

技术学习

sully

2025 年 2 月 4 日

目录

目录	1
1 引言	2
2 Java 代码示例	2
3 图片插入示例	2
4 引言	3
5 插入 C++ 代码	6

1 引言

这里是文章的引言部分。

2 Java 代码示例

以下是一个嵌入的 Java 代码示例：

```
public class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Hello, world!");
    }
}
```

Listing 1: HelloWorld.java

3 图片插入示例

下面是一个插入图片的示例：

1

这是什么？

这是 Q-book L^AT_EX 书籍模板，当前版本为 v2.01。

这份模板主要基于上海交大的学位论文模板¹修改得到，结合少量个人审美喜好，重新定制了定义、定理等环境。由于这个模板本是用于一项书籍翻译计划，因此其中的一些环境，例如“观察”、“规则”、“关键点”等，读者可能用不到，可以根据自己的需求适当修改。

你也可以通过邮箱 je74165@163.com 给我发邮件反映遇到的问题。不过作者水平有限，或许有些问题也无法解答，还请见谅。

1.1 文档说明

1.1.1 准备工作

- 要想灵活使用、魔改这个模板来撰写自己的书籍，需要对 *TeX* 系统有一定的了解，也需要掌握基本的 *TeX* 技能。
- *TeX* 系统：所使用的 *TeX* 系统要支持 Xe_{La}TeX 引擎，且带有 ctex 2.x 宏包。一般来说，只要安装了完整 TeXLive 或 MacTeX 发行版就不会出现问题。
 - *TeX* 技能：尽管提供了对模板的必要说明，但这并不是一份“L^AT_EX 入门文档”。用户应当有一定的 L^AT_EX 使用经验。

¹<https://github.com/sjtug/SJTUThesis>

图 1: 示例图片

4 引言

这里是文章的引言部分。

```

class Solution {
    public int[] twoSum(int[] nums, int target) {
        int[] aaa = { 7, 5, 3, 2, 8, 0, 1 };

        System.out.println("冒泡排序");
        for (int i = 0; i < aaa.length; i++) {
            for (int j = 0; j < aaa.length - i - 1; j++) {
                int tmp = aaa[j + 1];
                if (aaa[j] > aaa[j + 1]) {
                    aaa[j + 1] = aaa[j];
                    aaa[j] = tmp;
                }
            }
        }

        for (int i = 0; i < aaa.length; i++) {
            System.out.print(aaa[i] + ",");
        }

        System.out.println("");
        System.out.println("插入排序");
        int[] bbb = { 7, 5, 3, 2, 8, 0, 1 };
        for (int i = 1; i < bbb.length; i++) {
            int tmpV = bbb[i];
            int j = i - 1;

            /* 方法一，将比 key 大的元素向右移动 */
            /*
            while (j >= 0 && bbb[j] > tmpV) {
                bbb[j + 1] = bbb[j];
                j = j - 1;
            }
            bbb[j + 1] = tmpV;
            */
            //方法二
            for (; j >= 0; j--) {
                if (bbb[j] > tmpV) {
                    bbb[j + 1] = bbb[j];
                } else {
                    break;
                }
            }
            bbb[j + 1] = tmpV;
        }

        for (int i = 0; i < bbb.length; i++) {
            System.out.print(bbb[i] + ",");
        }

        System.out.println("two_sum");
        Map <Integer,Integer> myMap = new HashMap
        <>();
        for(int i=0;i<nums.length;i++){
            if(myMap.containsKey(target-nums[i])) {
                return new int []{myMap.get(target-
                nums[i]),i};
            }
            myMap.put(nums[i],i);
        }
        return new int [0];
    }
}

```

```

    }
}

def twoSum(self, nums: List[int], target: int) ->
List[int]:
    n = len(nums)
    for i in range(n - 1):
        for j in range(i + 1, n):
            if nums[i] + nums[j] == target:
                return [i, j]
    return [] # No solution found

class Solution:
    def twoSum(self, nums: List[int], target: int)
    -> List[int]:
        numMap = {}
        n = len(nums)

        for i in range(n):
            complement = target - nums[i]
            if complement in numMap:
                return [numMap[complement], i]
            numMap[nums[i]] = i

        return [] # No solution found

go

func twoSum(nums []int, target int) []int {
    // Hash table to store number->index mapping
    ht := make(map[int]int)

    // Iterate through the array
    for i, num := range nums {
        // Check if complement exists in hash table
        if j, exists := ht[target-num]; exists {
            // If found, return indices of both
            numbers
            return []int{j, i}
        }

        // Store current number and its index in
        hash table
        ht[num] = i
    }

    // Return empty slice if no solution found
    return []int{}
}

javascript

var twoSum = function(nums, target) {
    let map = new Map();

    for (let i=0; i<nums.length; i++){
        if(map.has(target-nums[i])){
            return [map.get(target-nums[i]), i]
        }else{
            map.set(nums[i], i)
        }
    }
};

```

```
//c++
class Solution {
public:
    vector<int> twoSum(vector<int>& nums, int target) {
        unordered_map<int, int> numMap;
        int n = nums.size();

        for (int i = 0; i < n; i++) {
            int complement = target - nums[i];
            if (numMap.count(complement)) {
                return {numMap[complement], i};
            }
            numMap[nums[i]] = i;
        }

        return {}; // No solution found
    }
};

N.283 移动零问题
public void moveZeroes(int[] nums) {
    int j = 0;
    for (int i = 0; i < nums.length; i++) {
        if (nums[i] != 0) {
            nums[j] = nums[i];
            if (i != j) {
                nums[i] = 0;
            }
            j++;
        }
    }
}
```

5 插入 C++ 代码

本文展示了如何嵌入 Java 代码和图片，同时提供了目录、章节标题等结构。