



Dash



Articles



Videos



Problems

&lt;&lt; Prev

Next &gt;&gt;

&lt;/&gt; Problem

Editorial

Submissions

## Stock Buy and Sell – Max one Transaction Allowed



Difficulty: Easy

Accuracy: 49.33%

Submissions: 50K+

Points: 2

Given an array **prices[]** of length n, representing the prices of the stocks on different days. The task is to find the maximum profit possible by buying and selling the stocks on different days when at most one transaction is allowed. Here one transaction means 1 buy + 1 Sell. If it is not possible to make a profit then **return 0**.

Note: Stock must be bought before being sold.

### Examples:

**Input:** prices[] = [7, 10, 1, 3, 6, 9, 2]

**Output:** 8

**Explanation:** You can buy the stock on day 2 at price = 1 and sell it on day 5 at price = 9. Hence, the profit is 8.

**Input:** prices[] = [7, 6, 4, 3, 1]

**Output:** 0

**Explanation:** Here the prices are in decreasing order, hence if we buy any day then we cannot sell it at a greater price. Hence, the answer is 0.

Java (1.8)

Average Time: 10m

Your Time: 8m 42s



```
1 // } Driver Code Ends
27
28
29 // User function Template for Java
30
31 class Solution {
32     public int maximumProfit(int prices[]) {
33         if(prices==null || prices.length==0){
34             return 0;
35         }
36         int minPrice = Integer.MAX_VALUE;
37         int maxProfit = 0;
38         for(int price:prices){
39             minPrice = Math.min(minPrice,price);
40             int profit = price - minPrice;
41             maxProfit = Math.max(maxProfit,profit);
42         }
43         return maxProfit;
44     }
45 }
```



Custom Input

Compile &amp; Run

Submit