

CodeArts 盘古助手

快速入门

文档版本 01
发布日期 2025-01-20



版权所有 © 华为云计算技术有限公司 2025。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

商标声明



HUAWEI和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为云计算技术有限公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为云计算技术有限公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

华为云计算技术有限公司

地址：贵州省贵安新区黔中大道交兴功路华为云数据中心 邮编：550029

网址：<https://www.huaweicloud.com/>

目录

1 使用 CodeArts 盘古助手生成代码及对应单元测试.....	1
------------------------------------	---


1 使用 CodeArts 盘古助手生成代码及对应单元测试

示例将通过CodeArts 盘古助手智能问答生成一段Java冒泡排序代码，并对生成的代码生成对应单元测试用例。

前提条件

- 下载并安装JDK1.8或更高版本。
- 下载并安装IntelliJ IDEA 2021.3或更高版本。
- 下载并安装盘古助手（CodeArts 盘古助手），华为云底座插件（Huawei Cloud Toolkit Platform）。安装插件请参考：[在JetBrains系列IDE中安装CodeArts 盘古助手](#)
- 进入[CodeArts 盘古助手邀测申请说明页面](#)，单击“申请试用”按钮，等待CodeArts 盘古助手服务支撑授权试用。
- 已登录Huawei Cloud Toolkit Platform，登录Huawei Cloud Toolkit Platform请参考：[IntelliJ IDEA平台登录Huawei Cloud Toolkit Platform](#)

通过智能问答生成 Java 冒泡排序代码

步骤1 在CodeArts 盘古助手的研发对话窗口中输入问题，“生成一个Java的冒泡排序算法”，单击发送按钮.

步骤2 盘古助手将生成一段Java编写的冒泡排序代码及一个main函数，智能开发助手可能生成的代码如下。

```
public static void bubbleSort(int[] array) {
    int n = array.length;
    for (int i = 0; i < n-1; i++) {
        for (int j = 0; j < n-i-1; j++) {
            if (array[j] > array[j+1]) {
                // swap array[j] and array[j+1]
                int temp = array[j];
                array[j] = array[j+1];
                array[j+1] = temp;
            }
        }
    }
}
```


```
public static void main(String[] args) {
    int[] array = {64, 34, 25, 12, 22, 11, 90};
    bubbleSort(array);
    System.out.println("Sorted array");
    for(int i=0; i < array.length; i++){
        System.out.print(array[i] + " ");
    }
}
```

步骤3 单击回答卡片右上角“+”按钮，生成的代码另存为“BubbleSort.java”文件并添加必要的package语句。在IDEA中选中“BubbleSort.java”文件，单击右键选择菜单“Run 'BubbleSort.main()'”运行代码，控制台打印排序后数组。

----结束

对冒泡排序代码生成测试用例

步骤1 选中“BubbleSort.java”中“bubbleSort”方法的代码，使用快捷键Ctrl+Shift+Q将代码添加至CodeArts 盘古助手的研发对话窗口。

步骤2 在研发对话窗口输入框输入“/”后，在弹出的菜单中单击选择“/test”，单击发送按钮。

步骤3 CodeArts 盘古助手将对bubbleSort方法生成相关测试用例，单击回答卡片的“+”按钮将代码另存为“BubbleSortTest.java”。

```
import org.example.BubbleSort;
import org.junit.jupiter.api.Test;
import static org.junit.jupiter.api.Assertions.*;

public class BubbleSortTest{
    @Test
    void testBubbleSort(){
        int[] array={64,34,25,12,22,11,90};
        BubbleSort.bubbleSort(array);
        assertEquals(new int[]{11,12,22,25,34,64,90}, array);
    }
    @Test
    void testBubbleSortEmpty(){
        int[] array={};
        BubbleSort.bubbleSort(array);
        assertEquals(new int[] {}, array);
    }
    @Test
    void testBubbleSortSingleElement(){
        int[] array={5};
        BubbleSort.bubbleSort(array);
        assertEquals(new int[]{5}, array);
    }
    @Test
    void testBubbleSortAlreadySorted(){
        int[] array={1,2,3,4,5};
        BubbleSort.bubbleSort(array);
        assertEquals(new int[]{1,2,3,4,5}, array);
    }
    @Test
    void testBubbleSortReversed(){
        int[] array={5,4,3,2,1};
        BubbleSort.bubbleSort(array);
        assertEquals(new int[]{1,2,3,4,5}, array);
    }
}
```

```
}  
}
```

----结束

通过以上操作，开发者不仅成功地编写了一个函数。同时，开发者还输出了相应的测试用例，以验证函数的正确性和完整性。