Maximum Subarray

Given an integer array `nums`, find the contiguous subarray (containing at least one number) which has the largest sum and return *its sum*.

A **subarray** is a **contiguous** part of an array.

Example 1:

```
Input: nums = [-2,1,-3,4,-1,2,1,-5,4]

Output: 6

Explanation: [4,-1,2,1] has the largest sum = 6.
```

Example 2:

```
Input: nums = [1]

Output: 1
```

Example 3:

```
Input: nums = [5,4,-1,7,8]

Output: 23
```

```
public class Solution {
    public int MaxSubArray(int[] nums) {

        int highVal = nums[0];
        int currentVal = nums[0];

        for(int i=1;i<nums.Length;i++)
        {
            if(currentVal < 0)
            {
                currentVal = nums[i];
            }
            else
            {
                currentVal += nums[i];
            }

            if(currentVal > highVal)
            {
                highVal = currentVal;
            }
        }

        return highVal;
    }
}
```