

文档密级:外部公开

iFlyCode 智能编程助手

产品白皮书

版本 <2.0>

版本历史

日期	文档版本	描述	升级说明	作者
2023/8/15	1.0			
2023/10/24	2.0			

科大讯飞

2023 年 8 月

目录

目录.....	2
1 产品概述.....	3
1.1 产品套件.....	4
1.2 产品架构与组件.....	4
2 特点与优势.....	5
2.1 产品特点.....	5
2.2 产品优势.....	5
3 产品功能.....	7
3.1 代码生成.....	7
3.2 代码补齐.....	7
3.3 代码解释.....	8
3.4 代码纠错.....	9
3.5 SQL 生成/优化.....	9
3.6 单元测试.....	11
3.7 知识问答.....	11
3.8 企业研发质效管理平台.....	11
4 iFlyCode 使用指南.....	14
4.1 iFlyCode 安装指南.....	15
4.1.1 安装 iFlyCode 插件.....	15
4.1.2 注册讯飞开放平台账号.....	15
4.1.3 账号登录.....	15
4.2 快捷键和设置.....	17
5 应用场景.....	18
5.1 设计阶段.....	19
5.2 编码阶段.....	19
5.3 自测阶段.....	19
6 应用案例.....	21
6.1 解决“4K 智慧屏”产品书写卡顿问题.....	21
6.2 辅助“智慧课堂”产品跨平台改造.....	22
7 服务与支持.....	23
8 公司简介.....	23

1 产品概述

iFlyCode 智能编码助手是基于讯飞星火大模型代码能力的编程辅助工具，它可以通过自然语言处理、大模型等技术，为用户提供智能化的代码编写建议和错误检测功能。传统的编程方式需要手动输入代码，并进行反复的调试和修正，而 iFlyCode 智能编码助手可以通过自然语言描述需求，快速生成代码片段，并具备代码解释、代码纠错、单元测试、知识问答等功能，提升开发人员的工作效率。

iFlyCode 智能编码助手的核心是其背后的大模型，这个模型需要在大量的代码库上进行训练，以便能够理解各种编程语言的语法规则、代码风格和最佳实践。通过学习这些代码样本，模型可以预测出其他开发者可能会编写的代码段，并给出相应的建议。这些建议可能包括变量命名、函数设计、算法选择等方面的内容。

iFlyCode 智能编码助手还可以根据开发者的反馈不断进行学习和优化，提高代码生成的准确性和效率。它不仅可以提高编码效率，而且可以帮助开发者更好地理解业务逻辑和技术需求，提高代码质量和可维护性。极大地提高了程序员的工作效率和代码质量，降低了开发成本和风险。

iFlyCode 智能编码助手还可以帮助开发者学习新的编程语言和技能，提供相关的教程和示例代码。

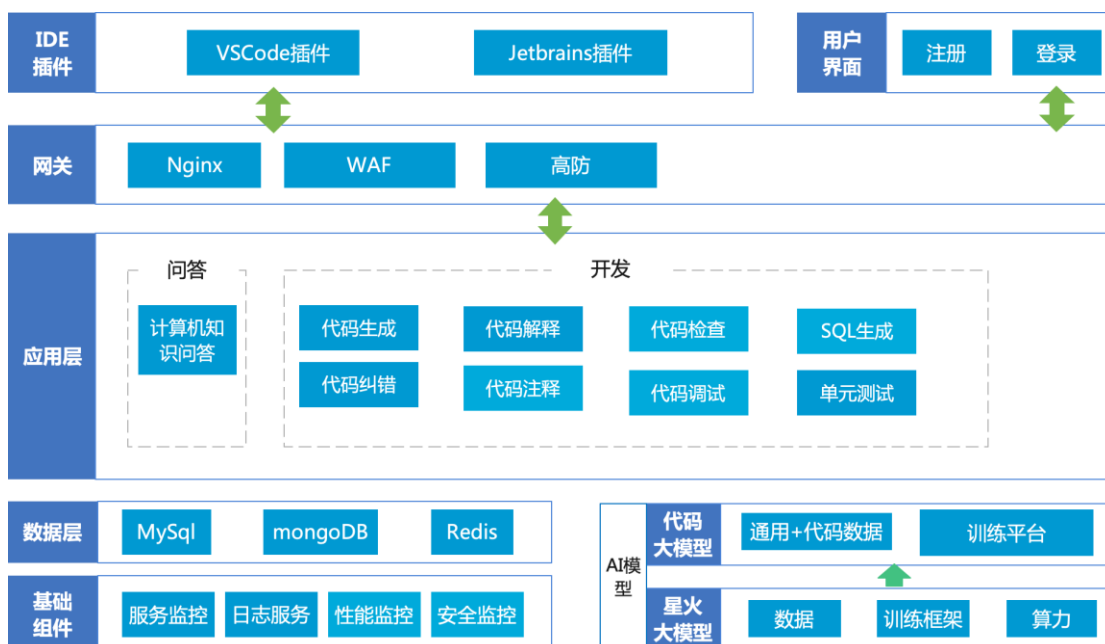
iFlyCode 智能编码助手在许多领域都有广泛的应用前景。在软件开发领域，它可以帮助开发者提高代码质量，缩短开发周期，降低维护成本。在教育培训领域，它可以帮助学生更好地理解和掌握编程语言和编程思维，提高学习效果。在企业级应用开发领域，iFlyCode 智能编码助手可以协助开发人员快速构建和优化复杂的业务系统，提高企业的竞争力。

1.1 产品套件



代码大模型产品套件

1.2 产品架构与组件



产品架构与组件

2 特点与优势

2.1 产品特点

目前，iFlyCode2.0 版本支持 Python、Java、C++、JavaScript 等主流开发语言，适配多种主流编辑器，可以在程序员编程过程中沉浸式交互生成代码建议，目前支持代码生成与补全、代码解释与注释、代码纠错与调试、代码检查、单元测试、SQL 生成与优化以及智能问答等功能。



iFlyCode 功能界面概览

目前，iFlyCode 提供个人版、企业版、私有化版三种服务模式，全面兼顾用户的便捷性和安全合规需求。此外，在智能编程助手的基础上，iFlyCode 通过质效管理平台进一步拓展应用场景，覆盖企业研发全流程。

