121. Best Time to Buy and Sell Stock

Easy.
Array, Dynamic programming

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

If you were only permitted to complete at most one transaction (i.e., buy one and sell one share of the stock), design an algorithm to find the maximum profit.

Note that you cannot sell a stock before you buy one.

Example 1:

```
Input: [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.
             Not 7-1 = 6, as selling price needs to be larger than buying price.
```

```
Example 2:

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

方法一

用数组便利的方式，用两个数组相减来查找最大的差值。时间复杂度O(n^2)。

Java

```java
class Solution {
    public int maxProfit(int[] prices) {
        int res=0;
        for(int i=0;i<prices.length-1;i++){
            for(int j=i+1;j<prices.length;j++){
```

```
                res=Math.max(prices[j]-prices[i],res);
            }
        }
        return res;
    }
}
```

方法二：
更简单的方法，只需要O(n);
这个方法中，一直在查找买入时的最小值，以及profit的最大值。

Java

```
class Solution {
    public int maxProfit(int[] prices) {
        int buy = Integer.MAX_VALUE;
        int profit = 0;
        for(int p:prices){
            buy = Math.min(p,buy);
            profit = Math.max(profit,p-buy);
        }
        return profit;
    }
}
```