[Best Time to Buy and Sell Stock II](https://leetcode.com/problems/best-time-to-buy-and-sell-stock-ii/)

Task: get the max profit by buying and selling the stocks any number of times but it should be in a sequence first buy then sell.

**Profit:** selling price - buying price; so take buying price -ve and selling price +ve.

**Recursive Approach:**  f(ind, buy)

1. On any day I should have an idea, is this turn to buy or sell the stocks so for that we will pass a parameter buy: 0/1

1-> buy; 0-> not-buy

1. If (buy==1) that means we can buy here: 2 possibilities buy and not buy

-> buy: take which gives the max profit

max(-prices[ind] + f(ind+1, 0), 0+f(ind+1, 1))

1. Else we have chance to sell it: 2 possibilities sell and not-sell

-> sell: take which gives the max profit

max(prices[ind]+ f(ind+1, 1), 0+f(ind+1, 0))

**Tabulation:** make dp table in reverse ind= n-1 to 0, buy = 0 to 1

1. Make dp[n+1][2]
2. Use base case and the recurrence relation to build dp table
3. [n+1] to take care of base case=> if(ind==n) dp[n][0] = dp[n][1] = 0
4. Ans will be stores at dp[0][1]