# [买卖股票的最佳时机](https://leetcode-cn.com/explore/interview/card/bytedance/246/dynamic-programming-or-greedy/1042/)

**头条重点**

## 题目

给定一个数组，它的第 i 个元素是一支给定股票第 i 天的价格。如果你最多只允许完成一笔交易（即买入和卖出一支股票），设计一个算法来计算你所能获取的最大利润。

注意你不能在买入股票前卖出股票。

示例 1:

输入: [7,1,5,3,6,4]
输出: 5
解释: 在第 2 天（股票价格 = 1）的时候买入，在第 5 天（股票价格 = 6）的时候卖出，最大利润 = 6-1 = 5 。
 注意利润不能是 7-1 = 6, 因为卖出价格需要大于买入价格。
示例 2:

输入: [7,6,4,3,1]
输出: 0
解释: 在这种情况下, 没有交易完成, 所以最大利润为 0。

## 解题思路

1. 要先买入才能卖出，先找最低价格点
2. 再找最低价格之后的最高价格，用 res 表示最大利润

public int maxProfit(int[] prices) {
 if (prices.length <= 1) {
 return 0;
 }

 int res = 0;
 int minBuy = prices[0];
 for (int price : prices) {
 res = Math.max(res, price - minBuy);
 minBuy = Math.min(minBuy, price);
 }

 return res;
}