**Range Sum Query - Immutable**

class NumArray {

public:

    vector<int> prefixSum;

    NumArray(vector<int>& nums) {

        prefixSum.resize(nums.size());

        prefixSum[0] = nums[0];

        for (int i = 1; i < nums.size(); ++i) {

            prefixSum[i] = prefixSum[i - 1] + nums[i];

        }

    }

    int sumRange(int left, int right) {

        if (left == 0) {

            return prefixSum[right];

        } else {

            return prefixSum[right] - prefixSum[left - 1];

        }

    }

};

**In other way**

class NumArray {

private:

vector<int>PrefixArray;

public:

NumArray(vector<int>& nums) : PrefixArray(nums.size()+1) {

partial\_sum(nums.begin(), nums.end(), PrefixArray.begin()+1);

}

int sumRange(int left, int right) {

return PrefixArray[right+1] - PrefixArray[left];

}

};