**DP-S007 (E026)**

Solved Challenges 0/2

Back To Challenges List

## Stock Buy & Sell Once - Maximum Profit

**ID:10852** **Solved By 636 Users**

The program must accept the stock prices on **N** days as the input. A person can buy a stock on a particular day and he can sell it once on any other given day. He can not buy and sell on the same day. The program must print the maximum possible profit **P** that can be obtained by buying and selling 1 stock once as the output.

**Boundary Condition(s):**
2 <= N <= 10^5
1 <= Each integer value <= 10^5

**Input Format:**
The first line contains N.
The second line contains N integers separated by a space.

**Output Format:**
The first line contains P.

**Example Input/Output 1:**
Input:
7
50 100 40 60 70 50 80

Output:
50

Explanation:
Here N = **7**.
The stock price on the **1st** day is **50** and the stock price on the **2nd** day is **100**.
On buying the stock on the 1st day and selling it on the 2nd day can earn the maximum profit **50** (100 - 50 = 50).
Hence the output is 50

**Example Input/Output 2:**
Input:
10
15 10 60 70 45 5 70 30 100 90

Output:
95

**Max Execution Time Limit: 500 millisecs**

Ambiance

Python3 (3.x )

Reset

```
N = int(input())
prices = list(map(int, input().strip().split()))

minPrice = prices[0]
maxProfit = 0

for index in range(1,N):
    if(prices[index]<minPrice):
        minPrice = prices[index]
    else:
        profit = prices[index] - minPrice
        if(profit>maxProfit):
            maxProfit = profit

print(maxProfit)
```

Code did not pass the execution

TestCase ID: 60942

Input:

**7**
**50 100 40 60 70 50 80**

Expected Output:

**50**

Your Program Output:

100

Save Run

Run with a custom test case (Input/Output)