122. Best Time to Buy and Sell Stock II

Easy.
Greedy,Array.

Say you have an array for which the ith element is the price of a given stock on day i.

Design an algorithm to find the maximum profit. You may complete as many transactions as you like (i.e., buy one and sell one share of the stock multiple times).

Note: You may not engage in multiple transactions at the same time (i.e., you must sell the stock before you buy again).

Example 1:

```
Input: [7,1,5,3,6,4]
Output: 7
Explanation: Buy on day 2 (price = 1) and sell on day 3 (price = 5), profit = 5-1 = 4.
             Then buy on day 4 (price = 3) and sell on day 5 (price = 6), profit = 6-3 = 3.
```

Example 2:

```
Input: [1,2,3,4,5]
Output: 4
Explanation: Buy on day 1 (price = 1) and sell on day 5 (price = 5), profit = 5-1 = 4.
             Note that you cannot buy on day 1, buy on day 2 and sell them later, as you are
             engaging multiple transactions at the same time. You must sell before buying
again.
```

Example 3:

```
Input: [7,6,4,3,1]
Output: 0
Explanation: In this case, no transaction is done, i.e. max profit = 0.
```

解法1

复杂度为O(n)
可以使用动态规划来解这道题。

Java

```
class Solution {
    public int maxProfit(int[] prices) {

        int len = prices.length;
        if(len<=1) return 0;
        int[] dp = new int[len];
        for(int i=0; i<len; i++){
            dp[i]=0;
        }
        for(int i=1; i<len; i++){
            if(prices[i]-prices[i-1]>0) dp[i]=dp[i-1]+prices[i]-prices[i-1];
            else dp[i]=dp[i-1];
        }
        return dp[len-1];
    }
}
```