四次工业革命推动了人类社会发展和变革

- 第一次工业革命（18世纪60年代~19世纪中期，大约是1760年到1860年）：
- 也称为蒸汽时代，发源于英格兰中部地区，是资本主义工业化的早期阶段；
- 蒸汽机的发明及运用成为了这个时代的标志，因此历史学家称这个时代为“蒸汽时代”；
- 蒸汽时代的显著特征是机械化生产和大规模生产，推动了生产力的飞跃，带来了极大的经济增长；
- 这一时期也出现了第一批大型企业和跨国公司；

四次工业革命推动了人类社会发展和变革

- 第二次工业革命（19世纪下半叶~20世纪初，大约1851年到1910年）：
- 也称电气时代，电力、化学、石油等工业得以大发展；
- 这一时期的标志是从机械化向自动化的转变，由于电力的应用和化学工业的发展，生产效率得到了进一步提高，加速了工业化的进程；

四次工业革命推动了人类社会发展和变革

- 第三次工业革命（20世纪后半期，大约1945年到2000年）：
- 也称为信息时代，以原子能、电子计算机、空间技术和生物工程的发明和应用为主要标志，涉及信息技术、新能源技术、新材料技术、生物技术、空间技术和海洋技术等诸多领域的一场信息控制技术革命；
- 这一时期的特点是信息技术和数字技术的广泛应用，推动了社会生产力的又一次飞跃；

四次工业革命推动了人类社会发展和变革

- 第四次工业革命（21世纪初至今，大约2000年开始直到现在的这段时间）：
- 也称工业4.0时代或智能化时代，这个时代以互联网产业化，工业智能化，工业一体化为代表，以人工智能，清洁能源，无人控制技术，量子信息技术，虚拟现实以及生物技术为主的全新技术革命;
- 这一阶段主要依赖于互联网、物联网、大数据、人工智能等新兴技术的深度融合和创新应用，旨在实现更高效、更智能、更可持续的生产和生活方式;

什么是人工智能?

- A system is ability to correctly interpret external data, to learn from such data, and to use those learnings to achieve specific goals and tasks through flexible adaptation.
- 系统正确解释外部数据的能力，从这些数据中学习的能力，以及通过灵活的适应利用这些学习来实现特定目标和任务的能力；
- 人工智能（Artificial Intelligence，缩写为AI）是一门新的技术科学，旨在开发、研究用于模拟、延伸和扩展人的智能的理论、方法、技术及应用系统，它结合了数学、计算机科学、心理学等多学科的理论，通过让计算机模拟人类的思考和行为过程，实现人机交互，提高计算机的智能水平，以更好地服务于人类社会；

人工智能的发展历程

- **起步发展期**：1943年—20世纪60年代 （1943 - 1969年）
  - 1943年，提出神经元的数学模型，这是现代人工智能学科的奠基石之一；
  - 1950年，艾伦·麦席森·图灵（Alan Mathison Turing）提出“图灵测试” （测试机器是否能表现出与人无法区分的智能），让机器产生智能这一想法开始进入人们的视野；
  - 1956年，正式使用了人工智能（artificial intelligence，AI）这一术语；
- **反思发展期**：20世纪70年代（1970 - 1979年）
  - 计算力及理论等的匮乏使得不切实际目标的落空，人工智能的发展走入低谷；

人工智能的发展历程

- **应用发展期**：20世纪80年代 （1980 - 1989年）
  - 人工智能走入应用发展的新阶段，专家系统模拟人类专家的知识和经验解决特定领域的问题，实现了人工智能从理论研究走向实际应用、从一般推理策略探讨转向运用专门知识的重大突破，而机器学习(特别是神经网络)探索不同的学习策略和各种学习方法，在大量的实际应用中也开始慢慢复苏；
- **平稳发展期**：20世纪90年代—2010年（1990 - 2010年）
  - 由于互联网技术的迅速发展，加速了人工智能的创新研究，促使人工智能技术进一步走向实用化，人工智能相关的各个领域都取得长足进步；