2.2 产品优势

iFlyCode 智能编程助手基于讯飞星火大模型内核，量身打造、自主可控，并在此基础上进一步加强代码能力，涵盖的数据集范围广，语言支持全面。

iFlyCode 智能编程助手辅助开发者进行产品全流程开发，在编码前期，iFlyCode 能够辅助进行系统的需求分析、架构设计等前期工作。在编码阶段，iFlyCode 可以根据设计需求自动生成代码框架，并提供代码检查、调试等编程功能，快速转换设计文档到质量可控的程序代码，提升开发效率。

一方面，iFlyCode 提供沉浸式智能编码模式，作为一款 IDE 插件，iFlyCode 智能编程助手可以在集成开发环境中为开发者提供沉浸式智能编码体验，随时生成代码建议，帮助提升编码效率；同时，iFlyCode 的代码补全功能，能基于编写上下文推荐相关代码，辅助开发者高效完成编程工作。另外，iFlyCode 还能够检验代码质量，对不合规范之处实时提出修改建议，以一键方式协助开发者修正代码问题。

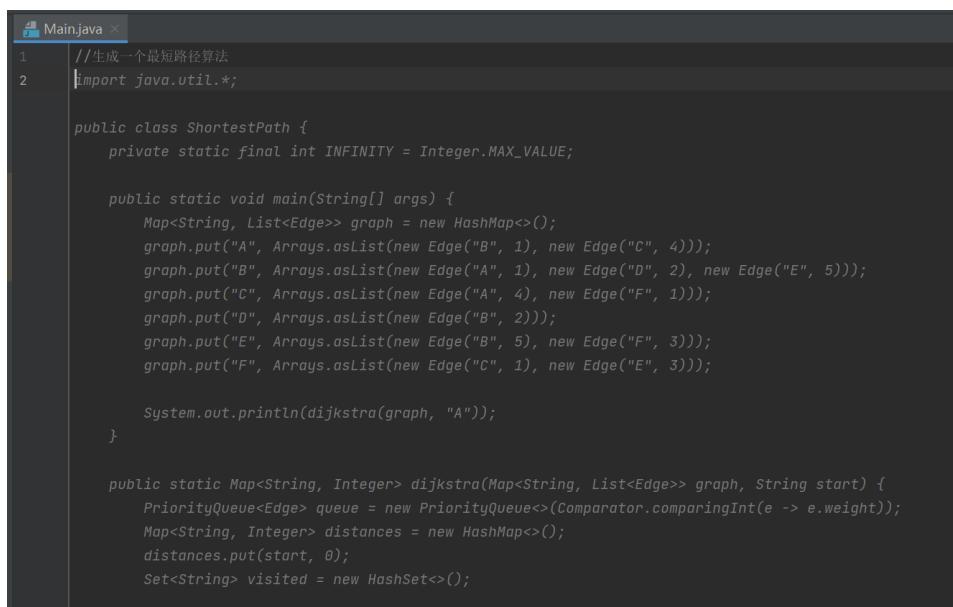
另一方面，iFlyCode 提供问答式智能互动模式，iFlyCode 支持开发者用自然语言提出问题，以获取专业化的编程指导。开发者可以询问系统代码的具体逻辑，iFlyCode 能够使用自然语言对代码进行解释说明。若代码发生错误，开发者询问错误原因，iFlyCode 能够精准定位问题所在并提出修改方案。在设计测试用例时，也可直接询问 iFlyCode 测试该代码，快速生成对应测试样本。

3 产品功能

iFlyCode 目前支持代码生成/补全、代码解释/注释、代码纠错/检查、代码调试、SQL 生成/优化、单元测试、智能问答等功能。此外，iFlyCode 还配备企业版管理后台，支持管理企业内的注册使用成员、统计分析企业内效能分析、成员组织管理、插件端和平台端的权限管理等功能。

3.1 代码生成

iFlyCode 支持在编辑器内根据注释、函数名生成代码，以注释生成代码为例，编写完成注释后，回车即触发代码建议，使用 Tab 键快捷采纳建议、Esc 拒绝建议或直接继续编程忽略建议：



```
1 //生成一个最短路径算法
2 import java.util.*;

public class ShortestPath {
    private static final int INFINITY = Integer.MAX_VALUE;

    public static void main(String[] args) {
        Map<String, List<Edge>> graph = new HashMap<>();
        graph.put("A", Arrays.asList(new Edge("B", 1), new Edge("C", 4)));
        graph.put("B", Arrays.asList(new Edge("A", 1), new Edge("D", 2), new Edge("E", 5)));
        graph.put("C", Arrays.asList(new Edge("A", 4), new Edge("F", 1)));
        graph.put("D", Arrays.asList(new Edge("B", 2)));
        graph.put("E", Arrays.asList(new Edge("B", 5), new Edge("F", 3)));
        graph.put("F", Arrays.asList(new Edge("C", 1), new Edge("E", 3)));

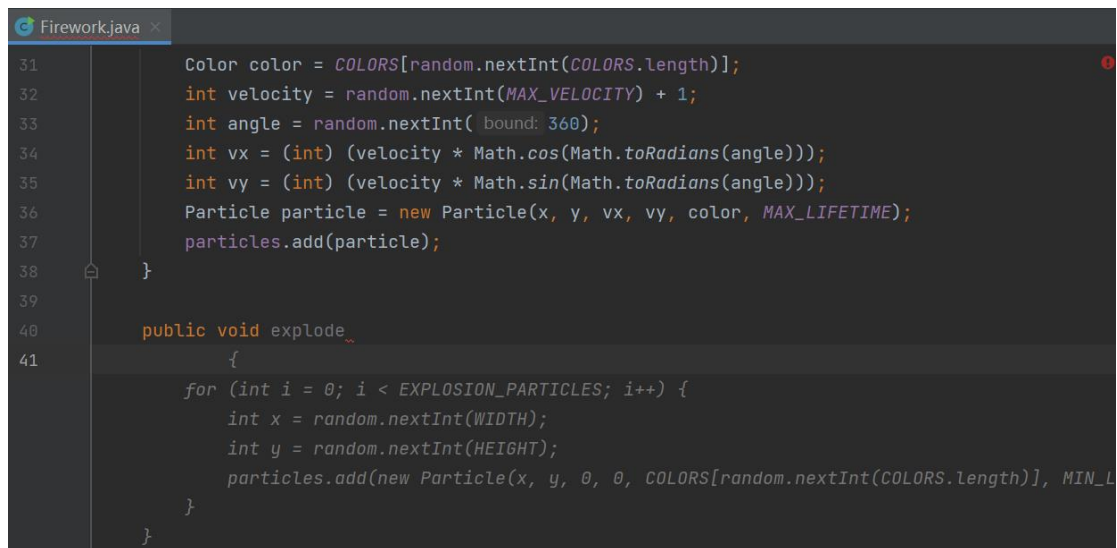
        System.out.println(dijkstra(graph, "A"));
    }

    public static Map<String, Integer> dijkstra(Map<String, List<Edge>> graph, String start) {
        PriorityQueue<Edge> queue = new PriorityQueue<>(Comparator.comparingInt(e -> e.weight));
        Map<String, Integer> distances = new HashMap<>();
        distances.put(start, 0);
        Set<String> visited = new HashSet<>();
    }
}
```

