

亚马逊云科技

利用生成式人工智能 推动软件开发转型

使用 Amazon CodeWhisperer 更快、更安全地进行开发



构建软件会遇到各种困难

企业越来越多地依靠软件来实现差异化、推动转型以及在行业中取得领先地位。但是，对于大多数企业而言，快速踏上软件驱动的转型之旅往往面临各种困难。这不仅是因为全球软件开发人员一直处于短缺中，¹ 而且开发人员没有足够的“专注时间”² 来处理重要的业务问题。

即使在开发人员可以专注于编写代码的有限时间内，他们也会遇到各种各样的障碍。除了要花费很大一部分时间来手动编写简单且大同小异的代码（例如单元测试、字符串操作、处理列表和上传文件等）之外，他们还要花费大量时间了解、学习和跟上复杂且不断变化的工具和技术发展趋势（新的编程语言、SDK、API、框架、工具包等），这样才能使用最新的技术和服务来构建创新的解决方案。这进一步压缩了可用于开发创新功能、差异化功能或关键业务功能的时间。

为了减少这种时间压力，开发人员经常会尝试互相学习和通过开发者论坛共享代码，但这可能会导致意外复制的代码无意中侵犯了开源软件版权或许可证。赶在截止日期之前交付成果的压力，重复使用来自网站的不符合生产质量要求的示例代码，以及在应对不断演变的软件威胁方面普遍缺乏全面的专业知识，这通常会安全漏洞渗透到代码库中。这些漏洞往往在投入生产之后才发现，那时已经造成了巨大的客户影响、业务损失和补救成本。

工程师每周大约有

19.6 小时
的专注时间，而每周要花费

10.9 小时
开会²

90%

的软件工程经理将专注时间与
工作效率关联起来²

77%

的软件工程经理认为专注时间与
业务收入的增长之间存在联系²

专注时间：定义为不少于 2 小时的、
不间断地专注于工作的时间²

生成式人工智能为构建者提供大力支持

生成式人工智能（AI）服务及其底层基础模型（FM）（这是在非常大的数据集上训练的神经网络模型，由数十亿个参数组成）拥有巨大的潜力，可以缓解软件开发组织面临的大部分压力。生成式人工智能可用于创建不同类型的原创内容，例如音乐、图像甚至软件代码，但这要依靠基础模型。云计算使得人工智能模型的复杂性和先进程度实现指数级增长成为可能。在最近的模型中，参数数量已从 2019 年的 3.3 亿，增加到 2023 年的 5400 亿（增长了 1600 倍）。因此，现在能够构建像 **Amazon CodeWhisperer** 这样的新服务，根据集成式开发环境（IDE）中的直白英语注释生成实时代码建议。

如今，生成式人工智能为软件组织带来了巨大的实际价值。这样在开发团队中，开发人员可以将复杂、耗时且无差别的繁重工作转交给人工智能，从而增强开发人员的能力，并加快日常工作流的速度。因此，开发人员可以将时间专注于创造新的颠覆性经济机会，帮助企业在竞争中脱颖而出，以更快的速度和更低的成本实现其业务目标。

未来，预计生成式人工智能还将改变企业构建和交付软件的方式。例如 IDC 报告称，公司如果利用生成式人工智能和机器学习（ML）等智能开发人员技术，加速软件或其他数字化解决方案（例如代码生成）开发，将在 2021 年至 2026 年期间实现 31.3% 的复合年增长率，总收入从 15.6 亿美元增长到 61 亿美元。³ 具有前瞻性思维的开发团队，为了加快创新的速度，正在迅速采用这些技术，以便在将来更好地为客户提供服务。

“20 多年来，亚马逊一直专注于人工智能和机器学习研究。我们在亚马逊有成千上万的工程师致力于机器学习研究，这是我们过往传统、当前宗旨和未来工作的重要组成部分。”

Swami Sivasubramanian, Amazon Web Services (AWS) 的 VP of Data & Machine Learning



人工智能编程助手 CodeWhisperer

CodeWhisperer 使用数十亿行亚马逊代码和开源代码进行了训练，是一款实时的生成式人工智能编程助手，可根据现有代码和注释生成代码建议，从而协助您快速编写安全的代码。例如，您可以使用自然语言（需使用英语）撰写代码注释，然后 CodeWhisperer 会自动加密并将注释提示（例如，解析 CSV 文件以提取第二个字段并按降序排序）和其他信息（例如，相邻的代码片段、注释、光标位置以及打开文件中的内容）发送到后端服务作为输入信息，用于提供代码建议。然后，CodeWhisperer 直接在开发人员的 IDE 代码编辑器中提供代码建议。除了提供代码建议，CodeWhisperer 还可以识别常见的代码漏洞，并在建议中标记对开源代码的引用。与当前的代码编写 workflow 一样，开发人员拥有他们编写的代码，包括 CodeWhisperer 提供的任何代码建议。

