75 Days of Code
Day 60
Problem no : Leetcode 746
Problem Title : Min Cost Climbing Stairs
Problem type : DP

You are given an integer array cost where cost[i] is the cost of ith step on a staircase. Once you pay the cost, you can either climb one or two steps.

You can either start from the step with index 0, or the step with index 1.

Return the minimum cost to reach the top of the floor.

Example 1:

Input: cost = [10,15,20]
Output: 15
Explanation: You will start at index 1.
- Pay 15 and climb two steps to reach the top.
The total cost is 15.
Example 2:

Input: cost = [1,100,1,1,1,100,1,1,100,1]
Output: 6
Explanation: You will start at index 0.
- Pay 1 and climb two steps to reach index 2.
- Pay 1 and climb two steps to reach index 4.
- Pay 1 and climb two steps to reach index 6.

- Pay 1 and climb one step to reach index 7.
- Pay 1 and climb two steps to reach index 9.
- Pay 1 and climb one step to reach the top.
The total cost is 6.

Constraints:

2 <= cost.length <= 1000
0 <= cost[i] <= 999

```typescript
function minCostClimbingStairs(cost: number[]): number {
    let stepsWithCost= [cost[0] ,cost[1]];

    for(let step =2 ;step<cost.length ;step++){

        stepsWithCost[step] = Math.min( stepsWithCost[step-1]+cost[step] ,stepsWithCost[step-2]+cost[step] )
    }
    return Math.min (stepsWithCost[stepsWithCost.length-1] ,stepsWithCost[stepsWithCost.length-2])

};
```