

TokenHub

平台功能



腾讯云

【 版权声明 】

©2013–2026 腾讯云版权所有

本文档（含所有文字、数据、图片等内容）完整的著作权归腾讯云计算（北京）有限责任公司单独所有，未经腾讯云事先明确书面许可，任何主体不得以任何形式复制、修改、使用、抄袭、传播本文档全部或部分内容。前述行为构成对腾讯云著作权的侵犯，腾讯云将依法采取措施追究法律责任。

【 商标声明 】



及其它腾讯云服务相关的商标均为腾讯云计算（北京）有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标，依法由权利人所有。未经腾讯云及有关权利人书面许可，任何主体不得以任何方式对前述商标进行使用、复制、修改、传播、抄录等行为，否则将构成对腾讯云及有关权利人商标权的侵犯，腾讯云将依法采取措施追究法律责任。

【 服务声明 】

本文档意在向您介绍腾讯云全部或部分产品、服务的当时的相关概况，部分产品、服务的内容可能不时有所调整。您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或默示的承诺或保证。

【 联系我们 】

我们致力于为您提供个性化的售前购买咨询服务，及相应的技术售后服务，任何问题请联系 4009100100或 95716。

文档目录

平台功能

模型广场

体验中心

AI 创作

在线推理

用量统计

模型监控

API Key 管理

Coding Plan

Coding Plan 概述

Coding Plan 接入 AI 工具

OpenClaw

CodeBuddy Code

OpenCode

Claude Code

Codex

Cline

Cursor

Kilo CLI

Kilo Code

Coding Plan 常见问题

平台功能

模型广场

最近更新时间：2026-04-03 16:27:22

模型广场 集中展示了平台所有可用的大模型，您可以按模型类型、服务商维度快速筛选模型。

TokenHub

模型广场

新用户福利免费体验

MiniMax-M2.7

自行构建复杂 Agent Harness, 完成高度复杂的任务 **文本生成**

发布时间: 2026-03-18

kimi-k2.5

Kimi 迄今最智能、最全能的模型 **深度思考**

发布时间: 2026-01-27

全部 语言模型 视觉模型 全部服务商 (22) 搜索模型名称

HY 2.0 Think

混元

深度思考 **文本生成**

擅长文创、多轮、指令遵循实用场景。数理、代码、Agent 能力显著增强。

发布时间: 2025-11-09

HY 2.0 Instruct

混元

文本生成

模型底座全面升级, 基础能力更稳健。知识、数学、写作与推理达一线水准。指令遵循、多轮及长文理解表现优异。

发布时间: 2025-11-11

Hunyuan-role

混元

角色扮演 **文本生成**

针对角色扮演场景, 人设高度贴合、极致的拟人对话。具有吸引力的剧情演绎和推进、情感陪伴与满足。

发布时间: 2026-03-04

模型详情

单击任一模型卡片，即可查看模型详细信息，包括：

- **模型介绍**：模型名称、模型能力标签、模型基本信息。
- **模型价格**：模型的按量计费价格，不同模型价格维度可能不同。
- **模型能力**：展示模型的能力维度。
- **输入输出类型**：展示模型支持的输入输出格式。
- **模型限制**：模型上下文窗口、最大输入长度、并发限制等信息。
- **调用方式**：通过快捷复制调用示例代码，将 `YOUR_API_KEY` 替换为您的 API Key，实现快速调用。

模型对比

在模型详情页，您可以通过单击右上角的**模型对比**，选择多个模型来对比参数，帮助您快速了解各模型差异：

模型对比 对比 2 个模型 (最多支持 4 个)

+ 添加模型(2/4)

HY 2.0 Instruct

模型体验 文本生成

↑

查看详情 立即体验

Deepseek-v3.2

模型体验 深度思考 文本生成

↑

查看详情 立即体验

模型价格

推理输入	¥3.18 / 百万tokens
推理输出	¥7.95 / 百万tokens

推理输入	¥2 / 百万tokens
推理输出	¥3 / 百万tokens

模型能力

模型体验 ①	☑
函数调用 ①	☑
文本输入 ①	☑
文本输出 ①	☑
文本生成 ①	☑
深度思考 ①	☒
结构化输出 ①	☒
可切换 ①	☒

模型体验 ①	☑
函数调用 ①	☑
文本输入 ①	☑
文本输出 ①	☑
文本生成 ①	☑
深度思考 ①	☑
结构化输出 ①	☑
可切换 ①	☑

模型限制

上下文窗口	144k
最大输入 Tokens	128k
最大输出 Tokens	16k
最大推理 Tokens	-
TPM	-
RPM	-
IPM	-
并发数	5

上下文窗口	128k
最大输入 Tokens	96k
最大输出 Tokens	32k
最大推理 Tokens	-
TPM	30w
RPM	-
IPM	-
并发数	-



体验中心

最近更新时间：2026-04-03 21:02:21

体验中心可快速体验不同类型模型效果，帮助您选择合适的模型。您可以在 [语言模型](#) 中体验生文类模型，在 [视觉模型](#) 中体验图片生成、视频生成、3D 效果生成相关的模型功能。

使用前提

在体验中心体验模型会消耗模型用量，对于支持免费体验的模型，首次使用时平台会自动领取免费体验包，详情请参见 [新人免费体验包](#)。

对于不支持免费体验的模型，您需要先在 [在线推理服务](#) 页面，找到该模型并在操作列中单击启用后付费。

语言模型

您可以在 [语言模型](#) 页面与平台接入的各类大语言模型进行实时对话，验证模型的理解、推理和创作能力。



序号	功能	说明
1	选择模型	单击模型名称右侧的切换按钮，即可打开模型列表，从列表中浏览并选择您希望体验的模型。 <div style="border: 1px solid #00aaff; padding: 5px; margin-top: 10px;"> <p>说明：</p> </div>

		如果您是从模型广场的模型详情页单击 立即体验 跳转而来，系统会自动预选对应模型，无需手动选择。
2	模型对比	在模型选择栏右侧，单击 模型对比 ，可以同时最多对比4个模型，对同一问题的回答结果进行横向对比。
3	深度思考	深度思考默认关闭。开启后（深度思考图标显示为蓝色代表开启），模型将进行更深入的推理分析，适合需要复杂逻辑推理、数学计算或专业分析的场景。 注意： 深度思考模式下，模型的响应时间可能较长，但回答质量通常会更高。
4	灵感推荐	提供多个灵感推荐内容，包含预设的示例问题。这些示例覆盖不同使用场景，帮助您快速了解模型的能力范围。

视觉模型

图像生成

提供 AI 图像生成与处理能力的技术服务，整合了文生图、单图生图、多图生图等能力，深度理解意图，角色、画面可保持高度一致性稳定输出，为高质量的图像内容创作、内容运营提供技术支持。




配置选项

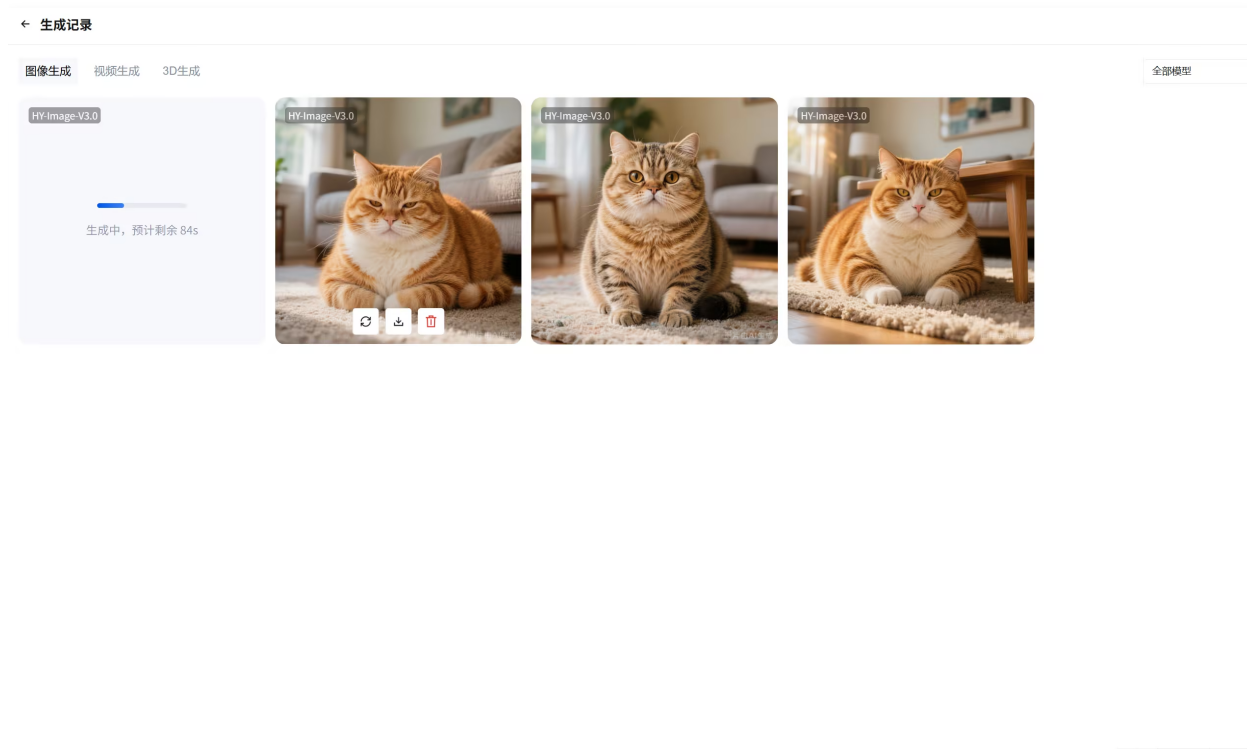
配置项	说明
图片上传	<ul style="list-style-type: none"> 支持以图生图。

	<ul style="list-style-type: none"> • 支持 jpg、jpeg、png、webp 格式。 • 单次最多可以上传 3 张。 • 单图 Base64 编码后 ≤10M。
自动 Prompt 改写	勾选后，可自动优化提示词，提升生成效果匹配度。
生成比例	支持3:4、4:3、1:1、9:16、16:9，适配不同场景尺寸。
生成数量	支持 1-4 张批量生成。

操作步骤

1. 进入功能：进入 [视觉模型](#) 页面，选择**图像生成**页签。
2. 选择模型：在**选择模型**下拉框，选择生图模型（例如 HY-Image-V3.0等）。
3. 图片上传（可选）：单击**图片上传**，上传参考图。
4. 填写提示词：**必填**。在**提示词**输入框，填写图像描述。您可以勾选**自动 Prompt 改写**优化指令。
5. 配置参数：选择**生成尺寸**、**生成数量**。
6. 生成图像：单击**立即生成**，等待模型输出结果。
7. 结果操作：您可以在页面右上角单击  图标，查看图像生成记录或对生成的图像进行管理。

生成记录



生成记录以任务卡片的形式呈现，展示生成作品缩略图、使用模型、任务状态（已完成 / 生成中 / 生成失败）。您可以对生成的结果执行如下操作：


控件	功能
重新生成（循环图标）	基于当前任务的提示词/参数，重新触发 AI 生成图像。
下载	下载生成的图像。
删除	删除该条生成记录及对应作品。

视频生成

该服务基于腾讯视频生成大模型等一系列领先的音视频 AI 技术，支持高质量地生成或处理视频内容。既能帮助专业视频创作者降低制作成本、发现视频创意，又能提升视频社交娱乐的趣味性。可广泛应用于短视频平台、影视制作、广告营销、社交媒体、游戏等领域。



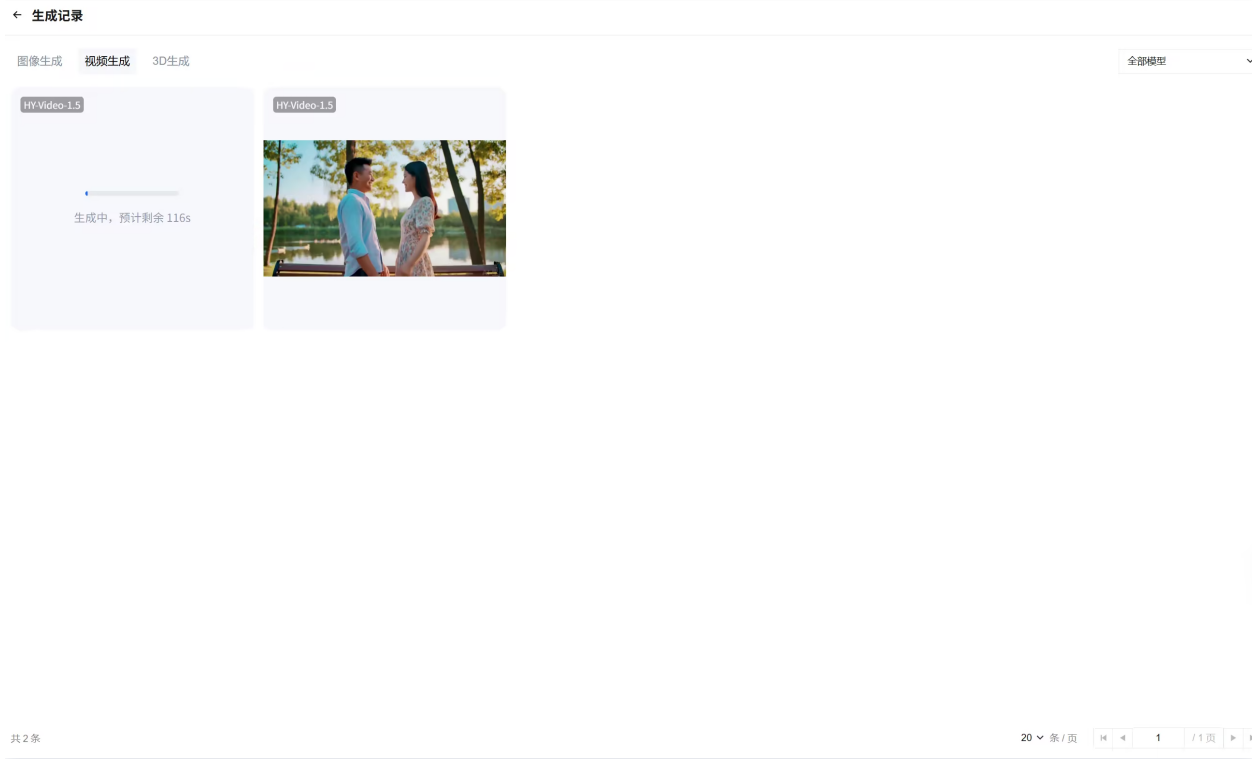
操作步骤

1. 进入功能：进入 [视觉模型](#) 页面，选择[视频生成](#)页签。
2. 选择模型：在[选择模型](#)下拉框，选择生视频模型（例如 Hunyuan-Video-1.5 混元生视频等）。
3. 图片上传：单击[图片上传](#)，上传参考图。
4. 填写提示词：**必填**。在[提示词](#)输入框，填写视频创意描述。
5. 参考同款（可选）：在页面右侧单击示例图下方的[做同款](#)，可一键复用对应视频的创意 prompt。
6. 生成视频：单击[立即生成](#)，等待模型输出结果。
7. 结果操作：您可以在页面右上角单击  图标，查看视频生成记录或对生成的视频进行管理。

