Challenge: Implement Partition

The partition function should partition the subarray array[p..r] so that all elements in array[p..q-1] are less than or equal to array[q] (the pivot) and all elements in array[q+1..r] are greater than array[q], and it returns the index q of where the pivot ends up.

Use the provided swap() function for swapping.

Java | Python | C++ | JS

```
class Solution {
  // Swaps two items in an arra
  static void swap(int[] array
      int temp = array[firstIn
      array[firstIndex] = array
      array[secondIndex] = tem
  };

  public static void partition
      // Compare array[j] with
      // maintaining that:
      //  array[p..q-1] are va
      //  array[q..j-1] are va
      //  array[j..r-1] haven't
      // If array[j] > array[r
      // If array[j] <= array[
      //   increment q, and in
      // Once all elements in a
      //  have been compared w
      //  swap array[r] with a

  }
}
```