根据注释生成代码

3.2 代码补齐

iFlyCode 支持在编辑器内，通过方法名、上下文等信息触发代码补齐，以上下文信息为例，在已有上下文的代码中，回车、空格均会自动触发代码建议，使用 Tab 键快捷采纳建议、Esc 拒绝建议或直接继续编程忽略建议：



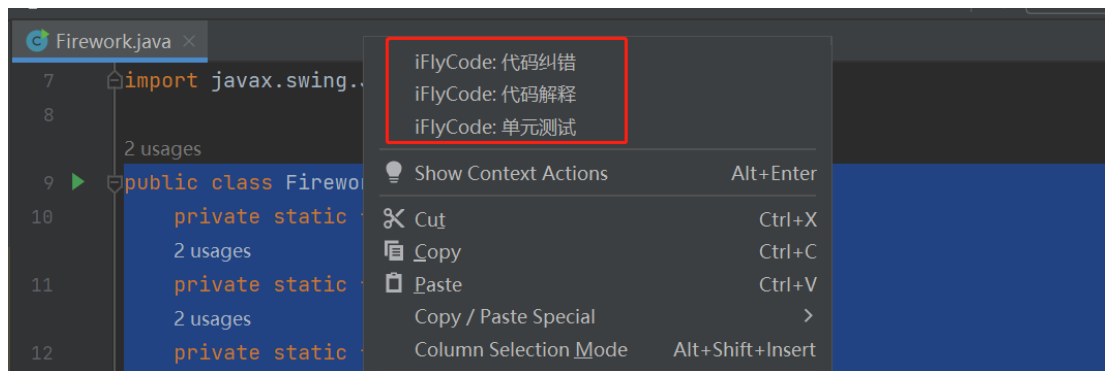
```
31 Color color = COLORS[random.nextInt(COLORS.length)];
32 int velocity = random.nextInt(MAX_VELOCITY) + 1;
33 int angle = random.nextInt( bound: 360);
34 int vx = (int) (velocity * Math.cos(Math.toRadians(angle)));
35 int vy = (int) (velocity * Math.sin(Math.toRadians(angle)));
36 Particle particle = new Particle(x, y, vx, vy, color, MAX_LIFETIME);
37 particles.add(particle);
38 }
39
40 public void explode_
41 {
    for (int i = 0; i < EXPLOSION_PARTICLES; i++) {
        int x = random.nextInt(WIDTH);
        int y = random.nextInt(HEIGHT);
        particles.add(new Particle(x, y, 0, 0, COLORS[random.nextInt(COLORS.length)], MIN_L
    }
}
```