人工智能的发展历程

- **蓬勃发展期**：2011年至今
  - 随着大数据、云计算、互联网、物联网等信息技术的发展，泛在感知数据和图形处理器等计算平台推动以深度神经网络为代表的人工智能技术飞速发展，大幅跨越了科学与应用之间的技术鸿沟，诸如图像分类、语音识别、知识问答、人机对弈、无人驾驶等人工智能技术实现了重大的技术突破，迎来爆发式增长的新高潮；
  - 2015年，马斯克等人共同创建OpenAI，它是一个非营利的研究组织，使命是确保通用人工智能 (即一种高度自主且在大多数具有经济价值的工作上超越人类的系统）将为全人类带来福祉，其发布热门产品的如：OpenAI Gym，GPT等；
  - 2016年，AlphaGo与围棋世界冠军、职业九段棋手李世石进行围棋人机大战，以4比1的总比分获胜；
  - 2022年11月30日，OpenAI研发的一款聊天机器人程序ChatGPT对外发布，引发AI的大爆发；
  - 2023年3月15日，OpenAI发布ChatGPT 4.0，引爆了AI；
  - 2023年3月16日，百度发布文心一言，文心一言（ERNIE Bot）是基于文心大模型技术推出的生成式对话产品，文心大模型是百度自主研发的产业级知识增强大模型，文心一言能够与人对话互动，回答问题，协助创作，高效便捷地帮助人们获取信息、知识和灵感；
  - 国内还有：科大讯飞认知智能大模型、阿里巴巴通义千问、华为盘古大模型、360智脑、京东言犀大模型等等；

什么是大模型?

- 大模型，是指具有大规模参数和复杂计算结构的机器学习模型。这些模型通常由深度神经网络构建而成，拥有数十亿甚至数千亿个参数。其设计目的在于提高模型的表达能力和预测性能，以应对更加复杂的任务和数据；
- 大模型，简单来说，就是一个特别聪明、特别能干的“大脑”，这个“大脑”由很多个小小的“神经元”组成，每个“神经元”都能处理一部分信息，当这些“神经元”一起工作时，大模型就能理解并回答各种问题，或者完成各种复杂的任务。就像你有一个超级聪明的助手，它能帮你写邮件、写PPT、回答你的各种问题等等，它就像是一个上知天文，下知地理，无所不知的人；

如何训练大模型?

- 要训练一个大模型不容易，需要给它提供很多学习材料，就像我们小时候读书学习一样。而且为了让这个“大脑”更聪明，还需要很多高级的计算机设备来帮助它学习；
- 训练大模型：
  - 高性能的CPU和GPU，多核心和高主频的CPU以及支持CUDA的GPU加速训练过程；
  - 大容量存储设备，训练大模型需要存储大量的数据集、模型参数和中间结果；
  - 高速网络连接，通过网络连接将训练任务分配到多个计算节点上；
  - 深度学习框架，如TensorFlow、PyTorch等，这些框架提供了构建和训练模型的工具和库；
  - 分布式训练框架，为了加速大模型的训练，可以使用分布式训练框架，如Horovod、Ray等；
  - 编程语言和工具，Python是深度学习领域最常用的编程语言，还有（如Git）来管理代码和版本迭代；
  - 训练大模型非常耗电，高性能计算机和GPU进行长时间的工作，需要消耗大量的电力；

What is Spring AI?

- 官网：https://spring.io/
- Spring AI is an application framework for AI engineering. Its goal is to apply to the AI domain Spring ecosystem design principles such as portability and modular design and promote using POJOs as the building blocks of an application to the AI domain.
- Spring AI是一个AI工程领域的应用程序框架；
- 它的目标是将Spring生态系统的设计原则应用于人工智能领域，比如Spring生态系统的可移植性和模块化设计，并推广使用POJO来构建人工智能领域应用程序；
- Spring AI并不是要构建一个自己的AI大模型，而是让你对接各种AI大模型；

Spring AI的主要特点

- Spring AI提供的API支持跨人工智能提供商的 聊天，文本到图像，和嵌入模型等，同时支持同步和流API选项；
- 1、Chat Models 聊天模型：
  - OpenAI
  - Azure Open AI
  - Amazon Bedrock
  - Cohere's Command
  - AI21 Labs' Jurassic-2
  - Meta's LLama 2
  - Amazon's Titan
  - Google Vertex AI Palm
  - Google Gemini
  - HuggingFace - access thousands of models, including those from Meta such as Llama2
  - Ollama - run AI models on your local machine
  - MistralAI

