1984. Minimum Difference Between Highest and Lowest of K Scores

Description

You are given a **0-indexed** integer array `nums`, where `nums[i]` represents the score of the `i th` student. You are also given an integer `k`.

Pick the scores of any `k` students from the array so that the **difference** between the **highest** and the **lowest** of the `k` scores is **minimized**.

Return *the **minimum** possible difference*.

Example 1:

```
Input: nums = [90], k = 1
Output: 0
Explanation: There is one way to pick score(s) of one student:
- [ 90 ]. The difference between the highest and lowest score is 90 - 90 = 0.
The minimum possible difference is 0.
```

Example 2:

```
Input: nums = [9,4,1,7], k = 2
Output: 2
Explanation: There are six ways to pick score(s) of two students:
- [ 9 , 4 ,1,7]. The difference between the highest and lowest score is 9 - 4 = 5.
- [ 9 ,4, 1 ,7]. The difference between the highest and lowest score is 9 - 1 = 8.
- [ 9 ,4,1, 7 ]. The difference between the highest and lowest score is 9 - 7 = 2.
- [9, 4 , 1 ,7]. The difference between the highest and lowest score is 4 - 1 = 3.
- [9, 4 ,1, 7 ]. The difference between the highest and lowest score is 7 - 4 = 3.
- [9,4, 1 , 7 ]. The difference between the highest and lowest score is 7 - 1 = 6.
The minimum possible difference is 2.
```

Constraints:

- `1 <= k <= nums.length <= 1000`
- `0 <= nums[i] <= 10 5`