根据上下文生成代码

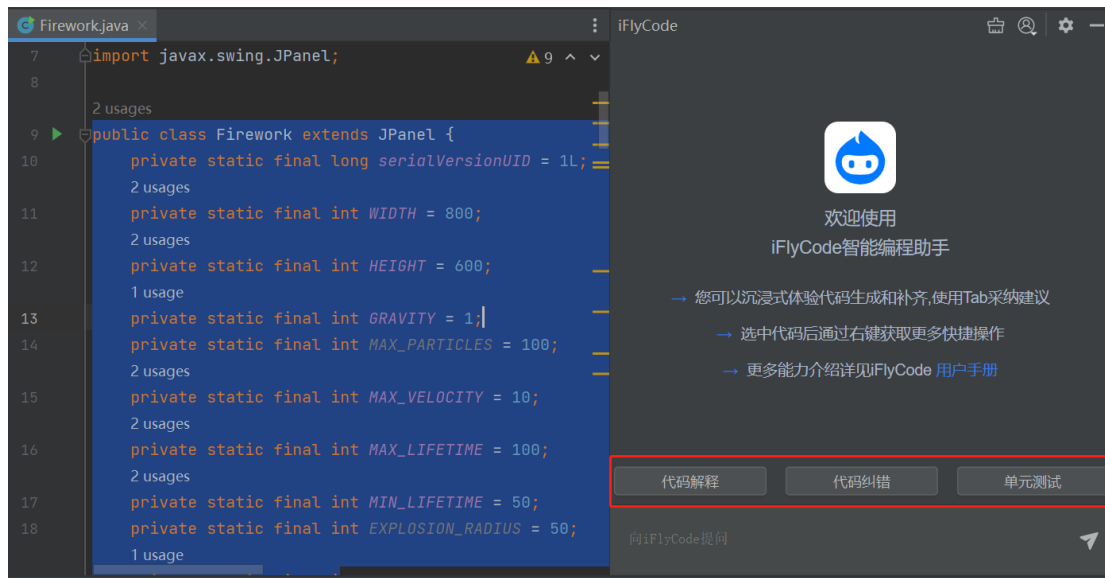
3.3 代码解释

iFlyCode 支持对选中代码进行代码解释，包括该段代码的作用和含义、代码中的类和方法的意义等。iFlyCode 提供两种快捷形式使用代码解释功能：

- 1)选中代码后，右键选择“iFlyCode: 代码解释”；
- 2)选中代码后，在对话框中上方选择“代码解释”快捷选项。



右键触发代码解释/代码纠错/单元测试



对话框快捷键触发代码解释/代码纠错/单元测试

3.4 代码纠错

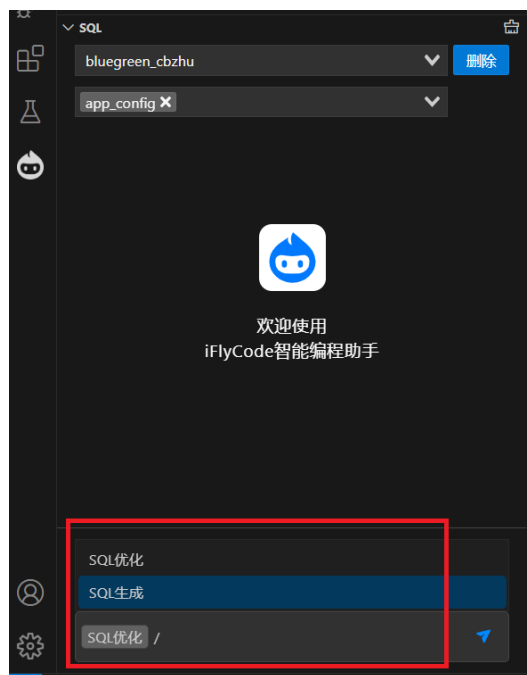
对选中代码进行代码纠错，包括该段代码中的拼写、语法和逻辑错误。

iFlyCode 提供两种快捷形式使用代码解释功能（如图 9、图 10）：

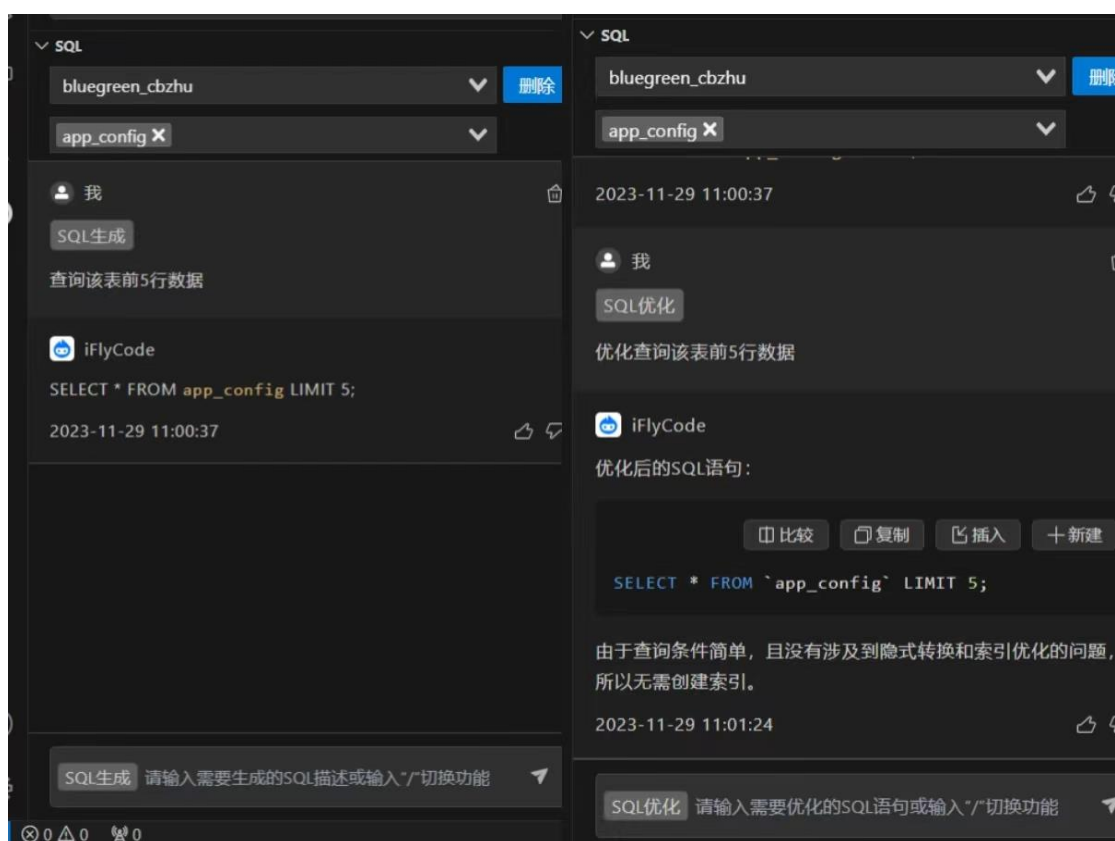
- 1)选中代码后，右键选择“iFlyCode: 代码纠错”；
- 2)选中代码后，在对话框上方选择“代码纠错”快捷选项。

