Partition Method

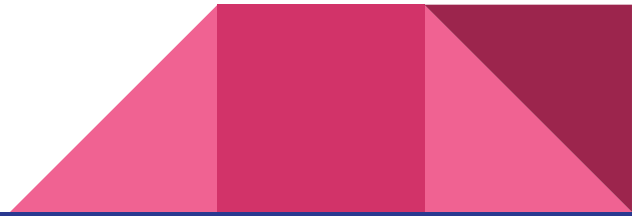
How would you solve the following problem?

Given an array of n unsorted integers, select a random element from the array and indicate the position where that element should be located if the array would be sorted. Assume, all array elements are distinct.

data = [10, 3, 6, 9, 2, 4, 15, 23]

randomElement= 6

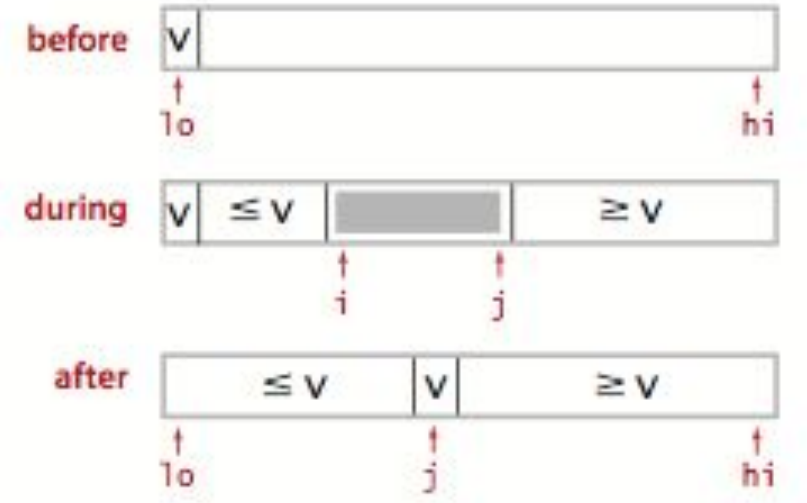
Output: 3



A possible solution: Partition Method

Choose an element from the array, we are going to call it pivot.

