Description Solution Discuss (999+) Submissions

121. Best Time to Buy and Sell Stock

Easy 24259 758 Add to List Share

You are given an array `prices` where `prices[i]` is the price of a given stock on the i^{th} day.

You want to maximize your profit by choosing a **single day** to buy one stock and choosing a **different day in the future** to sell that stock.

Return *the maximum profit you can achieve from this transaction.* If you cannot achieve any profit, return `0`.

Example 1:

```
Input: prices = [7,1,5,3,6,4]
Output: 5
Explanation: Buy on day 2 (price = 1) and sell on day 5 (price = 6), profit = 6-1 = 5.
Note that buying on day 2 and selling on day 1 is not allowed because you must buy before you sell.
```

Example 2:

```
Input: prices = [7,6,4,3,1]
Output: 0
Explanation: In this case, no transactions are done and the max profit = 0.
```

Constraints:

- `1 <= prices.length <= 10^5`
- `0 <= prices[i] <= 10^4`

Accepted 3,189,005 Submissions 5,869,332

Seen this question in a real interview before? Yes No

Companies

Related Topics

Similar Questions

C++ Autocomplete

```cpp
class Solution {
public:
    int minP[100000];
    int maxProfit(vector<int>& prices) {
        int ans=0;
        int n=prices.size();
        minP[0]=prices[0];
        for(int i=1; i<n; i++)
            minP[i]=min(prices[i],minP[i-1]);
        for(int i=1; i<n; i++){
            int curAns=prices[i]-minP[i-1];
            if(ans<curAns) ans=curAns;
        }
        return ans;
    }
};
```

NEW

Your previous code was restored from your local storage. Reset to default