Spring AI的主要特点

- 2、Text-to-image Models 文本到图像模型：
  - OpenAI with DALL-E
  - StabilityAI
- 3、Transcription (audio to text) Models 转录（音频到文本）模型
  - OpenAI
- 4、Embedding Models 嵌入模型
  - OpenAI
  - Azure OpenAI
  - Ollama
  - ONNX
  - PostgresML
  - Bedrock Cohere
  - Bedrock Titan
  - Google VertexAI
  - Mistal AI

Spring AI的主要特点

- 5、Vector Store API提供了跨不同提供商的可移植性，其特点是提供了一种新颖的类似SQL的元数据过滤API，以保持可移植性；
- 矢量数据库：
  - Azure Vector Search
  - Chroma
  - Milvus
  - Neo4j
  - PostgreSQL/PGVector
  - PineCone
  - Redis
  - Weaviate
  - Qdrant

Spring AI的主要特点

- 6、用于AI模型和矢量存储的Spring Boot自动配置和启动器；（xxxx-spring-ai-starter）
- 7、函数调用，您可以声明java.util.Function的OpenAI模型的函数实现，用于其提示响应。如果在应用程序上下文中注册为@Bean，则可以直接将这些函数作为对象提供，或者引用它们的名称。这一功能最大限度地减少了不必要的代码，并使人工智能模型能够要求更多信息来完成其响应；
- 支持的模型有：
  - OpenAI
  - Azure OpenAI
  - VertexAI
  - Mistral AI

Spring AI的主要特点

8、用于数据工程的ETL框架

- ETL框架的核心功能是使用Vector Store促进文档向模型提供者的传输。ETL框架基于Java函数式编程概念，可帮助您将多个步骤链接在一起；
- 支持阅读各种格式的文档，包括PDF、JSON等；
- 该框架允许数据操作以满足您的需求。这通常包括拆分文档以遵守上下文窗口限制，并使用关键字增强它们以提高文档检索效率；
- 最后，处理后的文档存储在矢量数据库中，以便将来检索；

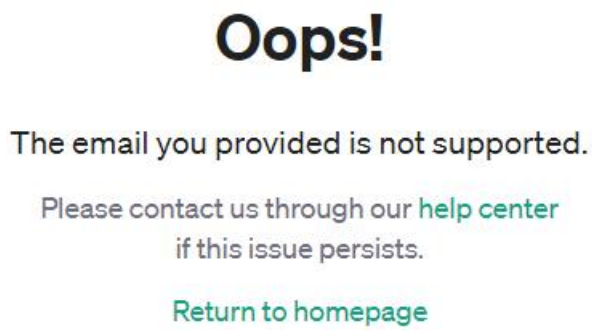
Spring AI的主要特点

- 9、广泛的参考文档、示例应用程序和研讨会/课程材料；
  - 未来的版本将在此基础上提供对其他人工智能模型的访问，例如，谷歌刚刚发布的Gemini多模式模态，一个评估人工智能应用程序有效性的框架，更方便的API，以及帮助解决“查询/汇总我的文档”用例的功能。有关即将发布的版本的详细信息，请查看GitHub；

开发Spring AI程序的前期准备

- 1、本机电脑要可以访问OpenAI网站 https://openai.com/； （科学上网）
- 2、要有OpenAI的API Key； （注册账号或者购买）

- API-Key：sk-3sfER03LDLG3SDFsdlwe283JSdw023lkrmrHDND32fmREKFD



开发Spring AI应用程序

- Spring AI应用程序也是基于Spring Boot进行开发；
- 1、建项目：创建一个Spring Boot项目；
- 2、加依赖：加入spring-ai-openai-spring-boot-starter依赖；
- **ai依赖：**

```
<dependency>
    <groupId>org.springframework.ai</groupId>
    <artifactId>spring-ai-openai-spring-boot-starter</artifactId>
</dependency>
```

开发Spring AI程序

- **继承父项目：**

```
<dependencyManagement>
    <dependencies>
        <dependency>
            <groupId>org.springframework.ai</groupId>
            <artifactId>spring-ai-bom</artifactId>
            <version>${spring-ai.version}</version>
            <type>pom</type>
            <scope>import</scope>
        </dependency>
    </dependencies>
</dependencyManagement>
```

开发Spring AI程序

- **配置项目依赖下载的仓库：**

```
<repositories>
    <repository>
        <id>spring-milestones</id>
        <name>Spring Milestones</name>
        <url>https://repo.spring.io/milestone</url>
        <snapshots>
            <enabled>false</enabled>
        </snapshots>
    </repository>
</repositories>
```