3.5 SQL 生成/优化

iFlyCode 支持使用 SQL 生成与优化功能，导入数据库，在对话框框内描述并生成 SQL 语句，并支持对已有 SQL 语句的优化。



对话框快捷键触发 SQL 生成/优化功能



SQL 生成/优化功能演示

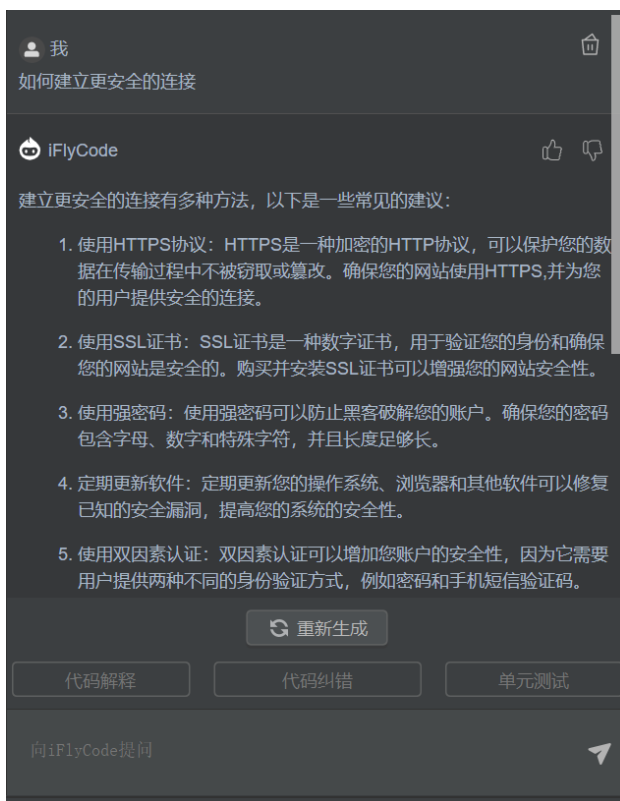
3.6 单元测试

iFlyCode 支持对选中代码生成单元测试用例。iFlyCode 提供两种快捷形式使用单元测试功能（如图 9、图 10）：

- 1)选中代码后，右键选择“iFlyCode：单元测试”；
- 2)选中代码后，在对话框上方选择“单元测试”快捷选项。

3.7 知识问答

iFlyCode 支持对选中代码进行提问或直接进行技术问题提问。



技术问答

3.8 企业研发质效管理平台

iFlyCode 企业研发质效管理平台目前包括数据看板、用户管理、权限管理、版本管理、认证管理等五个功能模块，分别介绍如下：

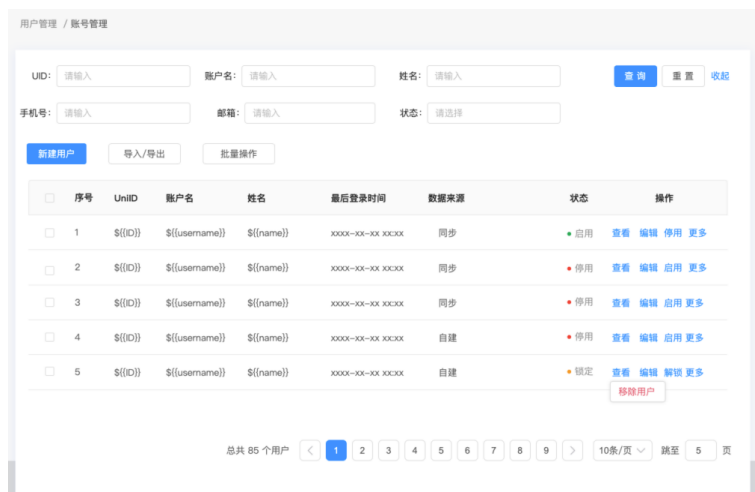
（1）数据看板：数据看板是用于统计和分析企业内成员使用 iFlyCode 代码大模型的使用效能分析的数据可视化分析看板，模块内包含了成员用量分析、成员分

析和效能分析三个板块。



企业质效管理平台部分看板示意图

(2) 用户管理：提供平台内企业、企业套餐、企业内成员的管理功能。用户管理功能包括但不限于成员注册、账号启用/禁用、成员信息管理、密码重置、成员分组管理等。



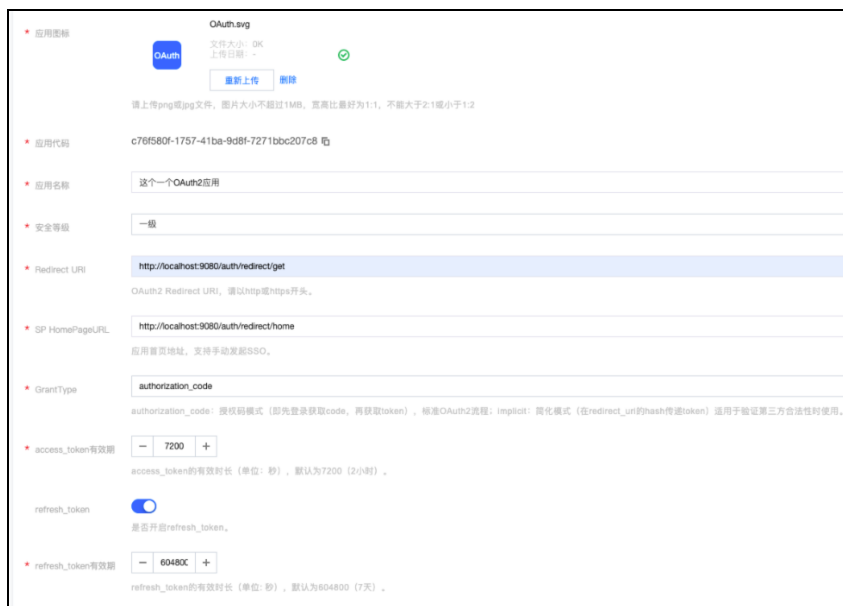
用户管理部分页面示意图

(3) 权限管理：提供平台内所有用户的查询、管理和个人公测授权功能。该模块采用 RBAC 模型，主要用于企业组织内部不同角色使用 iFlyCode 的使用权限。



权限管理部分页面示意图

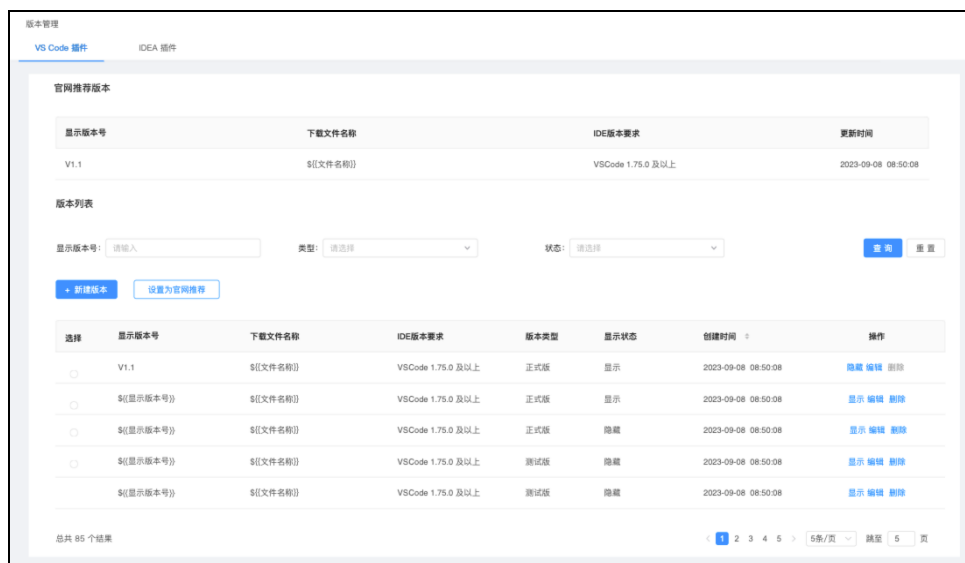
(4) 认证管理：用于配置 iFlyCode 认证登录的方式和安全策略，允许企业自行配置与其内部认证登录系统进行集成和对接。