生成记录

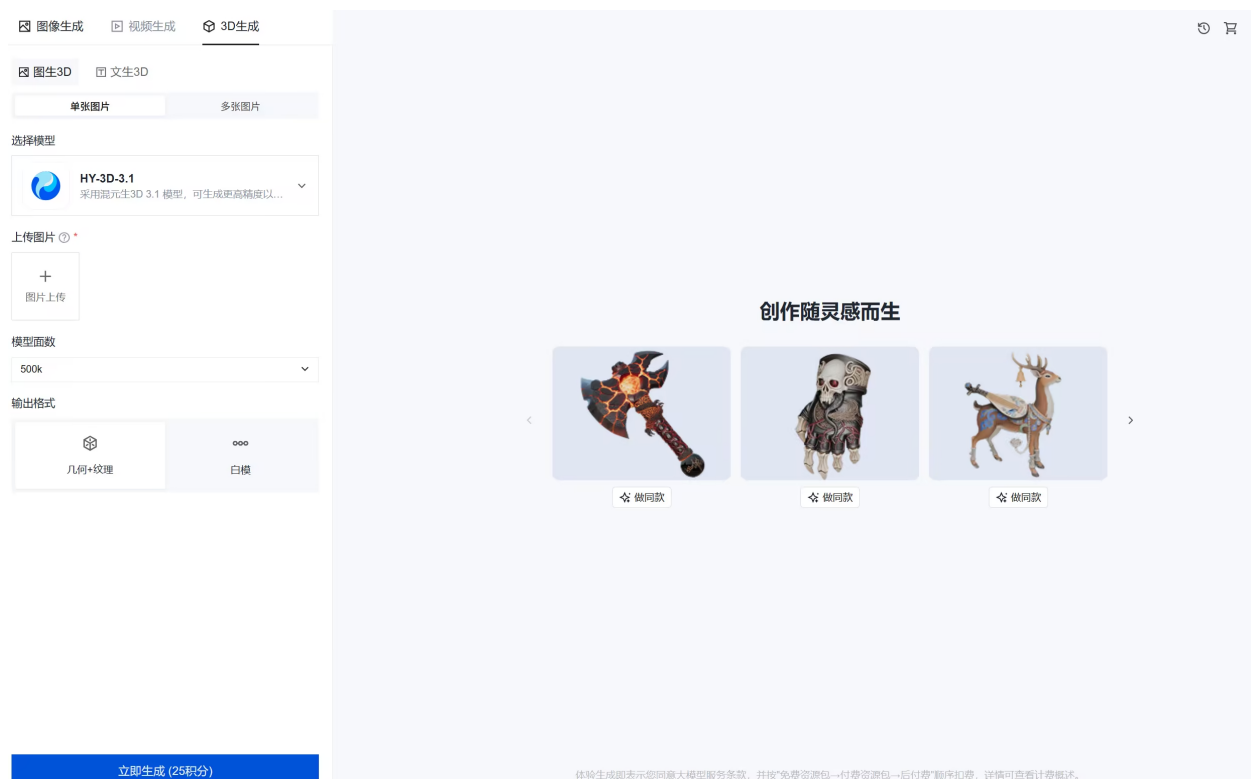
生成记录以任务卡片的形式呈现，展示生成的视频、使用模型、任务状态（已完成 / 生成中 / 生成失败）。您可以对生成的结果执行如下操作：

控件	功能
重新生成 (循环图标)	基于当前任务的提示词/参数, 重新触发 AI 生成视频。
下载	下载生成的原视频文件。
删除	删除该条生成记录及对应作品。



3D 生成

通过文本或图片, 生成几何结构精准, 纹理细节丰富的3D 模型。




配置选项

参数	参数说明
模型面数	<ul style="list-style-type: none"> 控制 3D 模型的精细度，面数越高模型越精细。 支持1.5m、1m、500k、50k。
输出格式	<ul style="list-style-type: none"> 几何 + 纹理：生成带纹理的完整 3D 模型。 白模：仅生成几何白模。

操作步骤

一、图生 3D 模式


- 进入功能**：进入 [视觉模型](#) 页面，选择**3D 生成 > 图生 3D**。
- 选择模型**：在**选择模型**下拉框，选择生3D 模型（例如 HY-3D-3.1 等）。
- 图片上传**，您可以选择如下方式：
 - 单张图片**：选择单张图片输入方式后，单击**图片上传**，上传单张参考图。
 - 多张图片**：选择多张图片输入方式后，单击**添加多视图**，按指引上传顶图、正面图、左 / 右 45° 图、左 / 右图、背图、底图等多视角参考图，单击**确定**。
- 配置参数**：选择**模型面数**、**输出格式**。
- 参考同款（可选）**：在页面右侧单击示例图下方的**做同款**，一键复用对应 3D 模型的生成参数。
- 生成3D**：单击**立即生成**，等待模型输出 3D 资产。

7. 结果操作：您可以在页面右上角单击  图标，查看3D 生成记录或对生成的3D 资产进行管理。

二、文生 3D 模式

1. 进入功能：进入 [视觉模型](#) 页面，选择**3D 生成 > 文生 3D**。


2. 选择模型：在**选择模型**下拉框，选择生3D 模型（例如 HY-3D-3.1 等）。

3. 填写提示词：**必填**。在**提示词**输入框，按**单主体 + 特征 + 风格结构**填写 3D 模型描述（可单击  获取提示词指南）。

4. 配置参数：选择**模型面数、输出格式**。

5. 参考同款（可选）：在页面右侧单击示例图下方的**做同款**，一键复用对应 3D 模型的生成参数。

6. 生成3D：单击**立即生成**，等待模型输出 3D 资产。

7. 结果操作：您可以在页面右上角单击  图标，查看3D 生成记录或对生成的3D 资产进行管理。

生成记录

生成记录以任务卡片的形式呈现，展示生成作品缩略图、使用模型、任务状态（已完成 / 生成中 / 生成失败）。其中：对于在“生成中”的任务，若为文生3D，则会显示用户的提示词 + 生成进度；若为图生3D，会显示参考图 + 生成进度。

您可以对生成的结果执行如下操作：

控件	功能
重新生成 （循环图标）	基于当前任务的提示词/参数，重新触发 AI 生成3D 资产。
下载	下载生成的3D 资产。
删除	删除该条生成记录及对应作品。


图像生成 视频生成 3D生成

全部模型

HY-3D-3.1

生成一只小狗，通用
生成中...

HY-3D-3.1

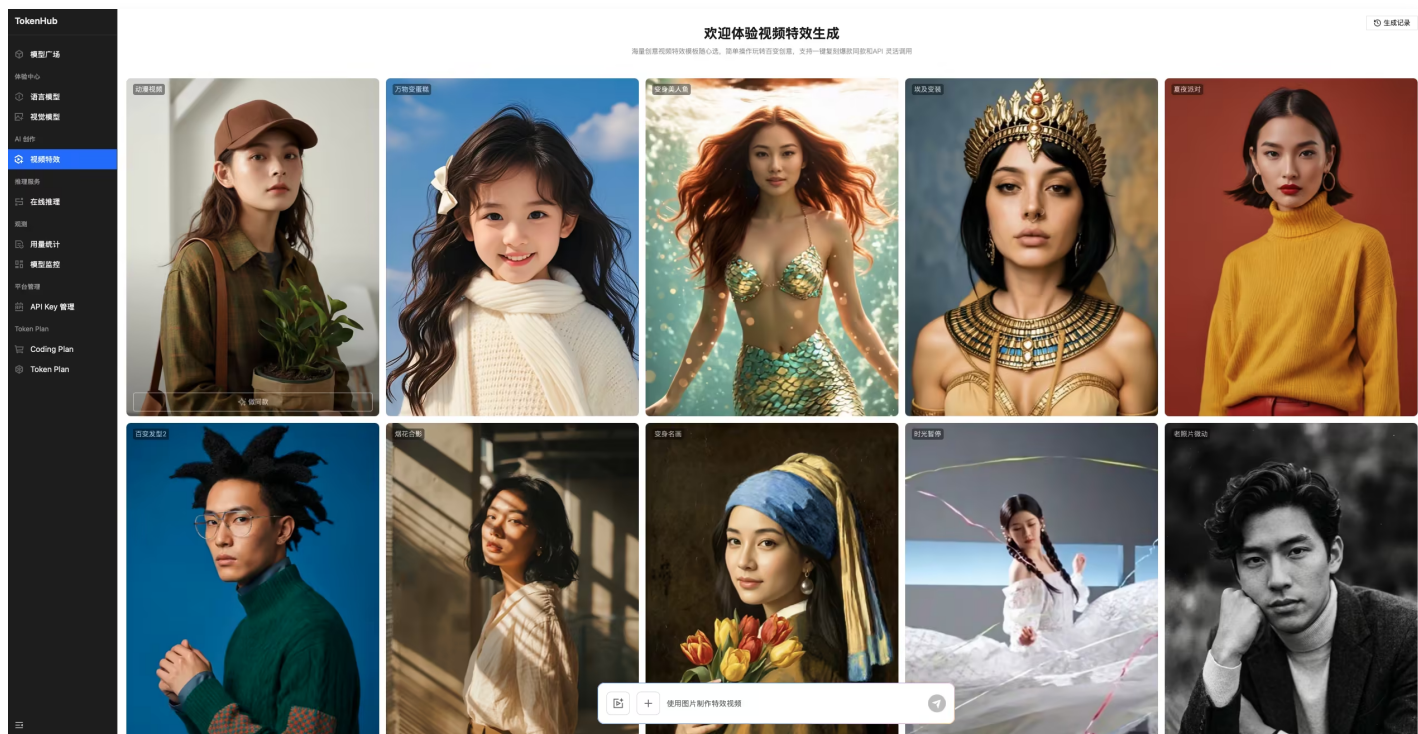


AI 创作

最近更新时间：2026-04-03 16:27:22

视频特效功能简介

视频特效 中心展示了海量创意视频特效模板，支持一键复刻爆款同款和 API 灵活调用，方便根据个人需求选择、体验和调用特效。



支持操作

控制台体验视频特效生成

1. 领取资源：进入 **模型广场** 页面，在右上角单击**新用户福利免费体验**，在弹窗中选择**视觉模型 > 生视频**后，单击**立即领取**，可领取50积分体验资源包。
2. 进入功能：进入 **视频特效** 页面，体验视频生成。
3. 选择模板：在主页 feed 流中选择模板，鼠标悬浮在目标模板上方后单击**做同款**，模板信息会自动填入页面下方对话框内；您也可以直接在页面下方对话框单击**特效模板**，选择特效模板填入对话框。
4. 替换参考图（可选）：单击**图片上传**，上传参考图，替换原有模板参考图。
5. 配置参数：设置**分辨率**、**背景音乐开关**。
6. 生成特效：在对话框右下角单击**发送图标**，等待输出结果。

API 文档调用

1. 进入 **视频特效** 页面，单击目标模板，进入模板详情页面。

2. 在模板详情页右下角，单击调用 API，跳转至目标 API 接口文档页面，您可参考文档进行调用。

配置项

配置项	说明
图片上传	支持上传单张参考图。
分辨率	支持选择不同分辨率，例如360p、540p、720p，请以实际模板支持为准。
背景音乐开关	支持设置背景音乐开关，对带有音效的模板生成 BGM。

模板详情

在视频特效页面，单击任一模板可查看详细信息，包括：

- 特效预览视频：所选特效的视频内容，支持播放。
- 模型：所选特效所使用的模型信息。
- 清晰度：所选特效的视频分辨率。
- 背景音乐：所选特效是否带有音频。

生成记录

生成记录以任务卡片的形式呈现，展示生成作品缩略图、使用模型、任务状态（已完成 / 生成中 / 生成失败）。

您可以在页面右上角单击生成记录，查看生成的作品，也可以对生成的结果执行如下操作：

控件	功能
重新生成（循环图标）	基于当前任务的特效模板，重新触发 AI 生成作品。
下载	下载该生成作品的原视频文件。
删除	删除该条生成记录及对应作品。

在线推理

最近更新时间：2026-04-03 16:27:22

功能概述

在线推理用于管理如何使用模型，例如免费额度用量、是否开启按 Token 计费，安全策略、限流策略等。与模型的关系参考下图：



服务类型

在线推理服务有两种类型：

1. 默认

平台会默认为所有支持的模型创建在线推理服务。用户可在 [模型广场](#)、模型详情页领取免费体验包，或在 [在线推理](#) 页面的服务列表中单击 [免费体验](#)，快速开始使用。

2. 自定义

当您需要自定义模型服务的计费策略，或希望创建多个服务以便按团队区分用量统计与权限管理时，可前往 [在线推理](#) 创建自定义推理服务。

自定义推理服务支持选择更丰富的计费方式，例如 TPM 保障包。未来，平台还将进一步支持在自定义服务中配置智能路由、限流规则、插件启停等能力，帮助您实现更灵活的服务管理与治理。

TokenHub

模型广场

体验中心

语言模型

视觉模型

AI 创作

视频特效

推理服务

在线推理

观测

用量统计

模型监控

平台管理

API Key 管理

Token Plan

Coding Plan

Token Plan

← 创建自定义推理服务

服务名称

选择模型 语言模型

HY 2.0 Think 深度思考 文本生成
擅长文创、多轮、指令遵循实用场景。数理、代码、Agent 能力显著增强。

HY 2.0 Instruct 文本生成
模型基座全面升级，基础能力更稳健。知识、数学、写作与推理达一线水准。指令遵循、多轮及长文理解表现优异。

Hunyuan-role 角色扮演 文本生成
针对角色扮演场景，人设高度贴合、极致的拟人语化。具有吸引力的剧情演绎和推进、情感陪伴与满足。

GLM-5 深度思考 文本生成
GLM-5是智谱新一代旗舰基座模型，面向Agentic Engineering打造，擅长复杂系统工程、长程Agent任务与编程，在Coding与Agent能力上取...

MiniMax-M2.7 文本生成
MiniMax 推出的自我进化大语言模型，具备卓越的软件工程能力和专业办公能力，支持复杂 Agent 交互与端到端项目交付。

kimi-k2.5 深度思考 文本生成 视觉理解
kimi k2.5 是 kimi 迄今最全能的模型，原生的多模态架构设计，同时支持视觉与文本输入、思考与非思考模式、对话与 Agent 任务。

计费方式 默认开启免费体验额度（剩余额度 500000 / 500000 tokens）赠送免费 Token 额度，零成本快速体验模型能力，额度用完后将按下方选择的计费方式继续计费

免费体验

赠送免费额度直接体验模型能力。试用结束后服务将暂停。

按 Token 计费

按实际消耗的 Token 数量计费，无最低消费，用多少付多...

TPM 保障

预留固定 TPM 吞吐额度，保障稳定调用；超出预留部分自动...