开发Spring AI程序

- 3、配文件
- spring:
- ai:
- openai:
- api-key: sk-3sfER03LDLG3SDFsdlwe283JSdw023lkrmrHDND32fmREKFD （换成你的api-key）
- base-url: https://api.openai.com

开发Spring AI程序

- 4、写代码
  - 注入OpenAiChatClient
    - @Resource
    - private OpenAiChatClient openAiChatClient;
  - 调用call方法
    - openAiChatClient.call(message);
- 5、去运行

开发Spring AI程序

- **OpenAI Chat的客户端call方法**
- openAiChatClient.call(message);

- openAiChatClient.call(new Prompt(msg, OpenAiChatOptions.builder()
- .withModel("gpt-4-32k") //gpt的版本，32k是参数量
- .withTemperature(0.4F) //温度越高，回答得比较有创新性，但是准确率会下降，温度越低，回答的准确率会更好
- .build()));

开发Spring AI程序

- **OpenAI Chat的客户端stream方法**
- openAiChatClient.stream(message);

- openAiChatClient.stream(new Prompt(msg, OpenAiChatOptions.builder()
- .withModel("gpt-4-32k") //gpt的版本，32k是参数量
- .withTemperature(0.4F) //温度越高，回答得比较有创新性，但是准确率会下降，温度越低，回答的准确率会更好
- .build()));

开发Spring AI程序-图像

- 写代码
  - 注入OpenAiImageClient
    - @Resource
    - private OpenAiImageClient openAiImageClient;
  - 调用call方法
    - openAiImageClient.call(message);

- 购买的api-key文档：https://jvuw7eyf9vx.feishu.cn/docx/XA3adyb2JoLnUDxANm4cCdhNnCb
- model: gpt-4-dalle

开发Spring AI程序-语音到文本

- 写代码
  - 注入OpenAiImageClient
    - @Resource
    - private OpenAiAudioTranscriptionClient openAiAudioTranscriptionClient;
  - 调用call方法
    - Resource audioFile = new ClassPathResource("cat.mp3");
    - openAiAudioTranscriptionClient.call(audioFile);

开发Spring AI程序-文本到语音

- 写代码
  - 注入OpenAiAudioSpeechClient
    - @Resource
    - private OpenAiAudioSpeechClient openAiAudioSpeechClient;
  - 调用call方法
    - openAiAudioSpeechClient.call(text);

开发Spring AI程序-多模态

- **多模态API**
  - 多模态是指模型同时理解和处理来自各种来源的信息的能力，包括文本、图像、音频和其他数据格式；
  - 多模式大语言模型（LLM）特征使模型能够结合其他模态（如图像、音频或视频）来处理和生成文本；
  - Spring AI 多模态API提供了所有必要的统一抽象和代码封装来支持多模式LLM；

开发Spring AI程序-多模态

- **多模态API**
  - 图片url：

    https://i.bjpowernode.com/static/uploads/index/course/20240305/1709618214@e73b6675913ead01a9a813cfbbf424cc.jpg
  - 请描述一下这张图片里面主要是什么？

  - http://localhost:8080/ai/multi?msg=%E8%AF%B7%E6%8F%8F%E8%BF%B0%E4%B8%80%E4%B8%8B%E8%BF%99%E5%BC%A0%E5%9B%BE%E7%89%87%E9%87%8C%E9%9D%A2%E4%B8%BB%E8%A6%81%E6%98%AF%E4%BB%80%E4%B9%88%EF%BC%9F&imageUrl=https://i.bjpowernode.com/static/uploads/index/course/20240305/1709618214@e73b6675913ead01a9a813cfbbf424cc.jpg