The screenshot shows the OAuth2 configuration page. It includes fields for application icon, application code, application name, security level, redirect URI, SP Home Page URL, grant type, access token expiration, refresh token status, and refresh token expiration. The application code is c76f580f-1757-41ba-9d8f-7271bbc207c8. The redirect URI is http://localhost:9080/oauth/redirect/get. The grant type is authorization_code. The access token expiration is 7200 seconds. The refresh token is enabled and expires in 604800 seconds.

认证接口配置页面示意图

(5) 版本管理：提供插件端产品发布和管理历史版本文件和更新日志，用户可根据需求自行上传不同 iFlyCode 的安装配置包，并可设定插件端推送通知和邮件通知的配置。



The screenshot shows the version management interface. It includes a table for online recommended versions and a table for version list. The online recommended version table has columns for display version number, download file name, IDE version requirement, and update time. The version list table has columns for selection, display version number, download file name, IDE version requirement, version type, display status, creation time, and operation. The version list table shows 5 results.

版本管理部分页面示意图

4 iFlyCode 使用指南

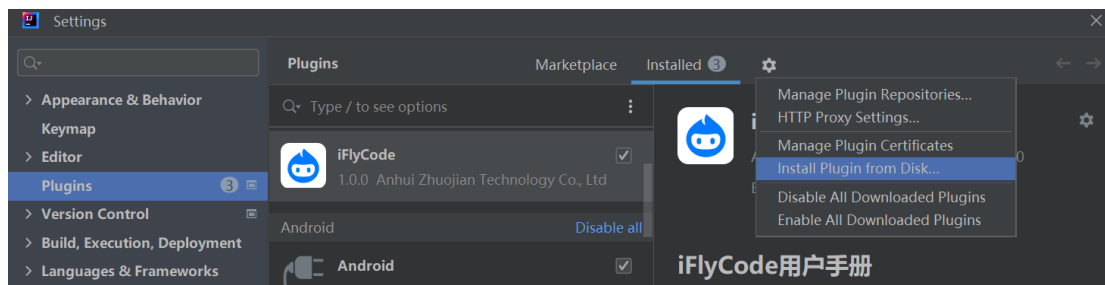
目前，iFlyCode 支持 VS Code、IntelliJ IDEA、CLion、PyCharm、WebStorm 等 IDE 插件，一键安装，即装即用。

4.1 iFlyCode 安装指南

4.1.1 安装 iFlyCode 插件

方法：从本地安装包安装插件

以 IDEA 为例：打开 IDEA，使用快捷键 Ctrl + Alt + S 或点击“File”-“Settings” - “Plugins”，点击右上方设置图标，选择“Install Plugin from Disk”



IDEA 安装插件

4.1.2 账号获取

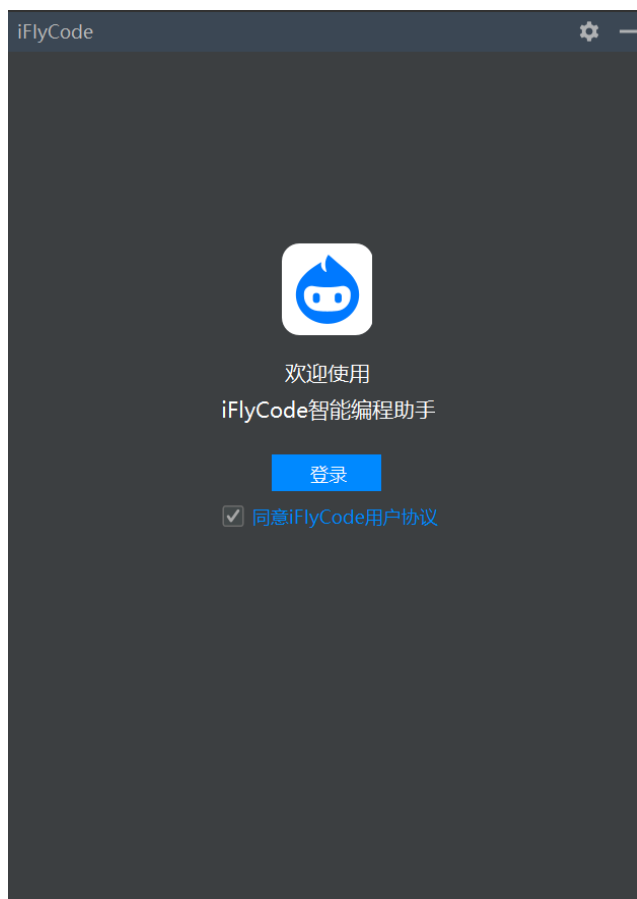
SaaS 版点击 <https://www.xfyun.cn/>，注册讯飞开放平台账号。

私有化部署版本，账号一般由管理员分发。

注：初始账号为员工企业邮箱，密码为 Abc1234%

4.1.3 账号登录

(1) 点击插件登录：



插件登录

(2) SaaS 版点击讯飞开放平台账号登录:



点击讯飞开放平台账号登录

私有部署版跳转至登录页，输入账号密码进行登录。

(3) 登录成功，返回 iFlyCode 插件即可开启智能编程之旅。SaaS 版若无权限，请申请试用；私有部署版若无账号，可向管理员获取。

4.2 快捷键和设置

日常使用中，iFlyCode 支持沉浸式生成/补齐代码，您只需进行正常编程工作，回车、空格等按键将自动生成代码建议，您可以使用 Tab 键采纳建议、Esc 拒绝建议或直接继续编程忽略建议。您可以选中代码后，通过右键或对话框上方选择代码解释/纠错/单元测试等功能、选中代码直接提问，或直接向 iFlyCode 提问技术问题。

iFlyCode 快捷键列表：

- Tab 采纳建议
- Esc 拒绝建议
- Alt+\ 主动触发建议

此外，iFlyCode 还提供多种自定义设置：

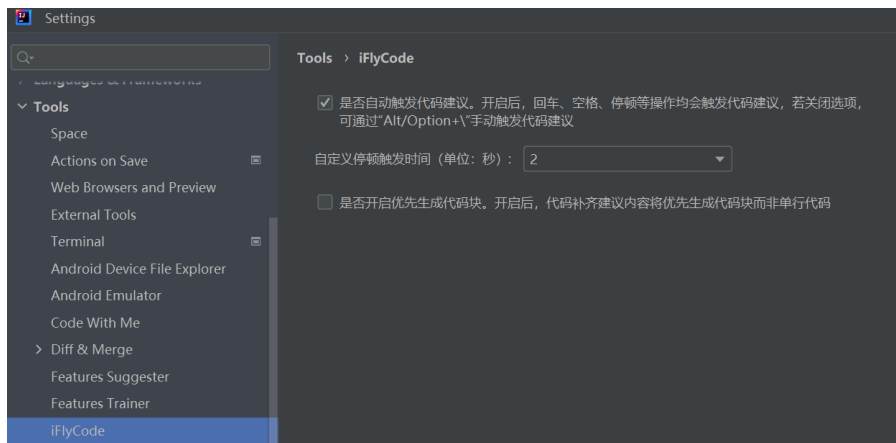
打开 iFlyCode 对话框，点击上方设置按钮，进入自定义设置页面。



iFlyCode 自定义设置按钮

iFlyCode 支持自定义设置，您可以根据使用习惯，自定义 iFlyCode 触发及交互：

- 自定义是否开启停顿触发及停顿触发的时间
- 自定义是否开启代码建议（若不开启则通过“Alt+I”主动触发代码建议）
- 自定义是否优先生成代码块（若不开启，则优先生成单行代码）



自定义设置页面

5 应用场景

iFlyCode 当前功能已覆盖研发流程中编码的大部分场景。在编码前期，辅助进行系统的需求分析、架构设计等前期工作。在编码阶段，iFlyCode 可以根据设计需求自动生成代码框架，并提供代码检查、调试等编程功能，快速转换设计文档到质量可控的程序代码，提升开发效率。



iFlyCode 覆盖研发全场景

5.1 设计阶段

在设计阶段，iFlyCode 的智能问答功能能够与开发者进行交互，根据提供的需求描述和问题，生成初步的设计方案，帮助开发者快速了解项目的整体框架和设计思路。

接下来，用户可以与 iFlyCode 继续交互，对初步设计方案进行细化和完善。通过向 iFlyCode 提问、讨论等方式，对设计方案进行调整和优化，提供更加符合用户需求的解决方案。

对于一些疑难杂症或复杂的设计问题，iFlyCode 还可以提供对比性的方案选择。通过分析不同的设计思路和技术选型，iFlyCode 可以为用户提供多个备选方案，辅助用户可以更好地理解各种方案的优缺点，并选择最适合自己的设计解决方案。

5.2 编码阶段

在编码阶段，iFlyCode 的代码生成功能可以辅助开发者完成部分代码的编写，iFlyCode 可以帮助开发者减少重复劳动，避免手动输入繁琐的代码，节省时间和精力。此外，iFlyCode 可以根据上下文和语法规则，智能推荐合适的代码片段，进一步提升编码效率。

其次，iFlyCode 的代码调试/纠错功能能够帮助开发者在编码过程中自动定位并修复代码问题。iFlyCode 可以根据代码逻辑和语法规则，分析代码中的错误，并提供相应的修改建议，大大减少手动调试的时间和工作量。

此外，iFlyCode 的代码解释/注释功能能够帮助开发人员快速理解代码逻辑。通过 iFlyCode 提供的代码解释和注释，开发人员可以更好地理解代码的功能、实现方式以及设计思路，提升团队开发效率，降低沟通成本，避免出现误解或错误。

5.3 自测阶段

iFlyCode 的代码检查功能可以帮助开发人员及时发现并修复代码问题，减少评审工作量，降低等待时间。通过一键选中进行问题报告修改意见，提高开发效

率。

iFlyCode 可以辅助开发人员一键生成代码块测试，快速生成测试用例，对已完成的代码进行测试，保证代码质量，并发现潜在的问题，有助于开发人员及时进行修复。同时，单元测试还可以帮助开发人员发现代码中的逻辑错误等，提高代码质量。

6 应用案例

iFlyCode 目前已在讯飞集团内部多个 BG/BU 的多个产线业务线项目中应用，取得显著成果。通过 iFlyCode 对讯飞内部多个项目、多轮实验的数据统计显示，iFlyCode 在设计阶段提效 50%，编码阶段提效 37%，单元测试阶段提效 44%。现在，iFlyCode 已服务客户数十家，注册用户 4W+，获得高度认可。

6.1 解决“4K 智慧屏”产品书写卡顿问题

电子白板的数字化板书是老师最常用的软件功能。讯飞教育事业部在南昌教育装备展的展览过程中发现，与竞品相比，在 4K 分辨率下，用户使用讯飞“4K 智慧屏”密集书写或擦除时会出现卡顿，严重影响体验。

在问题定位阶段，在板书擦除出现卡顿问题时，iFlyCode 通过多轮递进式问答排除不可行方案，最终精准定位根本原因，并给出了可直接使用的代码示例来修复这个难点问题。相比手工定位问题需耗费大量专家时间和试验成本。iFlyCode 辅助开发团队快速分析识别出卡顿问题的关键所在。

该软件复杂度高，代码量达 15 万行 C++ 和 3 万行 H5，采用 C++和 H5 混合开发模式，代码质量一般。而产品生命周期长，多次交接，存在难点。此外，电子白板研发力量不足，且 QT 图像渲染方面专家积累有限。

