75 Days of Code
Day 61
Problem no : Leetcode 198
Problem Title : House Robber
Problem type : DP

You are a professional robber planning to rob houses along a street. Each house has a certain amount of money stashed, the only constraint stopping you from robbing each of them is that adjacent houses have security systems connected and it will automatically contact the police if two adjacent houses were broken into on the same night.

Given an integer array nums representing the amount of money of each house, return the maximum amount of money you can rob tonight without alerting the police.

Example 1:

Input: nums = [1,2,3,1]
Output: 4
Explanation: Rob house 1 (money = 1) and then rob house 3 (money = 3).
Total amount you can rob = 1 + 3 = 4.
Example 2:

Input: nums = [2,7,9,3,1]
Output: 12
Explanation: Rob house 1 (money = 2), rob house 3 (money = 9) and rob house 5 (money = 1).
Total amount you can rob = 2 + 9 + 1 = 12.

```typescript

function rob(nums: number[]): number {
    if(nums.length<=2){
        return Math.max(...nums)
    }
    let  rob1 = 0;
    let  rob2 = 0;
     let decision =0
   for(let index =0;index<nums.length ;index++){
        decision  = Math.max(nums[index] +rob1 ,rob2)
        rob1 = rob2;
        rob2 = decision;
    }
     return decision;

};
```

Runtime

Details

47 ms

Beats 88.65% of users with TypeScript

Memory

42.04 MB

Beats 96.97% of users with TypeScrip