大模型工具Ollama

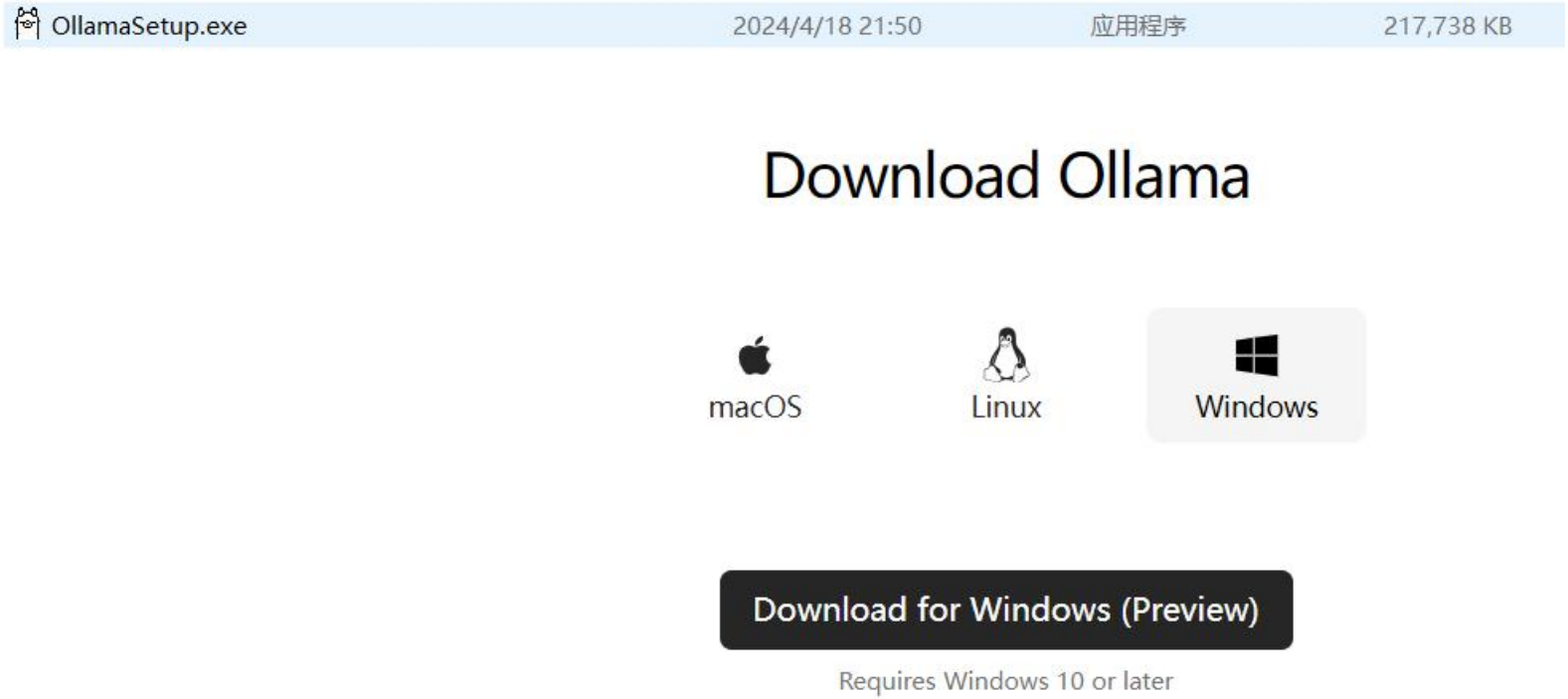
- 官网：https://ollama.com/
- Ollama是一个用于部署和运行各种开源大模型的工具；
- 它能够帮助用户快速在本地运行各种大模型，极大地简化了大模型在本地运行的过程。
- 用户通过执行几条命令就能在本地运行开源大模型，如Llama 2等；
- 综上，Ollama是一个大模型部署运行工具，在该工具里面可以部署运行各种大模型，方便开发者在本地搭建一套大模型运行环境；

大模型工具Ollama

- **下载：**https://ollama.com/download
- 说明：Ollama的运行会受到所使用模型大小的影响；
- 1、例如，运行一个7B（70亿参数）的模型至少需要8GB的可用内存(RAM)，而运行一个13B（130亿参数）的模型需要16GB的内存，33B（330亿参数）的模型需要32GB的内存；
- 2、需要考虑有足够的磁盘空间，大模型的文件大小可能比较大，建议至少为Ollama和其模型预留50GB的磁盘空间；
- 3、性能较高的CPU可以提供更好的运算速度和效率，多核处理器能够更好地处理并行任务，选择具有足够核心数的CPU；
- 4、显卡（GPU）：Ollama支持纯CPU运行，但如果电脑配备了NVIDIA GPU，可以利用GPU进行加速，提高模型的运行速度和性能；

