

Java

```
class Solution {  
    public boolean hasIncreasingSubarrays(List<Integer> nums, int k) {  
  
    }  
}
```

JavaScript

```
/**  
 * @param {number[]} nums  
 * @param {number} k  
 * @return {boolean}  
 */  
var hasIncreasingSubarrays = function(nums, k) {  
  
};
```

TypeScript

```
function hasIncreasingSubarrays(nums: number[], k: number): boolean {  
  
};
```

C++

```
class Solution {
```

```
public:
    bool hasIncreasingSubarrays(vector<int>& nums, int k) {

    }
};
```

C#

```
public class Solution {
    public bool HasIncreasingSubarrays(IList<int> nums, int k) {

    }
}
```

Kotlin

```
class Solution {
    fun hasIncreasingSubarrays(nums: List<Int>, k: Int): Boolean {

    }
}
```

Go

```
func hasIncreasingSubarrays(nums []int, k int) bool {

}
```