CodeWhisperer 可用于所有流行的开发环境（可能是原生环境，也可能是通过 Amazon Toolkit IDE 扩展），并且可以为 15 种编程语言提供建议。

```
J SQSIdentityFunction.java M src/main/java/com/amazonaws/services/sqs/SQSIdentityFunction.java
package com.amazonaws.services.sqs;
import java.util.Map.Entry;
import java.util.stream.Collectors;
import com.amazonaws.services.lambda.runtime.Context;
import com.amazonaws.services.lambda.runtime.RequestHandler;
import com.amazonaws.services.lambda.runtime.events.SQSEvent;
import com.amazonaws.services.lambda.runtime.events.SQSEvent.MessageAttribute;
import com.amazonaws.services.lambda.runtime.events.SQSEvent.SQSMessage;
import com.amazonaws.services.sqs.model.MessageAttributeValue;
import com.amazonaws.services.dynamodbv2.AmazonDynamoDBClientBuilder;
import com.amazonaws.regions.Regions;

/*Create a lambda function that stores the body of the SQS message
into a hash key of a DynamoDB table. */
public class SQSIdentityFunction implements RequestHandler<SQSEvent, String> {

    private static final String TABLE_NAME = "SQSMessage";
    private static final String HASH_KEY = "MessageId";

    private AmazonDynamoDBClientBuilder builder = AmazonDynamoDBClientBuilder.standard();
    private AmazonDynamoDB client = builder.withRegion(Regions.US_EAST_1).build();

    @Override
    public String handleRequest(SQSEvent event, Context context) {
        for (SQSMessage message : event.getRecords()) {
            String messageId = message.getMessageId();
            String messageBody = message.getBody();
            context.getLogger().log("MessageId: " + messageId + " MessageBody: " + messageBody);
            storeMessage(messageId, messageBody);
        }
        return "Success";
    }

    private void storeMessage(String messageId, String messageBody) {
        client.putItem(TABLE_NAME, HASH_KEY, messageId, messageBody);
    }
}
```

在亚马逊生产力挑战赛中，使用 Amazon CodeWhisperer 的参与者

27%

完成任务的可能性提升

57%

相比未使用 CodeWhisperer
的参与者的速度提升

CodeWhisperer 的主要功能

CodeWhisperer 在设计时考虑了现代开发人员的环境。该服务融入了开发人员工作流程中不可或缺的部分，例如安全性、尊重开源原则、广泛支持各种工具和技术以及满足企业需求。

CodeWhisperer 的主要功能包括：

- **实时代码建议**：CodeWhisperer 可以建议单行代码、更大的多行代码块，甚至是整个函数，其风格与开发者的个人代码编写风格和变量命名约定相一致。然后，开发人员可以选择接受或编辑头条建议、查看更多建议或者继续编写自己的代码。
- **负责任的人工智能**：CodeWhisperer 有一个引用跟踪器，用于进行检测，以发现生成的代码建议可能与用作训练数据的特定开源代码相似的情况。然后，根据开发人员的偏好，该跟踪器可以将这些代码从建议列表中筛选出来，也可以使用对存储库的引用和许可信息进行注释，以便开发人员可以正确使用引用的开源代码和注明出处。CodeWhisperer 还可以帮助开发人员筛选掉可能有偏见、侮辱性或包含个人身份信息的代码建议。
- **内置安全扫描**：CodeWhisperer 提供“内置”安全扫描功能，有助于发现难以检测的漏洞，例如十大全球开放应用程序安全项目（OWASP，Open Worldwide Application Security Project）中的漏洞或亚马逊云科技内部安全最佳实践中强调的漏洞。该功能还提供有关这些漏洞补救措施的建议。安全扫描可用于 Java、JavaScript 和 Python 代码。
- **针对亚马逊云科技进行优化**：CodeWhisperer 可提供一般代码建议，而且还针对编写与亚马逊云科技服务交互的代码进行了专门的优化，例如与 [Amazon Elastic Compute Cloud](#)（Amazon EC2）、[Amazon Lambda](#) 或 [Amazon Simple](#)

Storage Service（Amazon S3）交互。例如，该服务可以接受“从 Amazon S3 存储桶中获取过去 24 小时内使用 Amazon Lambda 上传的新文件”之类的提示，并提供高质量的代码建议，以便使用亚马逊云科技最佳实践调用相应的 API。对于那些在亚马逊云科技上开展广泛构建工作的人来说，这是一个理想的编程助手。