按算力单元计费

独享 GPU 算力单元，按实例时长计费。完全隔离，延迟最低。

计费信息 输入价格 3.18 - 4.605 元/百万tokens
输出价格 7.95 - 11.13 元/百万tokens

协议条款 我已阅读并同意 [《TokenHub 服务隐私协议政策》](#)、[《TokenHub 服务等级协议》](#) 和 [《TokenHub 服务条款》](#)

服务状态

每个在线推理服务都有状态，具体如下：

状态	说明
未启动	默认类型的推理服务，用户未使用前为未启动状态，开始免费体验后会变为运行中状态。
开通中	服务首次启用时，将会有个短暂的开通中状态，预计 5s 以内会变为运行中。
运行中	当前服务可正常访问。
已暂停	当账号欠费时，按量计费的服务会变为已暂停；当账号费用冲正后，服务会自动恢复到运行中。

已停止	当免费额度使用完后，用户未开启后付费时，或用户手动关闭了服务的后付费时，服务会变为 已停止 ，需要用户在 在线推理 页面手动开启后付费恢复服务。
-----	---

计费方式

计费方式标记当前服务的付费状态，具体含义如下：

状态	说明
免费体验	当前服务正在使用免费体验包，在免费体验包内的用量不计费。
按 Token 计费	当前服务已开通按 Token 使用量计费的后付费方式。
TPM 保障	当前服务已开通 TPM 保障，超出 TPM 限制的流量会按照 Token 计费。
无	当用户免费体验包用完后，未开启后付费时，将没有计费状态，服务会变为停止状态。

ⓘ 说明：

用户在免费体验包未使用完时也可提前打开后付费，打开后会同时展示：免费体验、按 Token 计费两种状态，平台会优先消耗免费体验包额度，免费体验包消耗完成后将开始按 Token 用量计费。

用量统计

最近更新时间：2026-04-03 16:27:22

功能概述

用量统计 页面帮助您全面监控 AI 资源消耗，支持从三个维度查看数据：

- 服务维度：查看特定在线推理服务的用量。
- 模型维度：查看每个模型的调用量、Token 消耗及免费额度。
- Key 维度：查看不同 API Key 的调用情况。

模型用量

按模型维度汇总调用数据，支持按模型类型分类查看。

分类筛选

页面顶部提供分类标签，快速筛选不同模型类型，同时支持按**在线推理服务**、**API Key** 筛选，查看特定服务的调用情况。

模型类型	包含模型
文本生成	DeepSeek V3、DeepSeek R1、HY-2.0、GLM-5、kimi-k2.5、MiniMax-M2.5 等
图片生成	HY-Image-Lite、HY-Image-V3.0
视频生成	HY-Video-1.5、YT-Video-2.0、YT-Video-FX、YT-Video-HumanActor
3D 生成	HY-3D-3.0、HY-3D-3.1、HY-3D-Express

调用指标

各模型在选定时间范围内的关键调用指标，统计粒度支持 1分钟 / 5分钟 / 1小时：

字段	模型类型	说明
总 Token 数	文本生成	输入 Token 数 + 输出 Token 数。
输入 Token 数		请求（Prompt）部分消耗的 Token 数量。
输出 Token 数		模型响应（Completion）部分消耗的 Token 数量。
每分钟总 Token 数		每分钟输入 Token 数 + 每分钟输出 Token 数。

每分钟输入 Token 数		每分钟输入侧的 Token 吞吐量 (tokens/min)。
每分钟输出 Token 数		每分钟输出侧的 Token 吞吐量 (tokens/min)。
调用次数		API 发起并执行的请求总次数，反映服务使用频次。
并发任务数峰值	<ul style="list-style-type: none"> • 图像生成 • 视频生成 • 3D 生成 	统计周期内，同一时间同时执行的任务数量的最大值，体现业务高峰期的并发压力。
并发任务数配额		平台允许用户同时执行的最大任务数量上限，为系统可控的并发限制阈值，超出将触发限流。
消耗积分数	<ul style="list-style-type: none"> • 视频生成 • 3D 生成 	调用接口过程中，根据模型、时长、并发等计费规则所扣除的积分总量，用于统计资源消耗成本。
时间	所有模型	请求时间区间。

用量趋势图

以可视化图表呈现调用趋势，每项指标均提供最大值、最小值、平均值三项统计摘要，帮助用户快速识别用量峰值与整体趋势。

文本生成

提供六项 Token 维度的趋势监控：

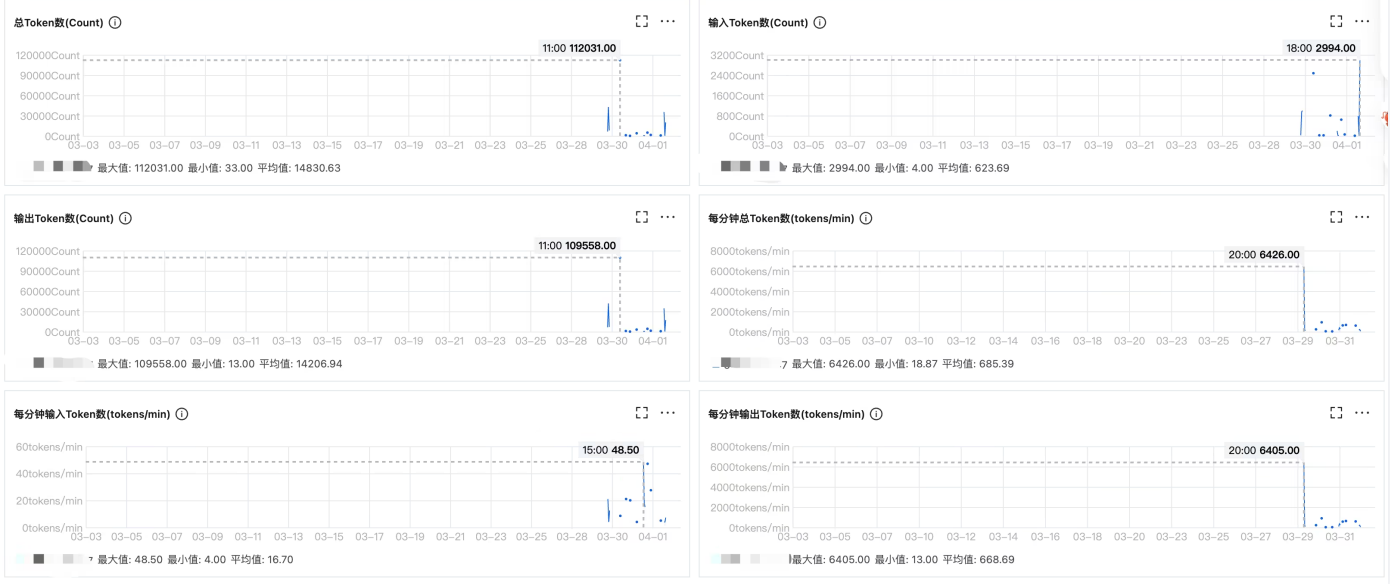
- Token 消耗趋势：总 Token 数 / 输入 Token 数 / 输出 Token 数随时间的变化走势
- Token 吞吐趋势：每分钟总 Token 数 / 每分钟输入 Token 数 / 每分钟输出 Token 数的并发量变化

用量统计

广州

文本生成 图像生成 视频生成 3D生成

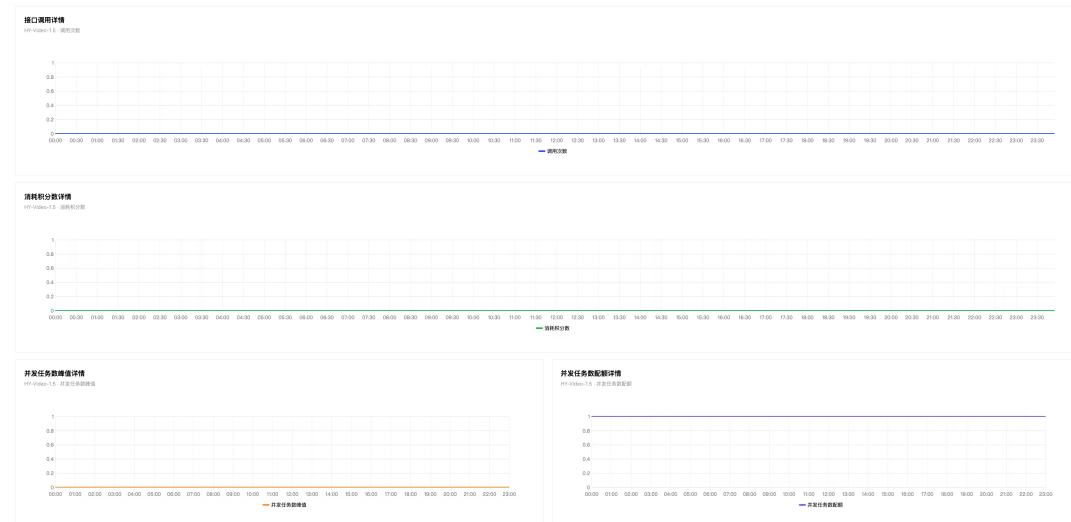
按服务 按模型 按Key 全部服务 1小时 今天 近3天 近7天 近30天 2026-03-03 12:15:40 ~ 2026-04-02 12:15:40



图像生成/视频生成/3D生成

提供四项调用维度的趋势监控：

图表	说明
接口调用详情	展示模型的调用次数随时间的变化趋势。
并发任务峰值详情	展示并发请求的峰值变化，用于评估突发流量。
并发任务配额详情	展示当前并发配额的使用情况，便于容量规划。
消耗积分数详情（仅视频生成、3D生成提供）	展示模型的积分消耗量随时间的变化趋势。



模型监控

最近更新时间：2026-04-03 16:27:22

功能概述

模型监控 提供实时的模型推理性能监控仪表盘，帮助您追踪各模型的请求并发、Token 吞吐、响应延迟等核心指标，及时发现性能瓶颈。

分类筛选

提供四个能力分类筛选：

分类	监控内容
文本生成	文本生成模型的 RPM（Requests Per Minute）、TTFT（Time To First Token）、TPOT（Time Per Output Token）等指标。
图像生成	图像生成模型的调用量、并发量等指标。
视频生成	视频生成模型的调用量、并发量等指标。
3D 生成	3D 生成模型的调用量、并发量等指标。

筛选条件

筛选项	说明
筛选维度	支持按服务/按模型筛选，切换不同的聚合视角。
服务选择	下拉选择特定推理服务（默认显示全部服务）。
时间范围	1小时/今天/近3天/近7天/近30天，或自定义时间区间。

监控指标说明

文本生成核心指标

以时序折线图的形式展示 3 个核心性能指标：

指标	全称	单位	说明
每分钟请求数（RPM）	Requests Per Minute	reqs/min	每分钟的请求并发数，反映当前的请求吞吐量。

首 Token 延迟 (TTFT)	Time To First Token	ms	从请求发送到收到第一个 Token 的响应时间。
每 Token 输出时延 (TPOT)	Time Per Output Token	ms	生成每个输出 Token 的平均耗时。

每项指标均提供最大值、最小值、平均值三项统计摘要，帮助用户快速定位性能异常。

示例如下：



图像/视频/3D 生成核心指标

图像、视频和 3D 生成模型的监控指标与文本生成不同，具体指标如下：

指标	说明
调用次数	每个时间点的调用请求数。
消耗积分数（仅视频生成、3D 生成提供）	每个时间点的积分消耗数。

示例如下：



API Key 管理

最近更新时间：2026-04-03 16:27:22

API Key 是调用 TokenHub 服务的身份凭证，长期有效。您可以在 [API Key 管理](#) 页面创建、编辑、启停和删除 API Key，并为每个 API Key 设置精细的访问范围。

⚠️ 注意：

API Key 长期有效，请在请求时通过请求头携带，并妥善保管，定期更新。请勿在公开环境中分享或泄露，以防未授权访问造成安全隐患或资金损失。

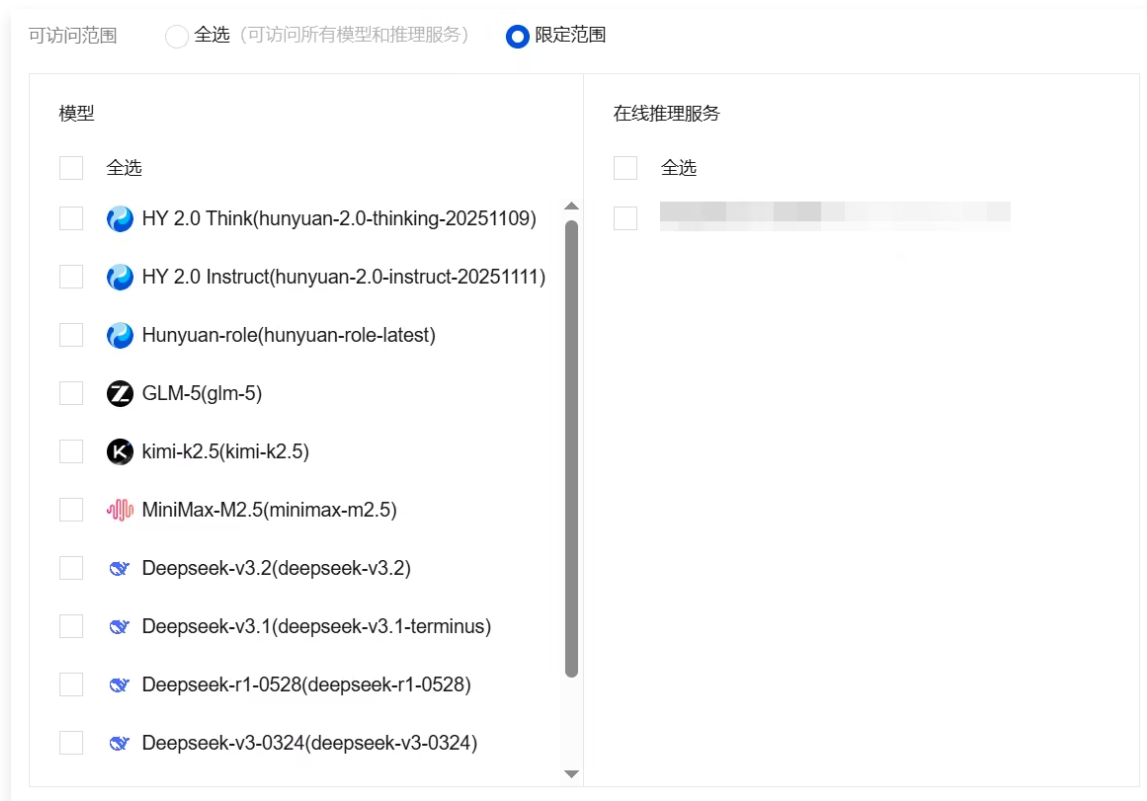
API Key 列表

字段	说明
名称	自定义的 API Key 名称。
说明	自定义的 API Key 说明，方便识别用途。
Key	脱敏显示（前 5 位 + *** + 后 4 位），支持一键复制完整 API Key。
状态	启用/禁用。 您可以通过状态开关切换 API Key 的启用、禁用状态。禁用状态下，被禁用 API Key 的所有 API 请求将被拒绝。
访问范围	<ul style="list-style-type: none">全部（可访问所有模型和服务）。限定范围（仅可访问勾选的模型/服务）。
创建时间	API Key 首次创建时间。
操作	<ul style="list-style-type: none">编辑：修改 API Key 的访问范围及关联的模型/服务。删除：永久删除 API Key，需二次确认。删除后使用该 API Key 的所有请求将立即失败。

