

198. House Robber

Medium

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.

```
class Solution:
    def rob(self, nums: List[int]) -> int:
        rob1, rob2 = 0, 0
        for n in nums:
            temp = max(n + rob1, rob2)
            rob1 = rob2
            rob2 = temp
        return rob2
```

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..

Q1: houses- [2, 1, 1, 2]; solution- 4

Q2: houses- [2, 7, 9, 3, 1]; solution- 12

Q3: houses- [2, 1, 6, 8, 5, 4]; solution- 14

Q4: houses- [2, 1, 8, 6, 5, 4]; solution- 15

i	0	1	2	3	4	5
n	2	1	6	8	5	4
rob2:	0	2	2	8	10	13
rob1 + n:	2	1	8	10	13	14
temp:	2	2	8	10	13	14
rob1:	0	2	2	8	10	13
rob2:	0	2	2	8	10	14

1. declare two variables that keep track of previous values
2. alternate the values between the two variables to ensure they are not adjacent