- **支持流行的编程语言**：CodeWhisperer 支持 Python、Java、JavaScript、TypeScript、C#、Go、Rust、PHP、Ruby、Kotlin、C、C++、Shell 脚本、SQL 和 Scala。



- **支持流行的开发环境**：CodeWhisperer 可用作 Visual Studio (VS) Code、JetBrains IDE (IntelliJ IDEA、PyCharm、CLion、GoLand、WebStorm、Rider、PhpStorm、RubyMine、DataGrip)、JupyterLab 和 [Amazon SageMaker Studio](#) 的扩展。该服务在 [Amazon Cloud9](#)、Lambda 控制台和 [Amazon Glue Studio](#) 中原生提供。



- **企业管理**：CodeWhisperer 可以使用群组范围的策略来集中管理访问权限（例如单点登录）或功能激活（例如自动筛选与开源培训代码相似的代码建议）。

为什么选择 CodeWhisperer ?



速度

能够快速完成基本的代码编写任务，加快新技术的采用并缩短应用程序开发周期。



更少的漏洞

能够更快地发现和修复软件漏洞，以增强应用程序安全性，同时减少应用程序开发的成本、时间和风险。



降低风险

能够识别从网络上复制源代码片段而不注明出处情况，降低可能发生的侵犯开源版权的风险。引用跟踪器可检测建议的代码片段与开源代码的相似性，并提供指向相关存储库和许可证的链接，以便您相应地做出决定和管理。



热门工具支持

能够与您偏好的编程语言和开发环境配合使用，而无需更改开发工作流程和流程。



让开发人员满意

分担简单（但不断重复）或复杂（但大同小异）的代码编写任务，使得开发人员可以减少思考负担，将“专注时间”用来开发更具创新性和令人兴奋的功能。



亚马逊云科技的专业知识

借助用于整合亚马逊云科技服务的高质量代码建议，轻松快速地提升开发团队中的亚马逊云科技服务专业知识水平。

实现软件开发转型

CodeWhisperer 是一款功能强大的人工智能编程助手，有助于为任意任务编写代码，并以各式各样的方法提供协助。您可以借助以下几种方法来更快地完成更多任务。



提高应用程序安全性

运行安全扫描，在开发周期的前期识别和修复软件漏洞。



简化分析和人工智能计划

使用 JupyterLab、Amazon Glue Studio 或 **Amazon SageMaker** 快速开发代码，用于探索、准备和处理用于机器学习或分析的数据。



加快应用程序开发

快速获取耗时但简单的代码编写任务的代码建议，例如处理输入文件中的字符串并存储到数据库中。



快速探索和采用新技术

利用代码建议，快速探索、学习和使用不熟悉的或新的 SDK 和 API 来完成复杂的任务。



构建下一代应用程序

借助实施了亚马逊云科技最佳实践的高质量代码建议，简单快速地采用亚马逊云科技服务，从而更轻松地构建复杂、创新和强大的云应用程序。



提高软件质量

生成单元测试代码，从而采用测试驱动型开发方法，并提高应用程序代码质量。



BUILDSTR 是一家云产品开发和平台现代化咨询公司。作为亚马逊云科技精选级服务合作伙伴，BUILDSTR 主要使用 Node.js、Python 和 React 为客户构建解决方案。要想赢得客户的信任，快速的开发周期、客户响应能力和应用程序安全性对于咨询解决方案提供商都至关重要。在将 **Amazon CodeWhisperer** 纳入其软件开发 workflow 后，BUILDSTR 的快速原型设计速度提高了 40%，客户部署的代码中的软件漏洞减少了 50%。

“由于咨询公司永远都在不停地招聘和培养新人才，因此我们利用 CodeWhisperer 来协助新开发人员入职上岗，该服务可促使 BUILDSTR Academy 将入职上岗的时间和复杂性减少 20% 以上 ... 我们为 Amazon Lambda 函数创建模板来与 Amazon DynamoDB 等其他亚马逊云科技服务进行交互，其快捷程度让我们惊叹不已。”

Kyle Hines, BUILDSTR 的 CRO

提高开发人员工作效率

20%+
开发者入职
上岗速度提升

50%
软件漏洞减少

40%
原型设计
速度提升





Accenture 是一家全球领先的专业服务公司，提供一流的战略和技术咨询服务。Accenture 依靠其 Velocity 云平台，推动客户降低在云中构建和运营企业应用程序的复杂性和成本。Velocity 开发团队整合了 **Amazon CodeWhisperer**，对其软件开发实践进行了现代化改造，并加快实施生成式人工智能和机器学习项目。借助 CodeWhisperer，团队中的新开发人员可以很快地开始高效工作，而现有开发人员可以减少花在重复性样板代码或在网络上寻找代码样本的时间，从而加快新代码和安全应用程序的开发。