创建 API Key

1. 进入 [API Key 管理](#) 页面。
2. 在页面上方选择地域后，单击 **创建 API Key**。
3. 在 **创建 API Key** 对话框进行配置：
 - **Key 名称**：例如，"生产环境-文本服务"、"测试用 Key"。
 - **说明（选填）**：为 API Key 设置说明，方便识别用途，区分 API Key 的用途。
 - **可访问范围**：

- **全选**：可访问您账号下的所有模型和推理服务。
- **限定范围**：您可以在**可访问范围**区域左、右两侧勾选可访问的模型和/或推理服务，实现访问范围控制。



4. 单击**确定**，完成创建。

创建完成后，请您务必**复制并妥善保管 API Key**，在后续调用 API 流程中将会使用该信息。

注意：

API Key 名称和 API Key 值创建后不可修改。

编辑 API Key

1. 进入 [API Key 管理](#) 页面。
2. 在页面上方选择目标 API Key 所属地域，找到 API Key。
3. 在 API Key 右侧的**操作列**单击**编辑**。
4. 您可以在**编辑 API Key**对话框修改 API Key 说明或 API Key 的访问范围。

注意：

切换为新的指定范围后，上次选择的访问范围将失效，以本次切换后的范围为准。

5. 单击**确定**，保存更改。

删除 API Key

注意：

删除 API Key 后将无法恢复，使用该 API Key 的所有服务调用将立即失效。删除时，会触发账号身份校验，请知悉。

1. 进入 [API Key 管理](#) 页面。
2. 在页面上方选择目标 API Key 所属地域，找到 API Key。
3. 在 API Key 右侧的**操作列**单击**删除**。
4. 在**确认删除 API Key**对话框确认删除信息无误后，单击**确认**，完成删除。



使用建议

建议	说明
按用途分配 API Key	为不同应用或环境（开发/测试/生产）创建独立的 API Key。
最小权限原则	使用 指定范围 限制 API Key 仅能访问所需的模型和服务。
及时轮换 API Key	定期更换 API Key，降低泄露风险。
禁用不用的 API Key	不再使用的 API Key 应及时禁用或删除。
妥善保管 API Key	API Key 创建后，请立即保存到安全位置。

Coding Plan

Coding Plan 概述

最近更新时间：2026-04-03 21:02:21

腾讯云 Coding Plan，是为 AI Coding 场景推出的专属订阅套餐。

- 支持模型：选用 Tencent HY 2.0 Instruct、GLM-5、Kimi-K2.5、MiniMax-M2.5 等多个模型。
- 适配工具：支持使用 OpenClaw、CodeBuddy、Claude Code、Cline、Cursor 等主流编程工具。

适用场景

个人开发场景：助力开发者完成个人项目、学习实践、工具搭建等编码任务。

- 说明：**
企业级开发需求，请通过 [文本生成服务](#) 调用模型服务。

套餐介绍

访问 [Coding Plan 购买页](#)，根据实际需求选择并购买套餐。

套餐详情

套餐内容	Lite 套餐	Pro 套餐
原价	40元/月	200元/月
用量限制	<ul style="list-style-type: none">每 5 小时：最多约 1,200 次请求每周：最多约 9,000 次请求每订阅月：最多约 18,000 次请求	<ul style="list-style-type: none">每 5 小时：最多约 6,000 次请求每周：最多约 45,000 次请求每订阅月：最多约 90,000 次请求
新客首购特惠活动	<ul style="list-style-type: none">活动时间：2026.03.05 16:00 – 2026.04.19 00:00 (UTC+8)发放规则：为保障订阅用户在调用高峰期的使用体验，特惠活动的参与名额每日限量，先到先得，售完即止购买资格：仅限从未订阅过任何腾讯云大模型 Coding Plan 套餐的腾讯云新用户参与，同一用户仅可享受一次优惠	
	Lite 套餐特惠价 <ul style="list-style-type: none">首月7.9元/月在活动时间内，购买特惠套餐并开启自动续费次月20元/月	Pro 套餐特惠价 <ul style="list-style-type: none">首月39.9元/月在活动时间内，购买特惠套餐并开启自动续费次月100元/月

● 第三月起恢复原价40元/月

● 第三月起恢复原价200元/月

● 额度消耗：

- 请求次数为模型调用的预估数值。通常一次用户提问会触发多次模型调用，且每次调用均会消耗 1 次用量额度，因此实际消耗的请求次数一般多于用户提问次数。
- 简单任务单次提问约消耗 5~15 次模型调用，复杂任务单次提问约消耗 15~30次或更多模型调用。用量信息可在 [Coding Plan 页面](#) 查看。

订阅前须知

Coding Plan 不支持退款。因此在订阅前请知悉以下重要内容：

- 1. 订阅账号规范：**为订阅人专享使用，**严禁账号共享**。若存在账号共享行为，可能导致**订阅权益受限**，腾讯不承担赔偿责任，敬请知悉。
- 2. 严禁 API 调用：**仅限在编程工具（例如：Claude Code、CodeBuddy Code、OpenClaw 等）中使用，禁止以 API 调用的形式用于自动化脚本、自定义应用程序后端或任何非交互式批量调用场景。将套餐 API Key 用于允许范围之外的调用将被视为违规或滥用，可能会导致**订阅被暂停或 API Key 被封禁**。对于违规而导致服务受损，腾讯不承担赔偿责任。

使用说明

获取专属 API Key 和 Base URL

如果您已经购买了套餐，您需要获取并在 AI 工具中配置套餐专属的 API Key 和 Base URL，才能正确使用并抵扣套餐额度。

- **API Key:** 在 [Coding Plan 页面](#)，获取 Coding Plan 专属 API Key（格式为 `sk-sp-xxxx`）。
- **Base URL:** 后续需在 AI 工具中配置以下其中一个 Base URL（因工具而异），具体操作请参见对应的 AI 工具文档。
 - **OpenAI 兼容协议：** `https://api.lkeap.cloud.tencent.com/coding/v3`。
 - **Anthropic 兼容协议：** `https://api.lkeap.cloud.tencent.com/coding/anthropic`。

说明：

Coding Plan 专属的 API Key 和 Base URL 与腾讯云预付费及后付费的 API Key（`sk-xxxx`）和 Base URL（`https://api.lkeap.cloud.tencent.com/xxxxx/`）不互通，请勿混用。

Pro 套餐

套餐状态：生效中

计费方式：200元/月 (按月计费)

有效期至：2026-03-06 至 2026-05-05

剩余天数：34天 立即续费

自动续费：

每5小时 (总量6000)

2026-04-02 11:20:58 刷新

0.0%

0% 100%

每周 (总量45000)

2026-04-02 11:20:58 刷新

0.0%

0% 100%

每订阅月 (总量90000)

2026-04-02 11:20:58 刷新

0.1%

0% 100%

API Key

CodingPlan APIKey (SK-SP-) 与混元大模型 APIKey (SK-) 单独计费使用。请妥善保存API KEY，请勿通过任何方式上传或公开分享您的密钥信息。

sk-sp***E9YQ 重置 复制

套餐支持的模型

通过配置 Model Name，切换模型。

选择 Auto 模型会通过算法自动匹配最优模型，请通过 model 参数值配置。

说明：

- kimi-k2.5 当前资源负载较高，高峰时段可能触发请求限频机制，为保障使用体验，请优先选用其他模型。
- Coding Plan 可用模型暂不支持多模态能力。

模型	model 参数值
Auto	tc-code-latest
Tencent HY 2.0 Instruct	hunyuan-2.0-instruct
Tencent HY 2.0 Think	hunyuan-2.0-thinking
MiniMax-M2.5	minimax-m2.5 (兼容别名 minimax-m-2-5)
Kimi-K2.5	kimi-k2.5 (兼容别名 kimi-k-2-5)
GLM-5	glm-5 (兼容别名 glm-5-0)
Hunyuan-T1 (即将下线)	hunyuan-t1
Hunyuan-TurboS (即将下线)	hunyuan-turbos

支持的 AI 工具

Coding Plan 支持主流的 AI 工具，可按需选择。

AI 工具		
OpenClaw 开源、自托管个人 AI 助手	CodeBuddy Code 腾讯云 AI 技术的智能编程工具	Claude Code AI 终端编程助手，支持自然语言编程
OpenCode 开源 AI 编程代码工具	Cline VSCode 扩展，代码补全和调试	Cursor AI 原生代码编辑器
Codex	Kilo CLI	Kilo Code

OpenAI 推出的命令行编程工具

轻量高性能命令行编程工具

高效编程助手

Coding Plan 接入 AI 工具 OpenClaw

最近更新时间：2026-04-03 16:27:22

OpenClaw 是一款开源的本地 AI 代理框架，能让 AI 从“回答问题”进化为“动手执行任务”。它支持在 Windows、macOS 和 Linux 上自主运行，通过内置工具和可扩展的插件体系，自动完成文件整理、邮件处理、代码编写等操作。

腾讯云 Coding Plan 支持在 OpenClaw 中使用，可参考本文进行配置及使用。

前置条件

在开始配置前，请确保您已订阅 Coding Plan 套餐并获取 API Key，可参见 [Coding Plan 概述](#)。

安装 OpenClaw

您需要根据实际环境类型，执行对应命令安装 OpenClaw。

macOS/Linux

```
curl -fsSL https://openclaw.ai/install.sh | bash
```

Windows

在 PowerShell 中执行以下命令。

```
iwr -useb https://openclaw.ai/install.ps1 | iex
```

根据提示信息完成 OpenClaw 配置，参考配置如下。

向导配置项	配置内容
I understand this is powerful and inherently risky. Continue?	选择 Yes 。
Onboarding mode	选择 QuickStart 。

Model/auth provider	选择 Skip for now ，后续可以配置。
Filter models by provider	选择 All providers 。
Default model	使用默认配置。
Select channel (QuickStart)	选择 Skip for now ，后续可以配置。
Configure skills now? (recommended)	选择 No ，后续可以配置。
Enable hooks?	按空格键选中选项，按回车键进入下一步。
How do you want to hatch your bot?	选择 Hatch in TUI 。

配置 Coding Plan

⚠ 注意：

如果出现配置正常但是访问404，401等问题是因为 OpenClaw 的缓存问题导致，您可以直接修改本地配置文件，重启 OpenClaw gateway，配置文件路径如下：

- Windows: `C:\Users\<用户名>\.openclaw\openclaw.json`
- macOS/Linux: `~/openclaw/openclaw.json`

1. 执行以下命令打开 OpenClaw 的 WebUI，然后在 WebUI 的左侧菜单栏中选择 **Config > Raw**。

```
openclaw dashboard
```

2. 在 `openclaw.json` 配置项中增加 `models.providers` 的配置。

- `baseUrl`: 参考样例，使用 Coding Plan 兼容 OpenAI 接口协议的 Base URL (`https://api.lkeap.cloud.tencent.com/coding/v3`) 配置。
- `<USER_API_KEY>`: 替换成您自己的 API Key。

```
"models": {
  "mode": "merge",
  "providers": {
    "tencent-coding-plan": {
      "baseUrl": "https://api.lkeap.cloud.tencent.com/coding/v3",
      "apiKey": "<USER_API_KEY>",
      "api": "openai-completions",
      "models": [
        {
          "id": "tc-code-latest",
```

```

    "name": "Auto",
    "reasoning": false,
    "input": ["text"],
    "cost": { "input": 0, "output": 0, "cacheRead": 0,
"cacheWrite": 0 },
    "contextWindow": 196608,
    "maxTokens": 32768
  },
  {
    "id": "hunyuan-2.0-instruct",
    "name": "Tencent HY 2.0 Instruct",
    "reasoning": false,
    "input": ["text"],
    "cost": { "input": 0, "output": 0, "cacheRead": 0,
"cacheWrite": 0 },
    "contextWindow": 128000,
    "maxTokens": 16000
  },
  {
    "id": "hunyuan-2.0-thinking",
    "name": "Tencent HY 2.0 Think",
    "reasoning": false,
    "input": ["text"],
    "cost": { "input": 0, "output": 0, "cacheRead": 0,
"cacheWrite": 0 },
    "contextWindow": 128000,
    "maxTokens": 32000
  },
  {
    "id": "hunyuan-t1",
    "name": "Hunyuan-T1",
    "reasoning": false,
    "input": ["text"],
    "cost": { "input": 0, "output": 0, "cacheRead": 0,
"cacheWrite": 0 },
    "contextWindow": 64000,
    "maxTokens": 32000
  },
  {
    "id": "hunyuan-turbos",

```

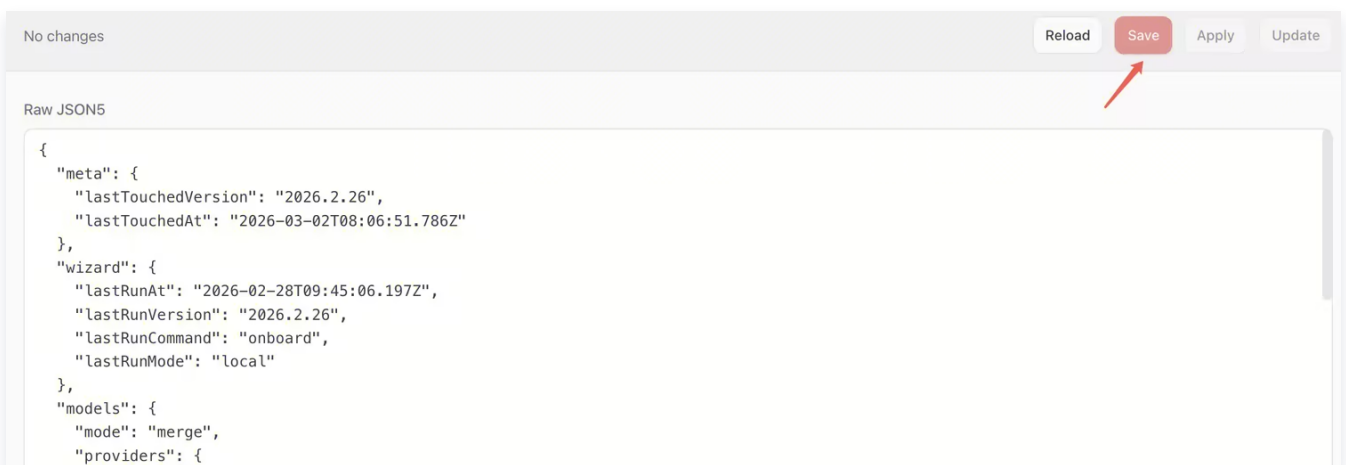
```
    "name": "hunyuan-turbos",
    "reasoning": false,
    "input": ["text"],
    "cost": { "input": 0, "output": 0, "cacheRead": 0,
"cacheWrite": 0 },
    "contextWindow": 32000,
    "maxTokens": 16000
  },
  {
    "id": "minimax-m2.5",
    "name": "MiniMax-M2.5",
    "reasoning": false,
    "input": ["text"],
    "cost": { "input": 0, "output": 0, "cacheRead": 0,
"cacheWrite": 0 },
    "contextWindow": 196608,
    "maxTokens": 32768
  },
  {
    "id": "kimi-k2.5",
    "name": "Kimi-K2.5",
    "reasoning": false,
    "input": ["text"],
    "cost": { "input": 0, "output": 0, "cacheRead": 0,
"cacheWrite": 0 },
    "contextWindow": 262144,
    "maxTokens": 32768
  },
  {
    "id": "glm-5",
    "name": "GLM-5",
    "reasoning": false,
    "input": ["text"],
    "cost": { "input": 0, "output": 0, "cacheRead": 0,
"cacheWrite": 0 },
    "contextWindow": 202752,
    "maxTokens": 16384
  }
]
```

```
}  
}
```

