Why use LCEL? | Langchain

Skip to main content LangChainDocsUse casesIntegrationsAPIMoreCommunityTutorialsContributingAlso by LangChainChat our docsLangSmithLangChain HubLangServePython DocsSearchCTRLKGet startedIntroductionInstallationQuickstartLangChain Expression LanguageInterfaceHow toCookbookWhy use LCEL?LangChain Expression Language (LCEL)ModulesModel I/ORetrievalChainsMemoryAgentsCallbacksModulesSecurityGuidesEcosystemLangChain Expression LanguageWhy use LCEL?Why use LCEL?The LangChain Expression Language was designed from day 1 to support putting prototypes in production, with no code changes, from the simplest “prompt + LLM” chain to the most complex chains (we’ve seen folks successfully running in production LCEL chains with 100s of steps). To highlight a few of the reasons you might want to use LCEL:optimised parallel execution: whenever your LCEL chains have steps that can be executed in parallel (eg if you fetch documents from multiple retrievers) we automatically do it, for the smallest possible latency.support for retries and fallbacks: more recently we’ve added support for configuring retries and fallbacks for any part of your LCEL chain. This is a great way to make your chains more reliable at scale. We’re currently working on adding streaming support for retries/fallbacks, so you can get the added reliability without any latency cost.accessing intermediate results: for more complex chains it’s often very useful to access the results of intermediate steps even before the final output is produced. This can be used let end-users know something is happening, or even just to debug your chain. We’ve added support for streaming intermediate results, and it’s available on every LangServe server.tracing with LangSmith: all chains built with LCEL have first-class tracing support, which can be used to debug your chains, or to understand what’s happening in production. To enable this all you have to do is add your LangSmith API key as an environment variable.PreviousAgentsNextLangChain Expression Language (LCEL)CommunityDiscordTwitterGitHubPythonJS/TSMoreHomepageBlog