Question: https://leetcode.com/problems/container-with-most-water/submissions/

So what we need to do is find the container which can be formed using 2 bars, with max capacity.

If we think carefully the highest and second highest bar must give the maximum capacity but the width or the distance between them plays a role too.

So what we can do is form a container with 0th and nth rod, thus width will be max here and then compare it with other bowls by shifting to the adjacent rod of the smaller between the present 2 rods.

So like this we will get the max capacity.

ex:

[1, 8, 6, 2, 5, 4, 8, 3, 7] res ("" means the newly updated)

l r 8

l r max("49", 8)

l r max("18", 49)

l r max("40", 49)

l r max("20", 49)

l r max("15", 49)

l r max("16", 49)

l r max("8", 49)

lr --> terminate while loop, and return 49 as the result (maximal possible water volume)

Code:  
class Solution {

public int maxArea(int[] height) {

int max=Integer.MIN\_VALUE;

int s=0, e=height.length-1;

while(s<e){

max=Math.max(((e-s)\*Math.min(height[s],height[e])), max);

//idea is to shift the pointer which has lesser val

if(height[s]<height[e]) s++;

else e--;

}

return max;

}

}

Github Link :<https://lnkd.in/ecwtJeaz>