3. 在 `openclaw.json` 配置项中修改 `agents.default`，增加模型列表信息，其中：**primary** 字段指定的模型可以根据您的偏好进行配置。

```
"agents": {  
  "defaults": {  
    "model": {  
      "primary": "tencent-coding-plan/glm-5"  
    },  
  },  
  "models": {  
    "tencent-coding-plan/tc-code-latest": {},  
    "tencent-coding-plan/hunyuan-2.0-instruct": {},  
    "tencent-coding-plan/hunyuan-2.0-thinking": {},  
    "tencent-coding-plan/hunyuan-t1": {},  
    "tencent-coding-plan/hunyuan-turbos": {},  
    "tencent-coding-plan/minimax-m2.5": {},  
    "tencent-coding-plan/kimi-k2.5": {},  
    "tencent-coding-plan/glm-5": {}  
  }  
}
```

4. 在界面上单击 **Save** 保存配置，保存完成后，单击 **Update** 使配置生效。

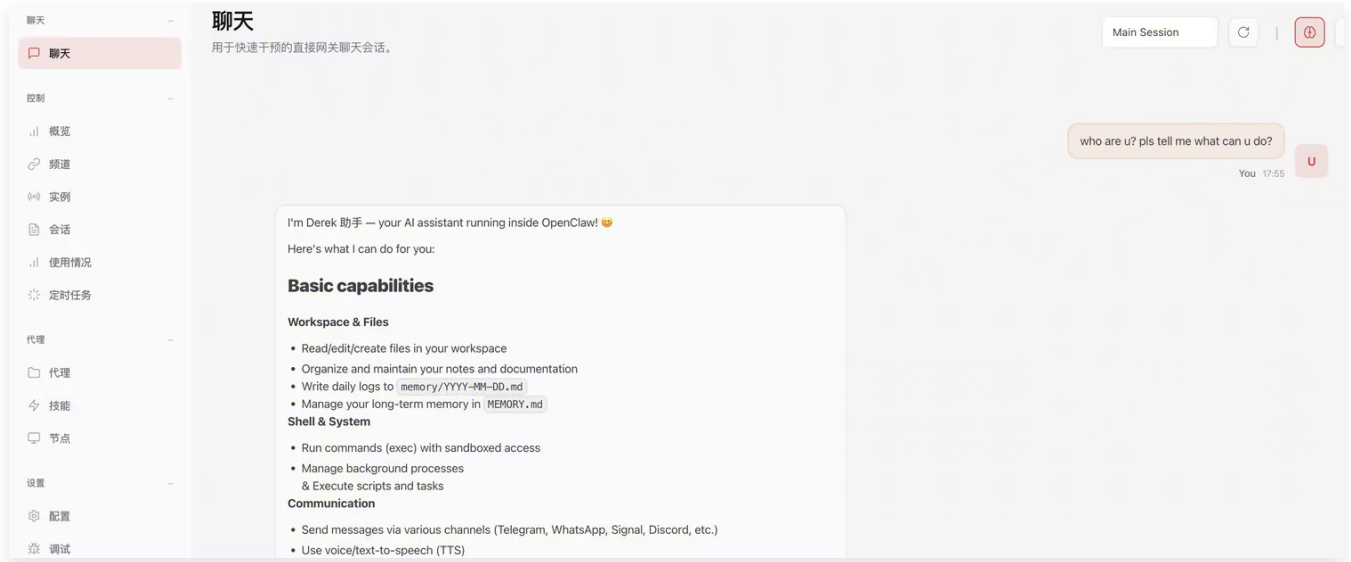


使用 OpenClaw

OpenClaw 支持通过 WebUI 和 TUI 的形式进行交互使用。

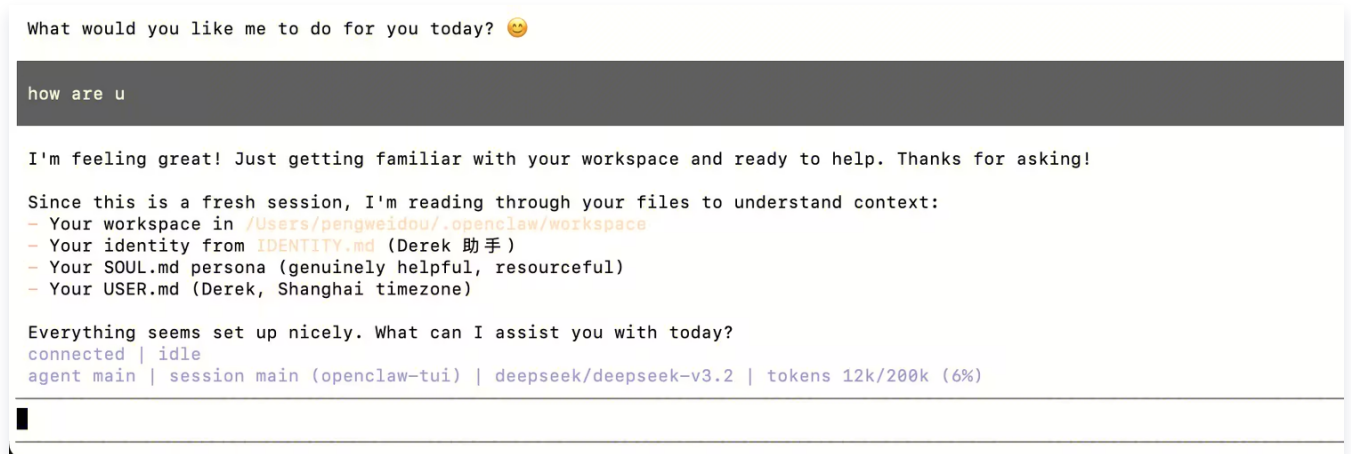
• WebUI

您可以通过 `openclaw dashboard` 命令打开 WebUI，单击 WebUI 左侧的聊天，即可进行交互。



• TUI

您可以通过 `openclaw tui` 命令打开 TUI 进行交互。



CodeBuddy Code

最近更新时间：2026-04-03 16:27:22

CodeBuddy Code 是基于腾讯云 AI 技术的智能编程工具，深度集成腾讯云生态，提供从代码编写到项目部署的全链路 AI 辅助。本文介绍如何在 CodeBuddy Code 中配置与使用腾讯云大模型 Coding Plan。

前置条件

在开始配置前，请确保您已订阅 Coding Plan 套餐并获取 API Key，可参见 [Coding Plan 概述](#)。

安装 CodeBuddy Code

请先确保本地已安装 Node.js 18.0 及以上版本，再执行以下命令安装 CodeBuddy Code。

```
npm install -g @tencent-ai/codebuddy-code
```

安装完成后，您可以执行如下命令检查，输出版本号则表示安装成功。

```
codebuddy --version
```

配置 Coding Plan

1. 找到 CodeBuddy Code 的配置文件，配置文件路径如下，配置文件不存在时新建即可。

- Windows: `C:\Users\<<用户名>\.codebuddy\models.json`。
- macOS/Linux: `~/.codebuddy/models.json`。

2. 复制粘贴如下配置，并保存。

ⓘ 说明：

请您将代码中的 `<USER_API_KEY>` 替换为 Coding Plan 专属的 API Key。

```
{
  "models": [
    {
      "id": "tc-code-latest",
      "name": "Auto",
      "vendor": "Tencent Cloud",
      "apiKey": "<USER_API_KEY>",
      "url": "https://api.lkeap.cloud.tencent.com/coding/v3"
    }
  ]
}
```

```
},
{
  "id": "hunyuan-2.0-instruct",
  "name": "Tencent HY 2.0 Instruct",
  "vendor": "Tencent Cloud",
  "apiKey": "<USER_API_KEY>",
  "url": "https://api.lkeap.cloud.tencent.com/coding/v3"
},
{
  "id": "hunyuan-2.0-thinking",
  "name": "Tencent HY 2.0 Think",
  "vendor": "Tencent Cloud",
  "apiKey": "<USER_API_KEY>",
  "url": "https://api.lkeap.cloud.tencent.com/coding/v3"
},
{
  "id": "hunyuan-t1",
  "name": "Hunyuan-T1",
  "vendor": "Tencent Cloud",
  "apiKey": "<USER_API_KEY>",
  "url": "https://api.lkeap.cloud.tencent.com/coding/v3"
},
{
  "id": "hunyuan-turbos",
  "name": "Hunyuan-TurboS",
  "vendor": "Tencent Cloud",
  "apiKey": "<USER_API_KEY>",
  "url": "https://api.lkeap.cloud.tencent.com/coding/v3"
},
{
  "id": "minimax-m2.5",
  "name": "MiniMax-M2.5",
  "vendor": "Tencent Cloud",
  "apiKey": "<USER_API_KEY>",
  "url": "https://api.lkeap.cloud.tencent.com/coding/v3"
},
{
  "id": "kimi-k2.5",
  "name": "Kimi-K2.5",
  "vendor": "Tencent Cloud",
```

```
"apiKey": "<USER_API_KEY>",
"url": "https://api.lkeap.cloud.tencent.com/coding/v3"
},
{
  "id": "glm-5",
  "name": "GLM-5",
  "vendor": "Tencent Cloud",
  "apiKey": "<USER_API_KEY>",
  "url": "https://api.lkeap.cloud.tencent.com/coding/v3"
}
],
"availableModels": ["tc-code-latest", "hunyuan-2.0-instruct", "hunyuan-2.0-thinking", "hunyuan-t1", "hunyuan-turbos", "minimax-m2.5", "kimi-k2.5", "glm-5"]
}
```

开始使用

配置完成后执行如下命令启动 CodeBuddy Code。

```
codebuddy
```

首次使用 CodeBuddy Code 时，您需要完成登录认证。启动后会显示登录方式选择界面：

```
Select login method:
> Log in via Chinese Site
  Log in via International Site
  Log in via Enterprise Domain
  Log in via iOA (Tencent only)
```

登录方式说明如下：

登录方式	适用场景	说明
Chinese Site	中国站用户	通过腾讯云中国站 (copilot.tencent.com) 进行认证，支持境内主流模型。
International Site	国际站用户	通过腾讯云国际站 (codebuddy.ai) 进行认证，支持境外主流模型。

Enterprise Domain	专享版/私有化部署	连接企业专享版或自建的 CodeBuddy 服务，需要输入企业提供的服务地址。
iOA	腾讯内部员工	通过腾讯 iOA 零信任系统进行认证，仅限腾讯内部员工使用。

使用 ↑↓ 键选择登录方式，按 Enter 确认后会自动打开浏览器完成认证。

输入 /model ，选择模型，下图以 hunyuan-2.0-instruct 为例。

```

CodeBuddy Code v2.54.0

Tips for getting started
Run /init to create a CODEBUDDY.md or AGENTS.md file with instruction...
Press / to use commands, @ to mention files.
Press Esc twice to reset the input box.

Recent activity
你好
/model

Tencent HY 2.0 Instruct · internal Usage Billing

> /model
  Switch model to hunyuan-2.0-instruct

> 你好

● 你好！我是 CodeBuddy Code，一个交互式 CLI 工具，可以帮你处理各种软件工程任务。

我注意到这是一个新会话。如果你有任何编程相关的问题或需要帮助完成任务，请告诉我具体需求。

如果需要帮助，可以使用 /help 查看使用说明，或者访问 https://cnb.cool/codebuddy/codebuddy-code/-/issues 提供反馈。

> /model

Select Model
Select a model for this conversation and future sessions.

> Tencent HY 2.0 Instruct [hunyuan-2.0-instruct] ✓
Enter to confirm · t for this session only · Esc to exit
    
```

OpenCode

最近更新时间：2026-04-03 16:27:22

OpenCode 是一款开源的 AI 编程代理工具，提供终端界面、桌面应用和 IDE 扩展等多种使用方式。本文介绍如何在 OpenCode 中配置与使用腾讯云大模型 Coding Plan。

前置条件

在开始配置前，请确保您已订阅 Coding Plan 套餐并获取 API Key，可参见 [Coding Plan 概述](#)。

安装 OpenCode

执行如下命令安装 OpenCode，更多操作可参考 [OpenCode 官方配置指引](#)。

```
npm install -g opencode-ai
```

安装完成后，您可以执行如下命令检查，输出版本号则表示安装成功。

```
opencode -v
```

配置 Coding Plan

您需要按照下面的示例修改 OpenCode 的配置文件，配置文件路径如下，配置文件不存在时新建即可。

- macOS / Linux: `~/.config/opencode/opencode.json`。
- Windows: `C:\Users\<用户名>\.config\opencode\opencode.json`。

ⓘ 说明：

请您将 `<USER_API_KEY>` 替换为 Coding Plan 专属的 API Key。

```
{
  "$schema": "https://opencode.ai/config.json",
  "provider": {
    "myprovider": {
      "npm": "@ai-sdk/openai-compatible",
      "name": "lkeap",
      "options": {
        "baseUrl": "https://api.lkeap.cloud.tencent.com/coding/v3",
        "apiKey": "<USER_API_KEY>"
      }
    }
  }
}
```

```
"models": {
  "tc-code-latest": {
    "name": "Auto",
    "modalities": {
      "input": [
        "text"
      ],
      "output": [
        "text"
      ]
    },
    "options": {
      "thinking": {
        "type": "enabled",
        "budgetTokens": 8192
      }
    },
    "limit": {
      "context": 196608,
      "output": 32768
    }
  },
  "hunyuan-2.0-instruct": {
    "name": "Tencent HY 2.0 Instruct",
    "modalities": {
      "input": [
        "text"
      ],
      "output": [
        "text"
      ]
    },
    "options": {
      "thinking": {
        "type": "enabled",
        "budgetTokens": 8192
      }
    },
    "limit": {
      "context": 128000,
```

```
        "output": 16000
      }
    },
    "hunyuan-2.0-thinking": {
      "name": "Tencent HY 2.0 Think",
      "modalities": {
        "input": [
          "text"
        ],
        "output": [
          "text"
        ]
      },
      "options": {
        "thinking": {
          "type": "enabled",
          "budgetTokens": 8192
        }
      },
      "limit": {
        "context": 128000,
        "output": 32000
      }
    },
    "hunyuan-t1": {
      "name": "Hunyuan-T1",
      "modalities": {
        "input": [
          "text"
        ],
        "output": [
          "text"
        ]
      },
      "options": {
        "thinking": {
          "type": "enabled",
          "budgetTokens": 8192
        }
      },
    },
```

```
"limit": {
  "context": 64000,
  "output": 32000
},
"hunyuan-turbos": {
  "name": "hunyuan-turbos",
  "modalities": {
    "input": [
      "text"
    ],
    "output": [
      "text"
    ]
  },
  "options": {
    "thinking": {
      "type": "enabled",
      "budgetTokens": 8192
    }
  },
  "limit": {
    "context": 32000,
    "output": 16000
  }
},
"minimax-m2.5": {
  "name": "MiniMax-M2.5",
  "modalities": {
    "input": [
      "text"
    ],
    "output": [
      "text"
    ]
  },
  "options": {
    "thinking": {
      "type": "enabled",
      "budgetTokens": 8192
    }
  }
}
```

```
    }
  },
  "limit": {
    "context": 196608,
    "output": 32768
  }
},
"kimi-k2.5": {
  "name": "Kimi-K2.5",
  "modalities": {
    "input": [
      "text"
    ],
    "output": [
      "text"
    ]
  },
  "options": {
    "thinking": {
      "type": "enabled",
      "budgetTokens": 8192
    }
  },
  "limit": {
    "context": 262144,
    "output": 32768
  }
},
"glm-5": {
  "name": "GLM-5",
  "modalities": {
    "input": [
      "text"
    ],
    "output": [
      "text"
    ]
  },
  "options": {
    "thinking": {
```

```
        "type": "enabled",
        "budgetTokens": 8192
    }
},
"limit": {
    "context": 202752,
    "output": 16384
}
}
}
}
}
```

使用 OpenCode

执行如下命令启动 OpenCode:

```
opencode
```

输入 `/models`，选择配置的模型并在 OpenCode 中使用。



完成上述配置后，即可开始使用 OpenCode。

你好

Thinking: 用户用中文说"你好", 这是一个简单的问候。
根据我的指令, 我应该简洁地回应, 不要有多余的开场白或结束语。
我应该直接回应问候。

由于用户用中文打招呼, 我也应该用中文回应。
但根据指令, 我应该保持简洁, 不超过4行文字 (不包括工具使用)。

对于简单的问候, 我不需要使用任何工具, 只需要简单地回应即可。

你好! 有什么我可以帮你的吗?