大模型工具Ollama

- **安装：**
- 点击OllmaSetup.exe进行安装



大模型工具Ollama

- 运行：

```
Welcome to Ollama!

Run your first model:

        ollama run llama2

PS C:\Windows\system32>
```

- ollama run qwen:0.5b-chat　（大模型的名字去ollama官网找：https://ollama.com/library）

大模型工具Ollama

- **Spring AI代码测试**
  - 默认Ollama api会监听11434端口，可以使用命令进行查看：
  - netstat -ano | findstr 11434

Spring AI使用Ollama

- 加依赖
- <dependency>
- <groupId>org.springframework.ai</groupId>
- <artifactId>spring-ai-ollama-spring-boot-starter</artifactId>
- </dependency>
- 写代码
  - 注入OllamaChatClient
    - @Resource
    - private OllamaChatClient ollamaChatClient;
  - 调用call方法
    - ollamaChatClient.call(msg);

Ollama 的 Web & Desktop

- **Web & Desktop**
  - Ollama的Web & Desktop非常多，比较流行的是 **Open WebUI**；
  - Open WebUI Github：https://github.com/open-webui/open-webui
  - Open WebUI 官网：https://www.openwebui.com/
  - Open WebUI是一个可扩展、功能丰富、用户友好的自托管WebUI，它支持完全离线操作，支持各种LLM（Large Language Model）运行程序，包括Ollama和OpenAI兼容的API；

Ollama 的 Web & Desktop

- 搭建部署Open WebUI有两种方式：
  - 1、Docker方式；（官方推荐的方式）
  - 2、源码部署安装方式；（文档：https://docs.openwebui.com/getting-started/ ）

Docker Desktop

- Window下使用Docker推荐使用Docker Desktop；
- Docker Desktop是一款适用于Windows操作系统的桌面应用，它为开发人员提供了一个界面化操作Docker的环境，以便在本地环境中轻松创建、构建和运行Docker容器；

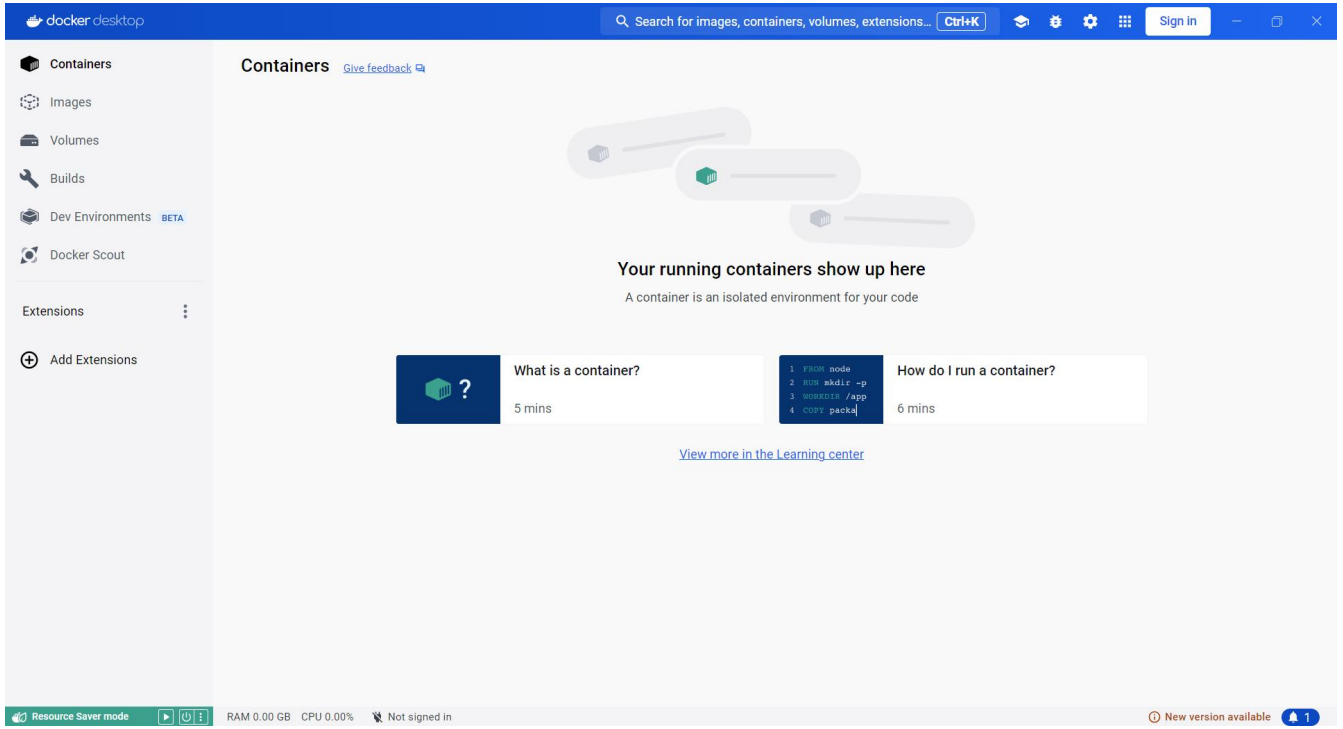
- Windows系统下Docker Desktop的安装
- 官网下载安装包：https://www.docker.com/products/docker-desktop/