“为了在我们的 Velocity 平台实施软件工程最佳实践，Accenture 目前使用 Amazon CodeWhisperer 来加快代码编写速度。Velocity 团队目前正在寻找提高开发人员工作效率的方法。在研究了多种方案之后，我们找到了 Amazon CodeWhisperer，该服务让我们的开发工作量得以减少 30%，现在我们可以将更多精力集中在提高安全性、质量和性能方面。”

Balakrishnan Viswanathan, Accenture Tech Architecture 部门的 Senior Manager

其他客户评价



“TCS 在我们的 Financial Spreading 解决方案上利用了 CodeWhisperer，可以满足整个 BFS 领域中信用风险官员的需求。我们看到，在 Python 和 NoSQL 中，每用户的生产效率平均提高了 20%，总体部署速度提高了 25%。我们认为，作为 BFSI 客户云转型之旅的一部分，CodeWhisperer 在加速应用程序现代化计划方面具有巨大的潜力。”

Pallavi Rao, Amazon Web Services Business Unit 的
Global Delivery Head - BFSI



“我们开始使用 CodeWhisperer 向开发人员提供人工智能驱动的编程助手。该服务有助于提高速度和快速应变能力，特别是面对诸如编写单元测试和编写无差别代码之类的任务时。该服务有助于创建更快的环境，让初级开发人员更快速高效地在新岗位上发挥作用。借助 CodeWhisperer，我们的开发人员可以专注于高业务优先级的任务，将重复性和单调的繁重任务交由 CodeWhisperer 处理，从而将整体生产率提高了 25-30%。”

Oren Sokoler, Aurora Labs 的 CTO



“Amazon Ads 每天都在使用 CodeWhisperer 来提高工作效率。团队中的 Python 开发人员发现，在 CodeWhisperer 的帮助下，编写 Java 代码几乎和编写 Python 一样流畅，而且该服务对两个应用场景特别有用：在编写单元测试时，该服务可以将所需时间缩短将近一半；另外就是在处理深奥的 CDK 问题时，让开发人员不必查找大量文档。使用 CodeWhisperer，Amazon Ads 得以将代码编写的速度提高 25-35%。”

Amazon Ads SSPA/ADAM/Atlas/Sword 的 Software Engineer



“Twilio 已准备好在 Amazon CodeWhisperer 的支持下，迎接大幅提升工程设计工作效率的未来。通过利用其变革性的功能，工程师将飞速提高效率水平，克服障碍，减少错误。Twilio 与亚马逊云科技一起踏上了效率之旅，现在工程师们可以专注于创新，而 CodeWhisperer 则负责其余工作，使他们能够充满信心地开拓新领域。”

Juan E. Riojas, Twilio 的 CIO

立即开始构建

开发团队可以通过两种方式使用 CodeWhisperer，更快、更安全地开始构建应用程序：

- **CodeWhisperer 个人套餐**：通过**个人套餐**，开发人员可以使用自己的电子邮件地址创建亚马逊云科技构建者 ID，在短短几分钟内开始免费使用 CodeWhisperer。CodeWhisperer 个人套餐提供代码建议、引用跟踪和安全扫描。该服务可能会存储和使用您的内容来改进服务；不过，您可以选择退出此数据收集。
- **CodeWhisperer 专业套餐**：**专业套餐**专为开发组织设计，使用 **Amazon IAM Identity Center**（Amazon Single Sign-On 的后代产品）提供用户访问。专业套餐以个人套餐为基础，具有额外的管理功能和增强的使用限制。团队支付“每用户每月”的即用即付订阅费。CodeWhisperer 专业套餐不会存储或使用您的内容来改善服务。

资源

（一些资源仅提供英文版。）

- [通过 Amazon CodeWhisperer 使用生成式人工智能开发应用程序的指南](#)
- 详细了解[亚马逊云科技上的生成式人工智能](#)
- [Amazon CodeWhisperer 定价](#)
- 文档：[使用 JupyterLab 设置 CodeWhisperer](#)
- 视频：[Amazon CodeWhisperer 概述](#)
- 视频：[在 VS Code 中开始使用 Amazon CodeWhisperer](#)
- 视频：[在 IntelliJ 中开始使用 Amazon CodeWhisperer](#)
- 视频：[使用 Amazon CodeWhisperer 构建 Python 事件驱动型无服务器应用程序](#)
- 文档：[CodeWhisperer 入门](#)
- [数据隐私中心](#)