Cookbook | Langchain

Skip to main content LangChainDocsUse casesIntegrationsAPIMoreCommunityTutorialsContributingAlso by LangChainChat our docsLangSmithLangChain HubLangServePython DocsSearchCTRLKGet startedIntroductionInstallationQuickstartLangChain Expression LanguageInterfaceHow toCookbookPrompt + LLMMultiple chainsRetrieval augmented generation (RAG)Querying a SQL DBAdding memoryUsing toolsAgentsWhy use LCEL?LangChain Expression Language (LCEL)ModulesModel I/ORetrievalChainsMemoryAgentsCallbacksModulesSecurityGuidesEcosystemLangChain Expression LanguageCookbookCookbookExample code for accomplishing common tasks with the LangChain Expression Language (LCEL). These examples show how to compose different Runnable (the core LCEL interface) components to achieve various tasks. If you're just getting acquainted with LCEL, the Prompt + LLM page is a good place to start.Several pages in this section include embedded interactive screencasts from Scrimba.

They're a great resource for getting started - you can edit the included code whenever you want, just as if you were pair programming with a teacher! Prompt + LLMOne of the most foundational Expression Language compositions is taking: Multiple chainsRunnables can be used to combine multiple Chains together: Retrieval augmented generation (RAG)Let's now look at adding in a retrieval step to a prompt and an LLM, which adds up to a "retrieval-augmented generation" chain: Querying a SQL DBWe can replicate our SQLDatabaseChain with Runnables. Adding memoryThis shows how to add memory to an arbitrary chain. Right now, you can use the memory classes but need to hook them up manually. Using toolsTools are also runnables, and can therefore be used within a chain: AgentsYou can pass a Runnable into an agent.PreviousUse RunnableMapsNextPrompt + LLMCommunityDiscordTwitterGitHubPythonJS/TSMoreHomepageBlog