer is time-limited. Visit the program page for the current intake.