在解决方案设计阶段，“4K 智慧屏”团队投入 2 名研发进行定位和解决。在多次提问中，大模型提供了有效的解决思路和样例代码，如使用缓存、OpenGL 渲染、多线程等技术方案。团队经验证发现，多线程+异步队列方案最终解决了该技术难题。

在优化编码实现阶段，iFlyCode 的代码纠错功能指出了存在的代码错误，并给出了修改建议，辅助开发团队优化逻辑。在测试阶段，iFlyCode 辅助编写性能测试脚本，验证优化效果。

根据团队反馈，在整个解决过程中，大模型所提供的专业建议和解决方案，展现出技术领域资深专家的技术深度和丰富经验，帮助团队快速完成技术难题的攻关。最终，团队以 7 天时间解决了原计划 15 天的问题，整体提效 60%。

6.2 辅助“智慧课堂”产品跨平台改造

在讯飞教育事业部“智慧课堂”产品进行跨平台改造的案例中，客户采购了新的操作系统和硬件设备，需要对产品进行跨平台改造。团队成员之前都没有接触过这个新的硬件和操作系统平台，缺乏移植的经验，iFlyCode 在多个方面为项目组提供了帮助。

在项目移植初期，团队由于不熟悉 Linux 平台，选型、调研和框架搭建花费了大量时间。后来，团队发现使用 iFlyCode 问答功能能够协助快速调研跨平台技术方案，通过不断尝试提出合适的 Prompt，就能直接获取高质量的答案和框架代码，从而快速上手 Windows 到 Linux 的移植。

在编码阶段，移植过程中包含近 20 万行 C++ 代码，团队通过大模型逐步学习了 Windows 和 Linux C++ 差异后，发现大部分转换工作是机械性的。iFlyCode 的代码生成和纠错功能辅助完成了代码迁移工作，减少了大量重复性的转换工作，并自动修改部分编译错误，简化了调试过程，从而节省了大量时间。

移植中仍会遇到一些技术难题，过去需要大量时间才能解决。现在团队可以将问题上下文一起提供给 iFlyCode，通常能快速获得建设性意见或直接找到解决方案。根据项目组反馈，这提升了团队处理复杂问题的能力，并节省了大量时间。

在测试阶段，iFlyCode 自动生成单元测试代码，验证代码在新平台的正确性。自动生成测试用例，减轻了编写测试代码的工作量。

通过协助跨平台改造的多个阶段，根据项目组的数据，相比预计手工改造的 3 个月时间，6 个人的团队在 iFlyCode 的辅助下用 1 个月完成了 20 万行代码的迁移，加快了交付进度。

7 服务与支持

如果您在安装或使用中有任何问题或者建议，可以通过以下方式联系我们：

■ 合作咨询

电话：18056031128

■ 电子邮件支持

联系邮箱：iflycode_support@iflytek.com

8 公司简介

- 国家智能语音高新技术产业化基地
- 语音及语言信息处理国家工程实验室
- 国家规划布局内重点软件企业
- 国家创新型企业
- 国家级企业技术中心
- 国家高技术产业化示范工程
- 中国语音产业联盟理事长单位
- 中国中文语音交互技术标准工作组组长单位
- 国家博士后科研工作站

科大讯飞股份有限公司是一家专业从事智能语音及语言技术、人工智能技术研究，软件及芯片产品开发，语音信息服务及电子政务系统集成的国家级骨干软件企业。公司的智能语音核心技术代表了世界的最高水平。2008 年，科大讯飞在深圳证券交易所挂牌上市，股票代码：002230。

科大讯飞成立于 1999 年，是我国产业化实体中，在语音技术领域中基础研究时间最长、资产规模最大、历届评测成绩最好、专业人才最多及市场占有率最高的公司。

科大讯飞作为中国最大的智能语音技术提供商，在智能语音技术领域有着长期的研究积累，并在语音合成、语音识别、口语评测、自然语言处理等多项技术上拥有国际领先的成果。科大讯飞是我国唯一以语音技术为产业化方向的“国家 863 计划成果产业化基地”、“国家规划布局内重点软件企业”、“国家高技术产业化示范工程”，并被原信息产业部确定为中文语音交互技术标准工作组组长单位，牵头制定中文语音技术标准。

2003 年、2011 年，科大讯飞两次荣获“国家科技进步奖”；2005 年、2011 年两次获得中国信息产业自主创新最高荣誉“信息产业重大技术发明奖”。自 90 年代中期以来，在历次的国内外语音合成评测中，各项关键指标均名列第一；2008 年至 2012 年，连续五年在国际说话人、语种识别评测大赛中名列前茅。2011 年，“国家智能语音高新技术产业化基地”、“语音及语言信息处理国家工程实验室”相继落户合肥，有利于进一步汇聚产业资源，提升科大讯飞产业龙头地位。

基于拥有自主知识产权的世界领先智能语音技术，科大讯飞已推出从大型电信级应用到小型嵌入式应用，从电信、金融等行业到企业和消费者用户，从手机到车载，从家电到玩具，能够满足不同应用环境的多种产品。科大讯飞已占有中文语音技术市场 70% 以上市场份额，以讯飞为核心的中文语音产业链已初具规模。随着移动互联网“语时代”的到来，科大讯飞率先发布了全球首个提供移动互联网智能语音交互能力的“讯飞语音云”平台，并持续升级优化。基于该平台，科大讯飞相继推出了“讯飞输入法”、“灵犀”等示范性应用，并与广大合作伙伴携手推动各类语音应用深入到手机、汽车、家电、玩具等各个领域，引领和推动着移动互联网时代大潮下输入和交互模式的变革。

科大讯飞在语音技术核心研究和产业化方面的突出成绩引起了社会各界的广泛关注，习近平、李克强、张德江、俞正声、刘云山、张高丽等多位党和国家领导人亲临科大讯飞视察，对智能语音技术的重大价值及科大讯飞做出的创新工作均给予充分肯定。国家各行业部委和权威机构已形成了科大讯飞“中文语音产业国家队”的基本共识。