■ Build · glm-5 · 6.2s

Greeting in Chinese conversation

Context

10,625 tokens

0% used

\$0.00 spent

LSP

LSPs will activate as files are read

█

Build GLM-5 lkeap

/~

tab agents ctrl+p commands

• OpenCode 1.2.15

Claude Code

最近更新时间：2026-04-03 16:27:22

Claude Code 是一款开源的 AI 编程助手，支持通过 Anthropic API 接入模型，帮助开发者快速编写、调试和管理代码。

Coding Plan 中的模型支持 Anthropic API 兼容接口，可以通过 Claude Code 调用。

前置条件

在开始配置前，请确保您已订阅 Coding Plan 套餐并获取 API Key，可参见 [Coding Plan 概述](#)。

安装 Claude Code

⚠ 注意：

在 Windows 上使用 Claude Code，需要提前安装 [WSL](#) 或 [Git for Windows](#)。

执行如下命令安装 Claude Code，更多操作可参考 [Claude Code 安装指引](#)。

```
npm install -g @anthropic-ai/claude-code
```

安装结束后，执行以下命令查看安装结果，若显示版本号则安装成功。

```
claude --version
```

配置 Coding Plan

在 Claude Code 中接入 Coding Plan，需要配置以下信息：

- ANTHROPIC_BASE_URL：设置为 `https://api.lkeap.cloud.tencent.com/coding/anthropic`。
- ANTHROPIC_AUTH_TOKEN：设置为 Coding Plan 专属 API Key。
- ANTHROPIC_MODEL：设置为 Coding Plan 支持的模型。

配置步骤如下：

1. 编辑或新增 settings.json 文件，需要修改或添加的配置信息如下：

- <USER_API_KEY>：替换为您自己的 API Key。
- <Model_Name>：更新为需要使用的模型信息，例如 glm-5。模型信息可参见 [Coding Plan 支持的 Model Name](#)。

⚠ 注意：

不同系统配置文件路径不同，具体如下：

- MacOS / Linux: `~/.claude/settings.json`。
- Windows: `C:\Users\<<用户名>\.claude\settings.json`。

```
{
  "env": {
    "ANTHROPIC_AUTH_TOKEN": "<USER_API_KEY>",
    "ANTHROPIC_BASE_URL":
      "https://api.lkeap.cloud.tencent.com/coding/anthropic",
    "ANTHROPIC_MODEL": "<Model_Name>"
  }
}
```

2. 编辑或新增 `.claude.json` 文件，修改或新增 `hasCompletedOnboarding` 字段值为 `true`。

⚠ 注意：

不同系统配置文件路径不同，具体如下：

- MacOS / Linux: `~/.claude.json`
- Windows: `C:\Users\<<用户名>\.claude.json`

```
{
  "hasCompletedOnboarding": true
}
```

3. 保存配置文件后，在新的终端窗口执行后续命令。

在命令行中使用 Claude Code

通过命令行进入项目目录后，执行 `claude` 命令启动 Claude Code 即可开始使用。

```
cd my-project
claude
```

选择 **Yes, I trust this folder** 信任当前目录。

```
Accessing workspace:
/Users/pengweidou/CodeProjects/atomic

Quick safety check: Is this a project you created or one you trust? (Like your
own code, a well-known open source project, or work from your team). If not,
take a moment to review what's in this folder first.

Claude Code'll be able to read, edit, and execute files here.

Security guide
> 1. Yes, I trust this folder
   2. No, exit

Enter to confirm · Esc to cancel
```

```
Claude Code v2.1.63

Welcome back!



glm-5 · API Usage Billing
~/CodeProjects/atomic

Tips for getting started
Run /init to create a CLAUDE.md file with i...

Recent activity
No recent activity

/model to try Opus 4.6

> try "refactor <filepath>"

? for short... Claude Code has switched from npm to native installer. Run `c...
Auto-updating...
```

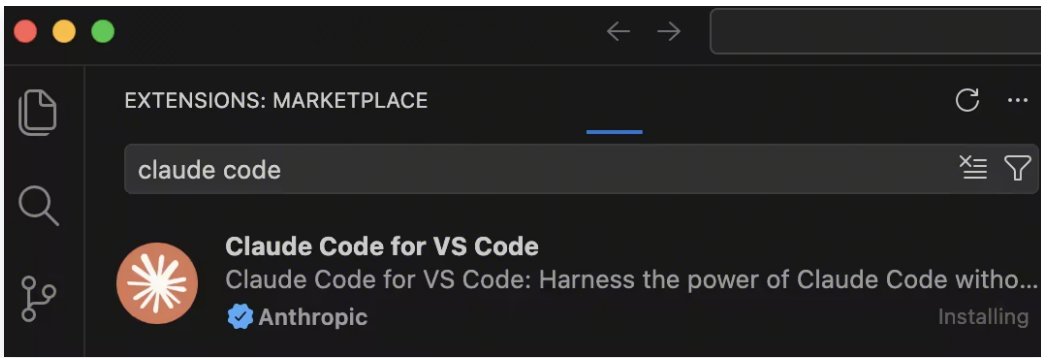
在 IDE 中使用 Claude Code

⚠ 注意:

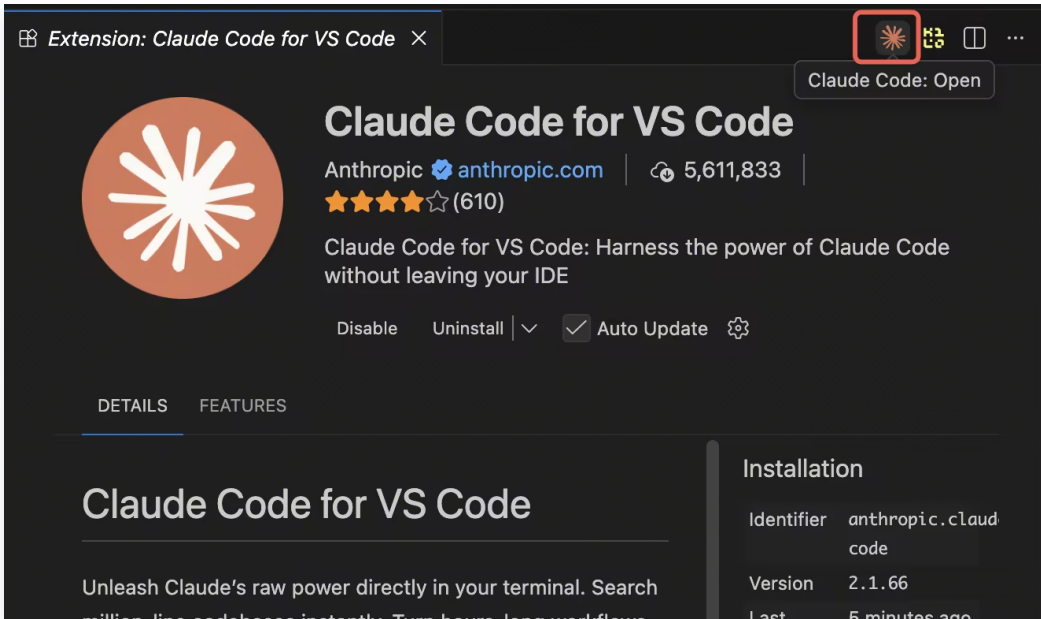
Claude Code IDE 插件依赖 Claude Code CLI 工具，需先完成 Claude Code 的安装及配置。

VS Code

1. 在扩展市场搜索 Claude Code 进行安装。



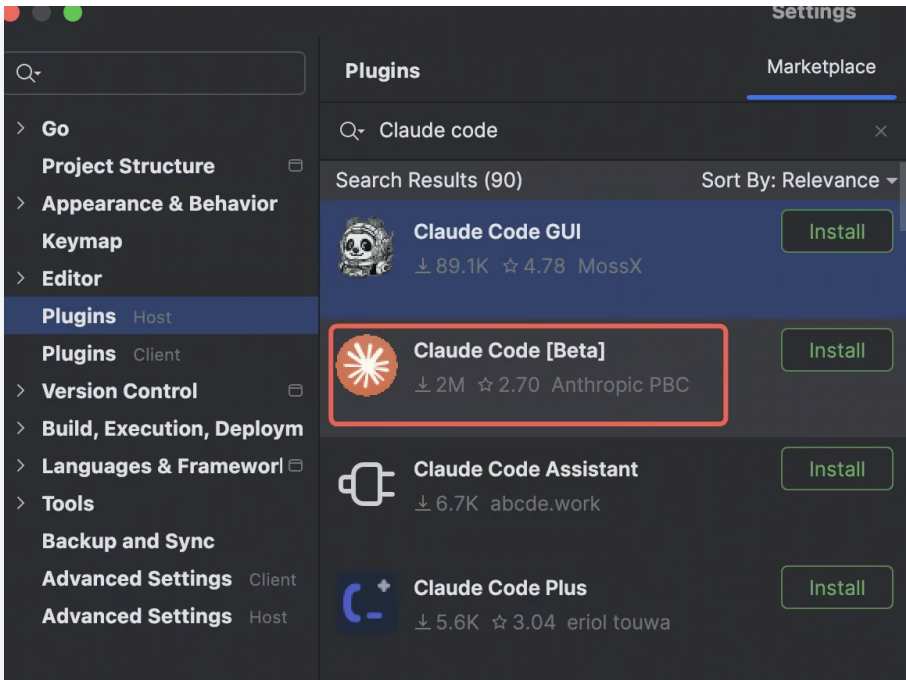
2. 安装成功后，单击 VS Code 右上角的 Claude Code 图标，进入 Claude Code 页面。



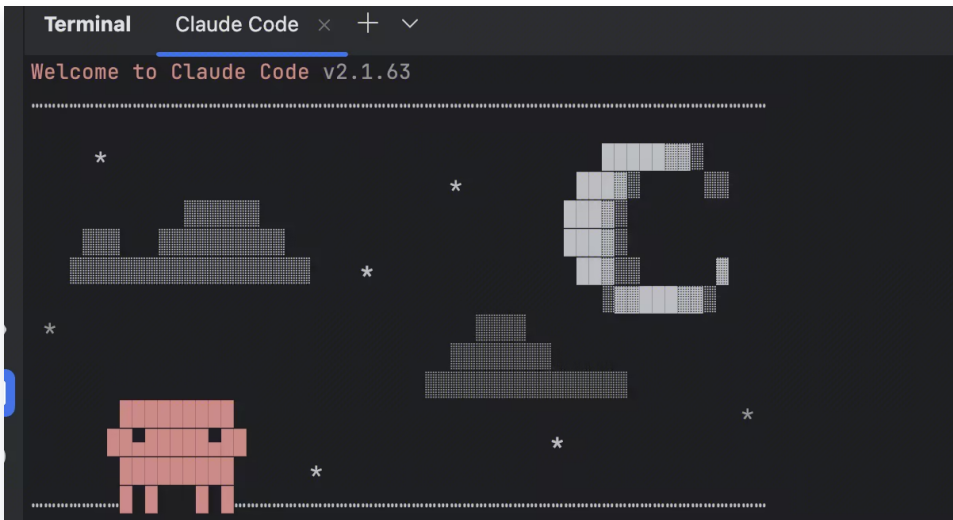
3. 切换模型：在对话框中输入 `/`，选择 **General config** 进入设置页面，在 **Selected Model** 中选择支持的模型，新建一个新窗口开始对话。

JetBrains

1. 在扩展市场中搜索 Claude Code 并安装。



2. 安装后重启 IDE，单击右上角图标即可使用，您可以通过 `/model <模型名称>` 命令切换模型。



更多信息

如需进一步了解 Claude Code 的更多功能，请参考 [Claude Code 官方文档](#)。

Codex

最近更新时间：2026-04-03 16:27:22

Codex 是由 OpenAI 开发的新一代 AI 智能编程助手，基于 GPT 架构打造强大代码生成引擎。Codex 通过深度学习技术，能够理解自然语言并转换为高质量代码，支持 Python、JavaScript、Java 等多种编程语言。本文介绍如何在 Codex 中配置与使用腾讯云大模型 Coding Plan。

前置条件

在开始配置前，请确保您已订阅 Coding Plan 套餐并获取 API Key，可参见 [Coding Plan 概述](#)。

安装 Codex

⚠ 注意：

新版本 Codex 采用 Responses API 接口，暂不支持 Chat/Completions API，只有 **0.80.0 及以下版本**才支持（详情请参见 [Codex 官方说明](#)）。请安装支持 Chat/Completions API 的旧版本 Codex，例如：0.80.0 版本。

在终端中执行以下命令安装 Codex。

```
npm install -g @openai/codex@0.80.0
```

在终端中执行以下命令，若输出版本号，则表示安装成功。

```
codex --version
```

配置 Coding Plan

完整复制以下内容并粘贴到 Codex 配置文件 `~/.codex/config.toml` 中。

1. 找到 Codex 的配置文件，配置文件路径如下，配置文件不存在时新建即可。

- Windows: `C:\Users\<<用户名>\.codex\config.toml`。
- macOS/Linux: `~/.codex/config.toml`。

