

# CodeGeeX插件试用指南

## 1. 代码生成与续写

代码生成与续写是CodeGeeX插件最基本的功能。用户在IDE中编写代码的过程中，插件会根据光标位置之前的代码及注释，预测并生成后续可能需要的代码。此功能的主要目的是在尽量不改变用户原有的编程习惯的前提下，为用户提供自动生成的代码建议，从而提高开发效率。此功能的典型使用场景如下：

- 用户按照正常开发时的方式撰写注释，描述后续代码段所实现的功能，插件根据注释生成代码；
- 用户编写函数定义，准确清晰地命名函数名称、参数名称，插件根据这些信息生成代码；
- 用户在编程任何代码的过程中，在出现停顿时，插件根据前文自动预测后续代码
- 以上几种情况的综合

举例：

用户输入注释及函数定义

```
1 /**
2  * @return ArrayList of all visible widgets
3  */
4 public ArrayList<Widget> getVisibleWidgets() {
5
6
```

插件给出生成的建议（灰色部分）

```
1 /**
2  * @return ArrayList of all visible widgets
3  */
4 public ArrayList<Widget> getVisibleWidgets() {
5     ArrayList<Widget> widgets = new ArrayList<Widget>();
6     for (Widget widget : getRootWidget()) {
7         if (widget.isVisible()) {
8             widgets.add(widget);
9         }
10    }
11    return widgets;
```

```
12 }  
13
```

用户按Tab键，接受生成的代码

```
1 /**  
2  * @return ArrayList of all visible widgets  
3  */  
4 public ArrayList<Widget> getVisibleWidgets() {  
5     ArrayList<Widget> widgets = new ArrayList<Widget>();  
6     for (Widget widget : getRootWidget()) {  
7         if (widget.isVisible()) {  
8             widgets.add(widget);  
9         }  
10    }  
11    return widgets;  
12 }  
13
```

Tips:

- a. 注释中用自然语言描述的部分，应按照正常编写注释时的叙述方式简单、准确地描述代码段的功能。不应把它当作ChatGPT类的工具，用对话、指令的方式描述，也不宜提出过于复杂或笼统的任务。

例如，以下是比较好的描述方式：

- ✓ • // 快速排序函数
- // Given two non-negative integers, num1 and num2 represented as string, return the sum of num1 and num2 as a string.
- // read the data in file "abc.txt", then print it out

以下叙述形式CodeGeeX可能无法准确理解：

- ✗ • // 写一个 python 脚本自动抓取 GitHub Trending 中有关ChatGPT 的项目
- // 我想用Python爬取豆瓣电影TOP250的数据，代码要怎么写？
- // 请写一段javascript代码求解鸡兔同笼问题：鸡和兔一共35个头，94个脚，求鸡兔分别多少。

- b. 插件会根据当前光标位置自动判断生成一行或生成多行。CodeGeeX作为辅助工具，并不保证每次生成的是一段完整的代码。如果生成的代码没有完成一项功能，只需要让插件再次调用模型，继续向后生成，直到完成即可。
- c. 插件对于给定的输入，多次生成的结果可能不同。这是因为通过大模型进行生成，结果带有随机性。这种随机性可以从宏观统计上提高生成的准确率。如果希望降低随机性，可以在插件设置中调整部分参数设置，但是不建议一般用户调整这些设置。

## 2. 自动添加注释

自动添加注释是CodeGeeX插件的一项特色功能。它可以给指定的代码添加逐行的注释，解释每句代码的含义。使用时，在IDE中选中需要注释的代码段，再在右键菜单里调用添加注释功能即可。

Tips:

- a. 此功能只能做逐行的注释，不能做整段代码功能的解释。
- b. 每次能注释的代码长度有限，每次添加注释的代码建议不要超过一屏。

## 3. 代码翻译

代码翻译是CodeGeeX插件的一项特色功能。它可以将代码片段从一种语言翻译成另外一种语言。

Tips:

- a. 目前翻译功能是按照代码语义进行翻译，并非逐行、逐字翻译。
- b. 翻译功能只能针对代码片段进行翻译，暂时无法实现整个文件或工程的翻译。

## 4. 问答功能 (Ask CodeGeeX)

问答功能是CodeGeeX插件的选配功能。在CodeGeeX的个人用户版本中可以免费使用Ask CodeGeeX功能。企业版如需要此功能，需要搭配ChatGLM模型。

问答功能 (Ask CodeGeeX) 有以下两种主要的使用场景：

- a. 直接在输入框中提出任何技术问题，AI可以给出回答。这个功能可以一定程度上取代原先开发过程中遇到问题时，转到浏览器使用搜索引擎寻找问答解答的过程，提高解决问题的效率，使开发过程更加专注。
- b. 在IDE中选中部分代码，在输入框中针对这段代码提出问题，例如“这段代码有什么需要优化的地方？”。或者输入“/explain”、“/comment”等命令，直接针对代码进行操作。这一方式可以更灵活地利用AI能力，解决开发中的实际问题。

使用ChatGLM的模型可以通过调用ChatGLM API或进行ChatGLM模型私有化部署实现。