Docker Desktop

- Docker Desktop的安装：

Docker Desktop Installer.exe

- 双击该文件下一步下一步即完成安装；



Docker部署Open WebUI

- 在docker中运行Open WebUI
- docker run -d -p 3000:8080 --add-host=host.docker.internal:host-gateway -v D:\dev\open-webui:/app/backend/data --name open-webui --restart always ghcr.io/open-webui/open-webui:main
  - 这是一个 docker run 命令，用于启动一个新的 Docker 容器，下面是这个命令各个部分的解释：
  - docker run：这是 Docker 的命令，用于从指定的镜像启动一个新的容器；
  - -d：表示在“分离”模式下运行容器，即后台运行；
  - -p 3000:8080：端口映射，表示将宿主机的3000端口映射到容器的8080端口，当你访问宿主机的3000端口时，实际上会访问容器内的8080端口；
  - --add-host=host.docker.internal:host-gateway：这个选项向容器的 /etc/hosts 文件中添加一条记录，这通常用于让容器能够解析到宿主机的名称，并且将其 IP 地址设置为宿主机的网关地址，这在某些网络配置中很有用，尤其是当容器需要知道宿主机的地址时；
  - -v D:\dev\open-webui:/app/backend/data：卷挂载，这表示将宿主机的 D:\dev\open-webui 目录挂载到容器内的 /app/backend/data 目录，这样，容器和宿主机之间可以共享这个目录中的数据；
  - --name open-webui：为容器指定一个名称，这里是 open-webui；
  - --restart always：这个选项告诉 Docker 在容器退出时总是自动重启它，无论容器是因为何种原因退出，它都会自动重启；
  - ghcr.io/open-webui/open-webui:main：这是你要运行的 Docker 镜像的完整名称，ghcr.io 是 GitHub Container Registry 的地址，open-webui/open-webui 是镜像的仓库和名称，main是标签，通常表示该镜像的最新或主分支版本；

Lobe Chat 界面框架

- 官网：https://lobehub.com/
- Github：https://github.com/lobehub/lobe-chat

- Built for you the Super Individual （专为你打造的超级个人）
- 现代化设计的开源 ChatGPT/LLMs 聊天应用与开发的UI框架；
- 支持语音合成、多模态、可扩展的（function call）插件系统；
- 一键免费拥有你自己的 ChatGPT/Gemini/Claude/Ollama 应用；

Lobe Chat 界面框架

- Lobe Chat 部署
  - 1、使用 Vercel、Zeabur 或 Sealos 部署；
  - 2、使用 Docker 部署；
    - docker run -d -p 3210:3210 -e OPENAI_API_KEY=sk-xxxx -e ACCESS_CODE=lobe66 --name lobe-chat lobehub/lobe-chat
  - 完整的部署文档：https://lobehub.com/zh/docs/self-hosting/start

大语言模型的选择

- 大语言模型主要分为国外大模型 和 国内大模型；
- 国外大模型，可能受到一些限制，或者不稳定；
- 国内也有非常优秀的大模型，国内大模型排行榜：
  - https://www.superclueai.com/
  - 基于中文语言理解测评基准，包括代表性的数据集、基准(预训练)模型、语料库、排行榜；
  - 选择一系列有一定代表性的任务对应的数据集，做为测试基准的数据集，这些数据集会覆盖不同的任务、数据量、任务难度；