2. 复制以下内容粘贴到 `config.toml` 配置文件中，将 `<MODEL_NAME>` 替换为 Coding Plan 支持的 [Model Name](#)。

```
model_provider = "Lkeap_Coding_Plan"
model = "<MODEL_NAME>"
[model_providers.Lkeap_Coding_Plan]
```

```
name = "Lkeap_Coding_Plan"
base_url = "https://api.lkeap.cloud.tencent.com/coding/v3"
env_key = "CODING_API_KEY"
wire_api = "chat"
```

配置环境变量

在命令行中将 CODING_API_KEY 环境变量设置为 Coding Plan 专属的 API Key。

macOS/Linux

在终端中执行以下命令，查看默认 Shell 类型。

```
echo $SHELL
```

根据 Shell 类型设置环境变量，命令如下：

Zsh

```
echo 'export CODING_API_KEY="USER_API_KEY"' >> ~/.zshrc
```

#配置生效

```
source ~/.zshrc
```

Bash

```
echo 'export CODING_API_KEY="USER_API_KEY"' >> ~/.bash_profile
```

#配置生效

```
source ~/.bash_profile
```

Windows

CMD

```
# 用您的Coding Plan API Key 替换 USER_API_KEY
set CODING_API_KEY "USER_API_KEY"

#新开 CMD 窗口检查是否生效
echo %CODING_API_KEY%
```

PowerShell

```
# 用Coding Plan的 API Key 替换 USER_API_KEY
[Environment]::SetEnvironmentVariable("CODING_API_KEY",
"USER_API_KEY", [EnvironmentVariableTarget]::User)

#新开一个窗口验证是否生效
echo $CODING_API_KEY
```

使用 Codex

1. 在终端执行如下命令并回车，启动 Codex 工具。

```
codex
```

2. 选择 Skip 跳过。

```
*+ Update available! 0.88.0 -> 0.107.0
Release notes: https://github.com/openai/codex/releases/latest
1. Update now (runs 'npm install -g @openai/codex')
2. Skip
3. Skip until next version
Press enter to continue
```

3. 选择采用 API Key 的方式使用 Codex，输入您套餐的 API Key。
4. 您可以输入问题或指令，开始对话。

```
> OpenAI Codex (v0.0.0)
model: gpt-5 /model to change
directory: -

Tip: NEW! Try shell snapshotting to make your Codex faster. Enable in /experimental!
Support for the "chat" wire API is deprecated and will soon be removed. Update your model provider to "responses".

: 你好

: 你好！有什么我可以帮你的吗？我可以协助你处理代码、调试问题、解释项目结构等编程相关的任务。
```

Cline

最近更新时间：2026-04-03 16:27:22

Cline 是一款内置于编辑器和终端的 AI 编码助手。它可以读写文件、运行终端命令、使用浏览器，并通过自然对话帮助您构建功能。本文介绍如何在 Cline 中配置与使用腾讯云大模型 Coding Plan。

前置条件

在开始配置前，请确保您已订阅 Coding Plan 套餐并获取 API Key，可参见 [Coding Plan 概述](#)。

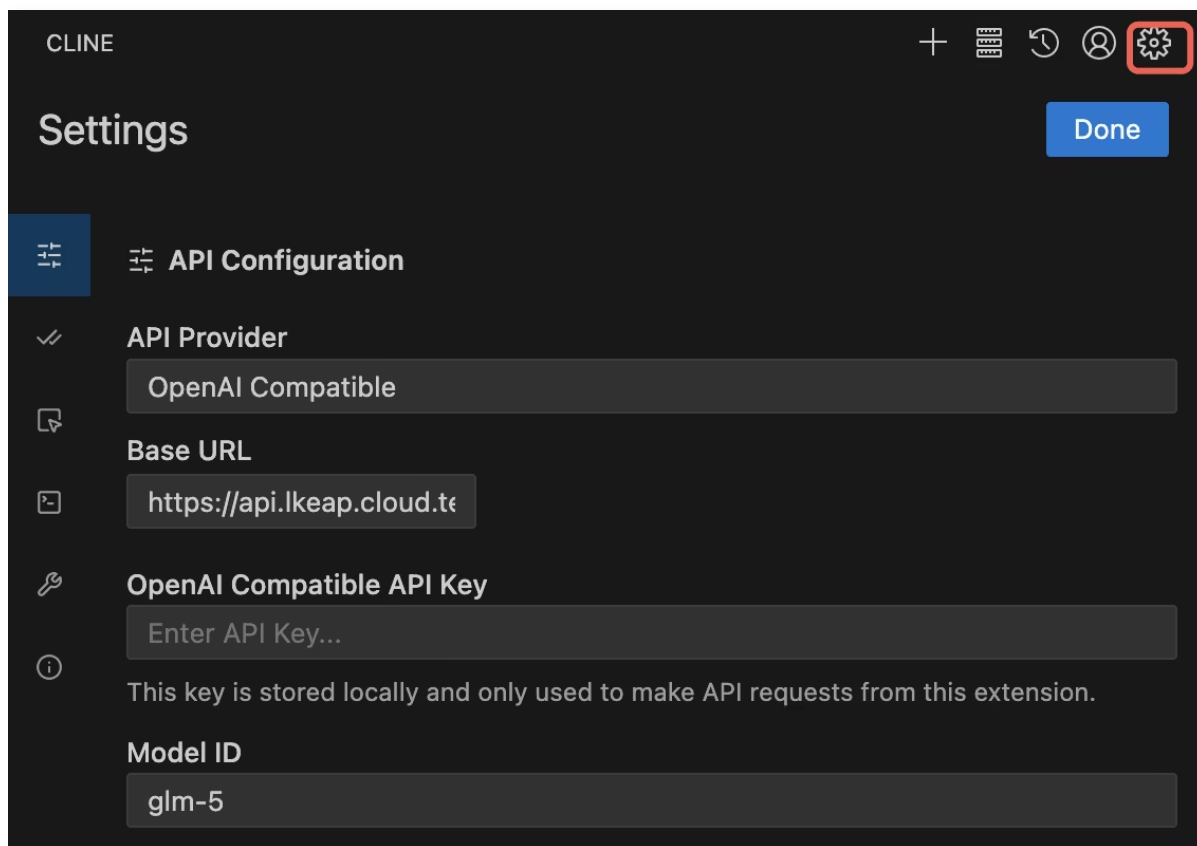
安装 Cline

打开 VSCode，在扩展市场搜索并安装 Cline，详情可参考 [Cline 官方安装指引](#)。

配置 Coding Plan

Cline 插件安装完成后，单击右上角的设置按钮，配置以下信息：

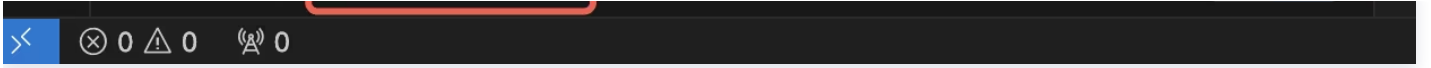
- API Provider: 选择 **OpenAI Compatible** (Coding Plan 接口兼容 OpenAI 标准)。
- Base URL: `https://api.lkeap.cloud.tencent.com/coding/v3`。
- API Key: 填写 Coding Plan 专属的 API Key。
- Model ID: 填写 Coding Plan 支持的 [Model Name](#)。



使用 Cline

配置完成后，您可以看到对话框下方显示对应的模型配置，并可以通过对话框与模型进行交互。

The screenshot shows the Cline AI interface. At the top, there's a header with 'CLINE' and various icons. Below that, a chat window shows a message: '你是谁呢' (Who are you?). A progress bar indicates 5.0k tokens used out of 128.0k. A green notification box says '1/1 All tasks have been completed!'. Below this, the AI is 'Thinking >'. A large green checkmark and the text 'Task Completed' are displayed. The AI's response is: '你好! 我是 Cline, 一位技术精湛的软件工程师助手。我具备以下能力:' followed by a bulleted list of capabilities: '编程语言: 熟悉多种编程语言, 包括 Python、JavaScript、TypeScript、Java、C++、Go、Rust 等'; '框架与技术: 了解主流框架如 React、Vue、Node.js、Django、Spring Boot 等'; '开发工具: 可以使用终端命令、读写文件、搜索代码等工具帮助你完成任务'; '最佳实践: 熟悉设计模式、代码规范和软件工程最佳实践'. Below the list, it says '我可以帮你:' followed by another bulleted list: '编写和修改代码', '调试和解决问题', '重构和优化代码', '创建新项目', '解释技术概念'. It ends with '有什么我可以帮助你的吗?'. At the bottom, there's a blue button 'Start New Task', a text input field with a placeholder 'Type a message...', and a status bar showing '@ + openai-compat:glm-5' and buttons for 'Plan' and 'Act'.



Cursor

最近更新时间：2026-04-03 16:27:22

Cursor 是基于 VS Code 深度改造、AI 原生的代码编辑器，内置 GPT-4/Claude 等模型，支持自然语言生成、代码理解、跨文件重构、智能调试与自主代理（Agent）。本文介绍如何在 Cursor 中配置与使用 Coding Plan。

前置条件

在开始配置前，请确保您已订阅 Coding Plan 套餐并获取 API Key，可参见 [Coding Plan 概述](#)。

安装 Cursor

官网下载安装包：通过 [Cursor 官网](#) 下载并安装 Cursor。

⚠ 注意：

由于 Cursor 的限制，只有订阅了 Cursor Pro 及以上套餐的用户才支持自定义配置模型。

配置 Coding Plan

Cursor 安装完成后，在 Cursor 中，点击右上角的 **Cursor Settings**，选择 **Models** 页面。Models 模块的具体配置如下：

- OpenAI API Key：填写 Coding Plan 的专属 API Key。
- Override OpenAI Base URL：`https://api.lkeap.cloud.tencent.com/coding/v3`。
- Add Custom Model：填写 Coding Plan 支持的 [Model Name](#)。

使用 Cursor

配置完成后，即可在聊天面板中选择配置的模型进行交互。

Kilo CLI

最近更新时间：2026-04-03 16:27:22

Kilo CLI 是 Kilo Code 全栈式 AI 编程代理平台的命令行形态。本文介绍如何在 Kilo CLI 中配置与使用腾讯云大模型 Coding Plan。

前置条件

在开始配置前，请确保您已订阅 Coding Plan 套餐并获取 API Key，可参见 [Coding Plan 概述](#)。

安装 Kilo CLI

在终端执行如下命令安装 Kilo CLI。

```
npm install -g @kilocode/cli
```

安装完成后，您可以执行如下命令检查，输出版本号则表示安装成功。

```
kilo --version
```

配置 Coding Plan

1. 打开 Kilo CLI 的配置文件，配置文件路径如下：

- macOS/Linux: `~/.config/kilo/config.json`。
- Windows: `C:\Users\<用户名>\.config\kilo\config.json`。

2. 在配置文件中，复制粘贴如下配置，并保存。

⚠ 注意：

请您将代码中的 `<USER_API_KEY>` 替换为 Coding Plan 专属的 API Key，将 `<MODEL_NAME>` 替换为 Coding Plan 支持的 [Model Name](#)。

```
{
  "$schema": "https://kilo.ai/config.json",
  "provider": {
    "lkeap": {
      "npm": "@ai-sdk/anthropic",
      "name": "Lkeap Coding Plan",
      "options": {
```

```
    "baseUrl":  
    "https://api.lkeep.cloud.tencent.com/coding/anthropic/v1",  
    "apiKey": "<USER_API_KEY>"  
  },  
  "models": {  
    "glm-5": {  
      "name": "<MODEL_NAME>",  
      "options": {  
        "thinking": {  
          "type": "enabled",  
          "budgetTokens": 1024  
        }  
      }  
    }  
  }  
}
```

使用 Kilo CLI

配置完成后执行如下命令启动 Kilo CLI。

```
kilo
```

输入 `/models`，选择 Lkeep Coding Plan 分组下的模型。

Kilo Code

最近更新时间：2026-04-03 16:27:22

Kilo Code 是一款开源 AI 编程助手，全场景适配您的开发工作流，无论在 IDE、终端、浏览器中，还是移动办公场景均可使用。借助深度理解代码库的 AI 能力，高效生成代码、自动化代码评审、调试问题，加速研发交付。本文介绍如何在 Kilo Code 中配置与使用腾讯云大模型 Coding Plan。

前置条件

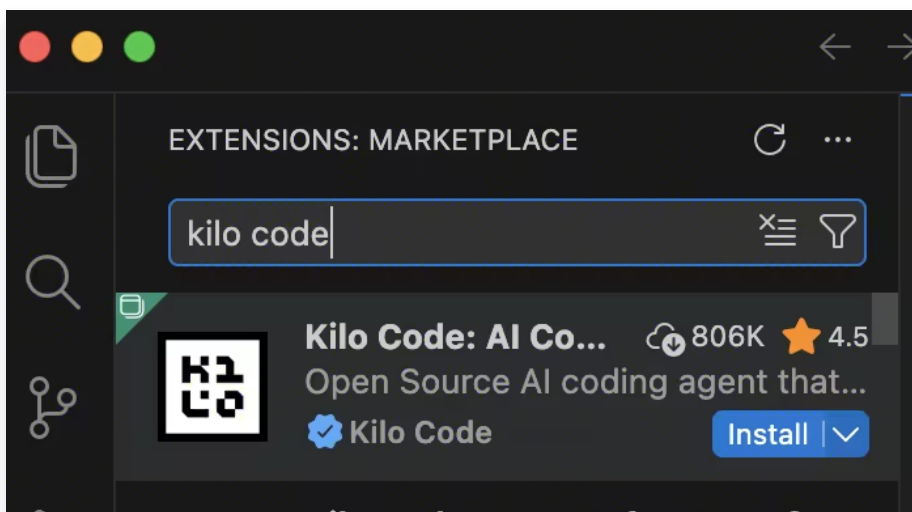
在开始配置前，请确保您已订阅 Coding Plan 套餐并获取 API Key，可参见 [Coding Plan 概述](#)。