Then, create partitions based on the pivot

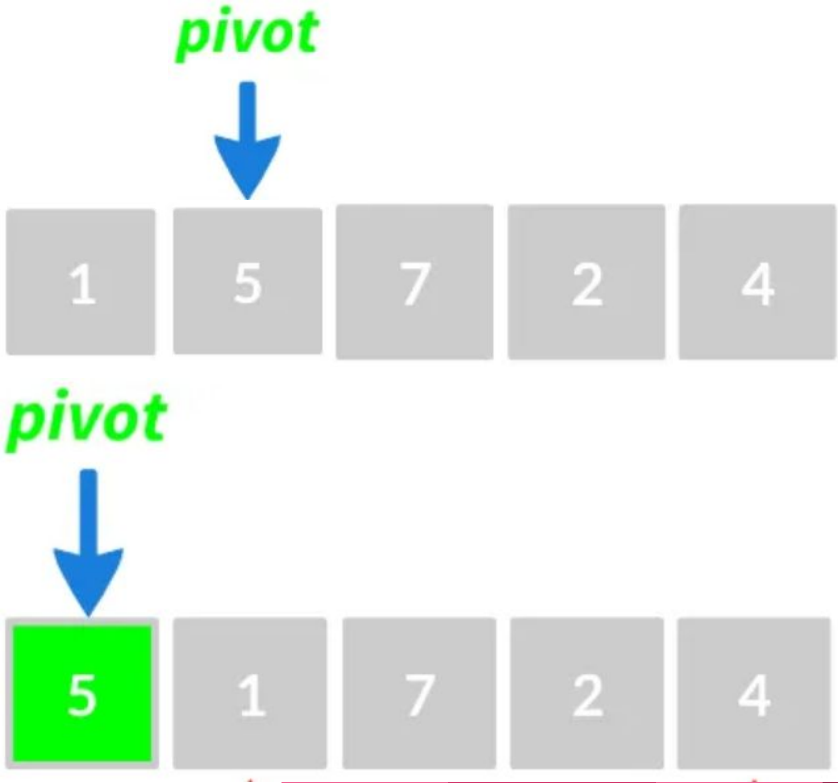


Partition Method: Step by Step

1. Create an array of integers

2. Randomly choose an index from the array
   The value at that index will be your pivot.

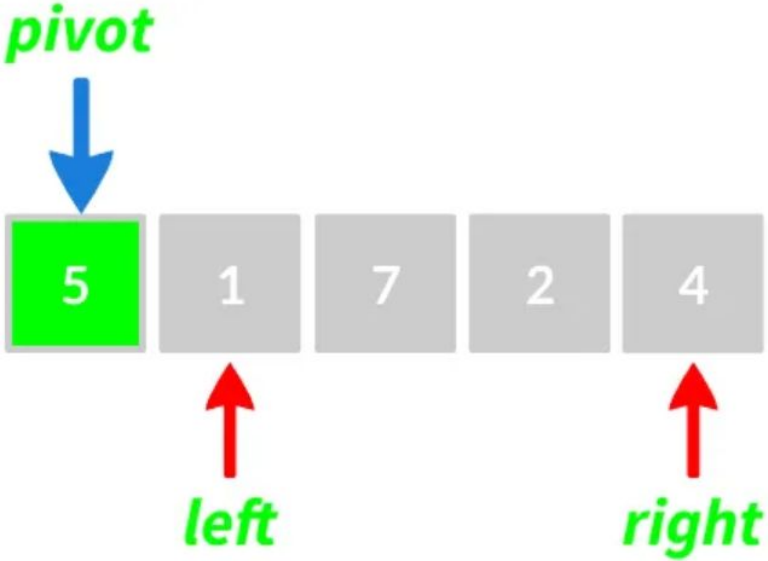
3. If the selected index is not at index zero
   Swap the pivot with the element at
   index zero.



Partition Method: Step by Step

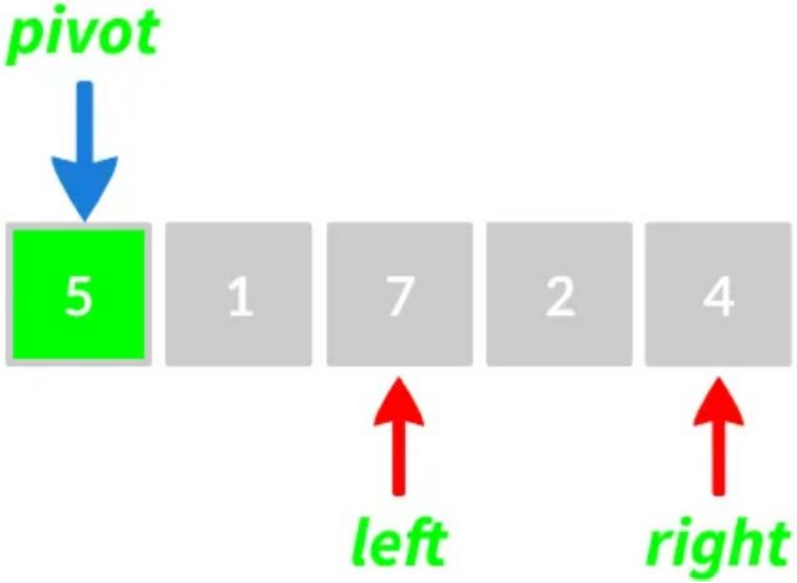
4. Set up "left" and "right" indices:

“left index” => “leftmost” element
“right index” => "rightmost” element



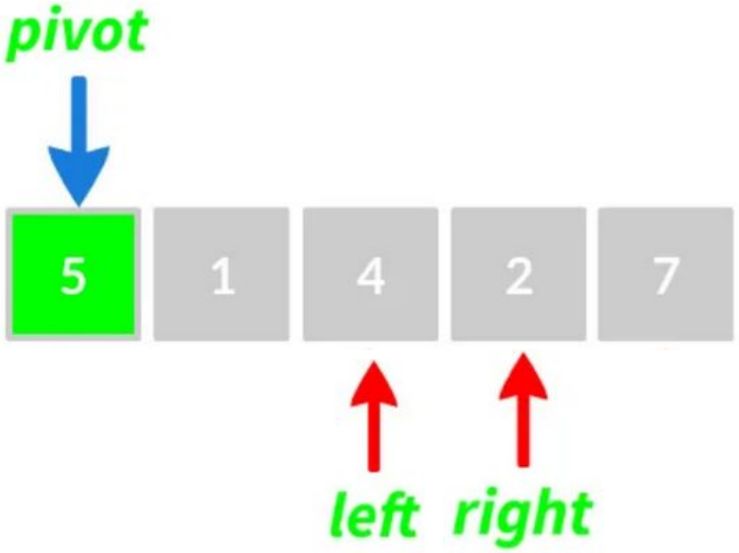
Partition Method: Step by Step

5. Move left cursor to the right



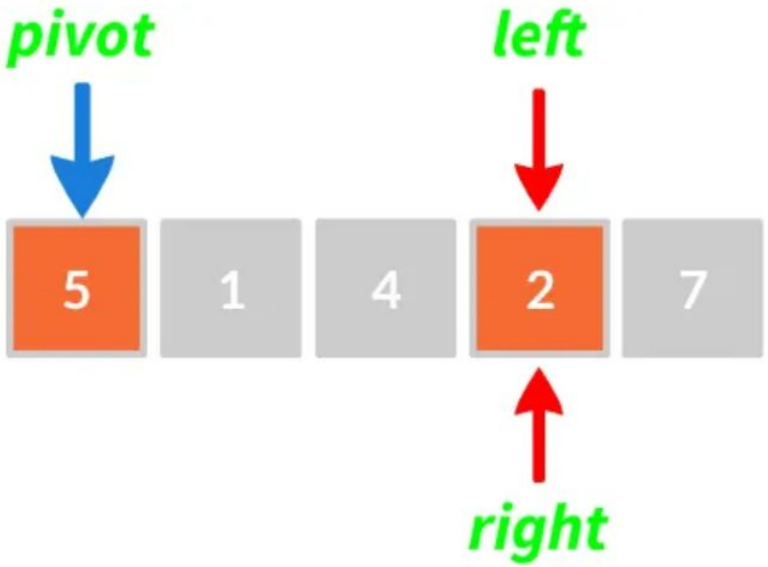
Partition Method: Step by Step

6. Swap between “right” and “left” elements



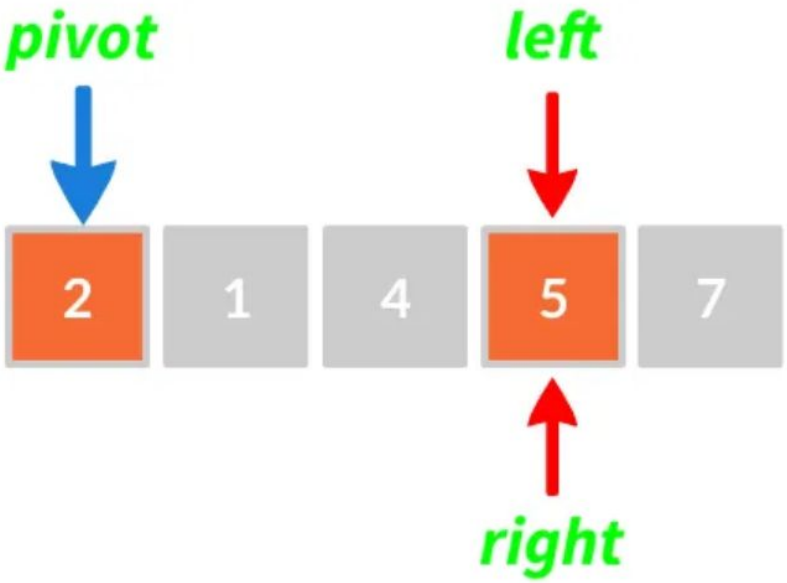
Partition Method: Step by Step

7. Move right cursor to the left.



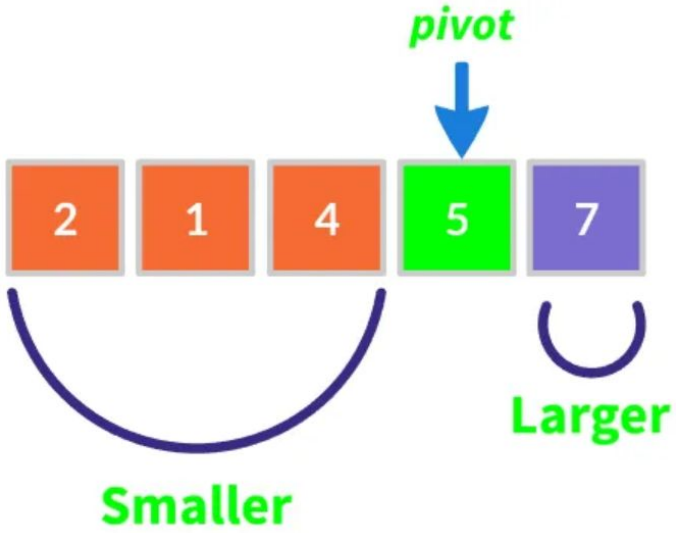
Partition Method: Step by Step

8. Swap with the pivot



Partition Method: Step by Step

9. Return the index of the final position of the pivot element => 3



Coding Time!!!!

Write your code here: classwork/xx_partition/Partition.java

You must implement the following method:

```
public static int partition( int [] data, int start, int end){
    // You must declare variables to track left and right indices
}
```

Call the method like this:

```
partition(yourArray, 0, yourArray.length-1));
```