安装 Kilo Code 并使用 Coding Plan

在 IDE 中搜索安装 Kilo Code 插件，下面分别以 JetBrains 和 VS Code 为例展示安装配置过程。

VSCode

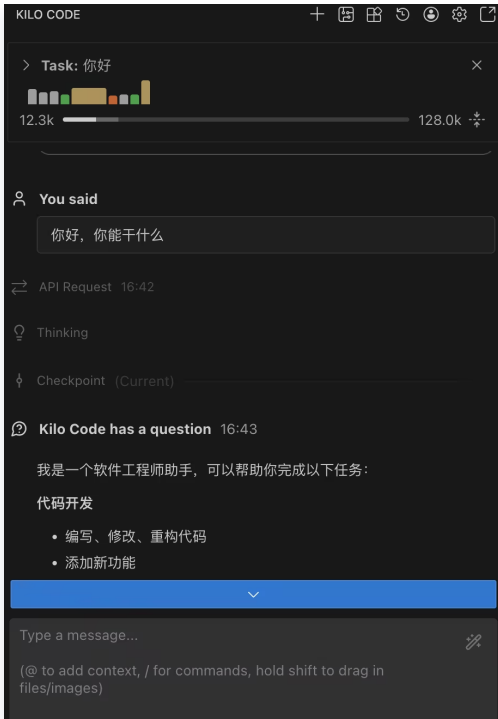
1. 打开 VSCode，在扩展市场中搜索 Kilo Code 并安装。



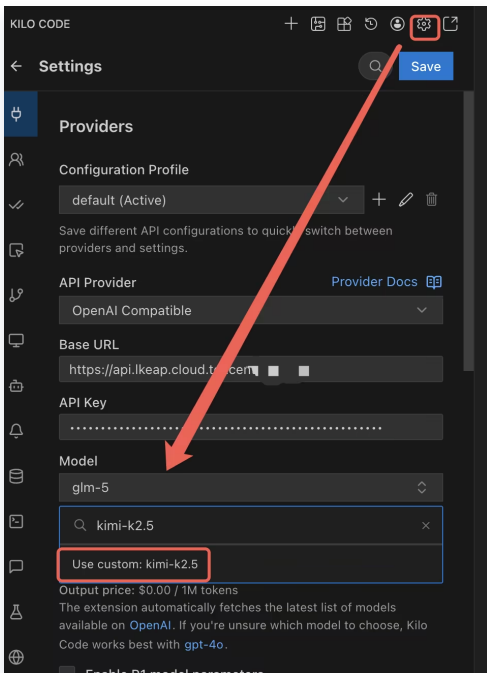
2. 单击 Kilo Code 右上角的设置按钮，选择 **Use your own API key**，并配置以下信息：

- API Provider: **OpenAI Compatible** (Coding Plan 接口兼容 OpenAI 标准)。
- Base URL: `https://api.lkeap.cloud.tencent.com/coding/v3`。
- API Key: Coding Plan 专属的 API Key。
- Model: 填写 Coding Plan 支持的 **Model Name**，单击 **Use custom**。

3. 配置完成后，您可以在输入框中输入需求，与模型进行交互。

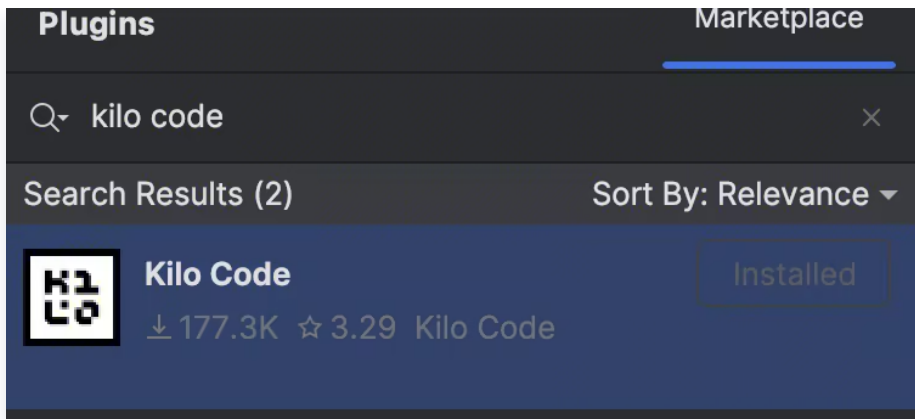


4. **模型切换:** 您可以通过单击右上角的设置, 在 Model 输入框搜索模型, 选择 Use custom 进行切换。



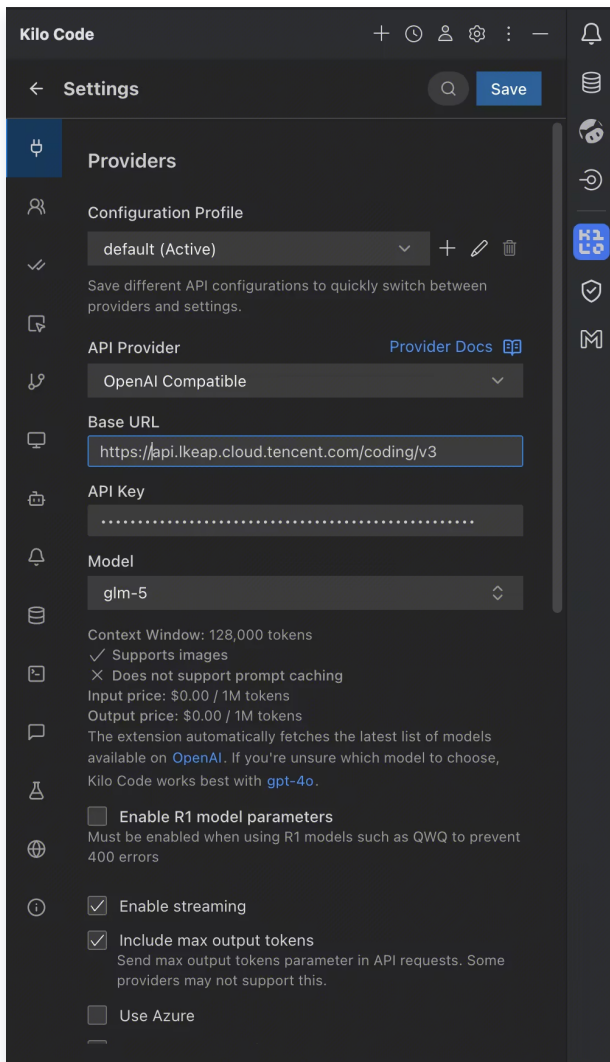
JetBrains

1. 打开 JetBrains , 在扩展市场中搜索 Kilo Code 并安装, 安装完成后重启 IDE 生效。

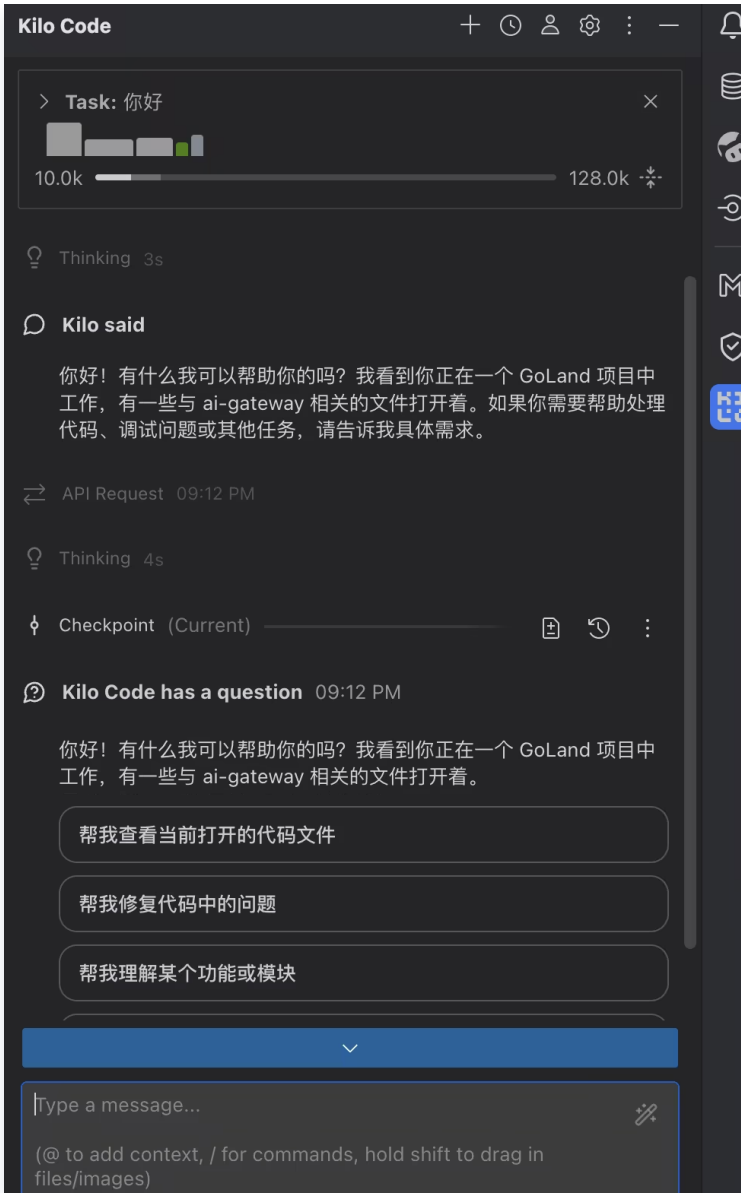


2. 安装成功后，单击 Kilo Code 右侧图标，选择 **Bring my own Key**，并配置以下信息：

- API Provider: **OpenAI Compatible** (Coding Plan 接口兼容 OpenAI 标准)。
- Base URL: `https://api.lkeap.cloud.tencent.com/coding/v3`。
- API Key: Coding Plan 专属的 API Key。
- Model: 填写 Coding Plan 支持的 **Model Name**，单击 **Use custom**。



3. 在 Kilo Code 中使用，单击右上角的设置按钮，修改 Model 即可切换模型。



Coding Plan 常见问题

最近更新时间：2026-04-03 16:27:22

购买及用量额度问题

可以购买几个 Coding Plan ？

同一主账号同时只能购买一个 Coding Plan 套餐（不区分 Lite 与 Pro 套餐）。如果一个主账号下有多个子账号，则主账号及其子账号合计只能购买一个套餐，且共用该套餐用量。

未开通自动续费，CodingPlan 套餐有效期多长？

Coding Plan 套餐自开通时起生效，有效期至次月对应日的同一时间结束。若次月没有对应日期，则有效期至次月最后一日 23:59:59（UTC+8）结束。

例如：2026 年 1 月 30 日 19:02:03 购买 Coding Plan，有效期至 2026 年 2 月 28 日 23:59:59。

Coding Plan 用量额度消耗完了怎么办？

每 5 小时、每周、每订阅月的额度用完后，可等待用量限额刷新。

Coding Plan 用量额度消耗完后会转为按量计费吗？

不会。Coding Plan 额度耗尽后，继续调用将会失败报错，不会自动转为按量付费。

如需继续使用，可升级至 Pro 版本获取更多额度，或等待下一订阅周期额度刷新。

Coding Plan 套餐用量额度刷新规则是什么？

- 每 5 小时：根据实际请求发生时间，滑动 5 小时窗口动态刷新。
- 每周：每周一 00:00:00 重置周限额。
- 每订阅月：每订阅月到期时间的下一秒重置订阅月限额。如：2026 年 3 月 6 日 19:02:03 购买 Coding Plan，在 2026 年 4 月 6 日 19:02:03 刷新订阅月限额。

Coding Plan 可以使用腾讯云模型的免费额度吗？

不可以。Coding Plan 为独立订阅产品，其计费体系不参与腾讯云 TokenHub 免费额度。

Coding Plan 套餐是否支持团队协作使用？

目前 Coding Plan 主要面向个人开发者。团队协作使用，请通过 [文本生成](#) 调用模型服务，采用后付费模式，详情请参见 [计费概述](#)。

如何查看 Coding Plan 套餐用量消耗？

可在控制台 [Coding Plan](#) 页面，查看套餐每 5 小时、每周、每订阅月的用量使用情况。



Coding Plan 和 Token Plan 有什么区别？

- 两款都是平台的大模型服务订阅套餐，Token Plan 是 Coding Plan 的全新升级，Coding Plan 之前是基于 Coding 场景设计的，而 Token Plan 是专门面向龙虾场景和编程场景设计的专属套餐，会有更多套餐档位可选，适合各种使用强度的场景，后续会持续集成在龙虾和编码场景下表现更优的模型，可以满足更多的使用场景。
- 用量统计方式和周期内使用限制不一样，Token Plan 是基于 Token 用量统计资源消耗，在订阅周内没有滑动窗口限制用量，您可以更灵活使用 Tokens。

我已经购买了 Coding Plan，还能再购买 Token Plan 吗？

Coding Plan 和 Token Plan 是独立的两款套餐，可以同时购买，两个套餐的用量和 API Key 都是互相独立的，双套餐权益不冲突。

使用相关问题

购买后如何获取 API Key？

您可以在控制台 [Coding Plan](#) 页面，获取并复制 Coding Plan 专属 API Key（格式为 `sk-sp-xxxx`）。



Coding Plan 可以生成多个 API Key 吗？

仅支持生成一个 API Key。

如何重置 API Key？

您可以在控制台 [Coding Plan](#) 页面，单击 API Key 右侧的**重置**。

⚠ 注意：

- 此操作将立即生成新的 API Key，并使旧 API Key 失效（5 min 左右）。
- 所有使用旧 API Key 的应用/服务将立即无法调用 API，请确保已准备好更新配置。
- 重置后无法恢复旧 API Key，请谨慎操作。

Coding Plan 到期不续费，重新开通后 API Key 会重置吗？

不会重置。

是否可以使用模型列表以外的模型？

Coding Plan 当前仅支持使用 [模型列表](#) 内的模型，使用其他模型将会报错。

Coding Plan 有并发请求数限制吗？

Coding Plan 没有额外的并发请求数限制。使用时受套餐用量限制，包括每 5 小时、每周、每订阅月的请求次数上限，具体请参见 [Coding Plan 概述](#)。

Lite 和 Pro 套餐的模型性能一样吗？

Lite 基础套餐与 Pro 高级套餐的模型性能一致，2 种套餐仅用量限制不同。

各模型的上下文长度是多少？最大输出长度是多少？

模型	model 参数值	上下文长度 (Tokens)	最大输出长度 (Tokens)
Tencent HY 2.0 Instruct	hunyuan-2.0-instruct	144k	16k
Tencent HY 2.0 Think	hunyuan-2.0-thinking	192k	64k
MiniMax-M2.5	minimax-m2.5	200k	192k
Kimi-K2.5	kimi-k2.5	256k	256k
GLM-5	glm-5	200k	128k
Hunyuan-T1	hunyuan-t1	96k	64k
Hunyuan-TurboS	hunyuan-turbos	48k	16k

常见错误码

错误码	错误信息	可能原因	解决方法
401	unauthorized	Base URL 或者 API Key 配置错误	1. 确认 URL 和 API Key 参考 Coding Plan 指引 正确获取。 2. 参考 Coding Plan 接入 AI 工具 指引 ，确认 URL 和 API Key 配置正确。
429	tpm rate limit exceeded	系统负载高触发限流	请重试 1-2 次或者切换其他模型。

2003 3	invalid model	模型名称配置错误	跟 Coding Plan 支持模型 比对，检查配置的模型名称是否正确。
2005 9	input length too long	输入长度超出了允许范围	可通过新建会话解决报错。
2009 7	hour allocated quota exceeded	每5小时请求额度已用完	每 5 小时是滑动窗口限制，遇到此问题可以稍后重试，或升级至 Pro 套餐。
2009 7	week allocated quota exceeded	每周请求额度已用完	等待至每周一 00:00:00 (UTC+8) 额度重置，或升级至 Pro 套餐。
2009 7	month allocated quota exceeded	每个订阅月请求额度已用完	等待至下个订阅月的同一时刻（订阅时刻）额度重置，或升级至 Pro 套餐。