Last Index of Key in Array.

In this blog we will try to solve a problem. The problem is that we need to find the last index of the key in the array.

What we can do is scan the array from first and check ahead if the key exists or not. But this can be bit complicated.

We can simply run the loop in reverse manner and find the first occurrence of key. As soon as we find the key we return the index for it. This is simple method to implement.

First we check for null and array empty condition.

**if** (a == **null** || a.length == 0) {

**return** ***INDEX\_NOT\_FOUND***;

}

Now let us traverse the array from last element.

**int** n = a.length - 1;

**for** (**int** i = n; i >= 0; i--) {

**if** (a[i] == key) {

**return** i;

}

}

**return** ***INDEX\_NOT\_FOUND***;

So we just implemented this method easily.

Below is the code for entire program with comments on it.

**package** arrays;

/\*\*

\* This class is used to find the last index

\* of the key in array.

\* \*/

**public** **class** LastIndexOfInArray {

/\*\*This variable is used to denote the number -1.\*/

**public** **static** **final** **int** ***INDEX\_NOT\_FOUND*** = -1;

/\*\*

\* This method is for int[]

\*/

**public** **static** **int** lastIndexOf(**final** **int**[] a, **int** key) {

/\*\*

\* if array is null or length is 0

\* then return -1.

\* \*/

**if** (a == **null** || a.length == 0) {

**return** ***INDEX\_NOT\_FOUND***;

}

/\*\*Set n as last index in array.\*/

**int** n = a.length - 1;

/\*\*Iterate array in reverse way.\*/

**for** (**int** i = n; i >= 0; i--) {

/\*\*If key found then return index\*/

**if** (a[i] == key) {

**return** i;

}

}

/\*\*Element not found return -1.\*/

**return** ***INDEX\_NOT\_FOUND***;

}

/\*\*

\* This method is for float[]

\*/

**public** **static** **int** lastIndexOf(**final** **float**[] a, **float** key) {

/\*\*

\* if array is null or length is 0

\* then return -1.

\* \*/

**if** (a == **null** || a.length == 0) {

**return** ***INDEX\_NOT\_FOUND***;

}

/\*\*Set n as last index in array.\*/

**int** n = a.length - 1;

/\*\*Iterate array in reverse way.\*/

**for** (**int** i = n; i >= 0; i--) {

/\*\*If key found then return index\*/

**if** (a[i] == key) {

**return** i;

}

}

/\*\*Element not found return -1.\*/

**return** ***INDEX\_NOT\_FOUND***;

}

/\*\*

\* This method is for byte[]

\*/

**public** **static** **int** lastIndexOf(**final** **byte**[] a, **byte** key) {

/\*\*

\* if array is null or length is 0

\* then return -1.

\* \*/

**if** (a == **null** || a.length == 0) {

**return** ***INDEX\_NOT\_FOUND***;

}

/\*\*Set n as last index in array.\*/

**int** n = a.length - 1;

/\*\*Iterate array in reverse way.\*/

**for** (**int** i = n; i >= 0; i--) {

/\*\*If key found then return index\*/

**if** (a[i] == key) {

**return** i;

}

}

/\*\*Element not found return -1.\*/

**return** ***INDEX\_NOT\_FOUND***;

}

/\*\*

\* This method is for short[]

\*/

**public** **static** **int** lastIndexOf(**final** **short**[] a, **short** key) {

/\*\*

\* if array is null or length is 0

\* then return -1.

\* \*/

**if** (a == **null** || a.length == 0) {

**return** ***INDEX\_NOT\_FOUND***;

}

/\*\*Set n as last index in array.\*/

**int** n = a.length - 1;

/\*\*Iterate array in reverse way.\*/

**for** (**int** i = n; i >= 0; i--) {

/\*\*If key found then return index\*/

**if** (a[i] == key) {

**return** i;

}

}

/\*\*Element not found return -1.\*/

**return** ***INDEX\_NOT\_FOUND***;

}

/\*\*

\* This method is for long[]

\*/

**public** **static** **int** lastIndexOf(**final** **long**[] a, **long** key) {

/\*\*

\* if array is null or length is 0

\* then return -1.

\* \*/

**if** (a == **null** || a.length == 0) {

**return** ***INDEX\_NOT\_FOUND***;

}

/\*\*Set n as last index in array.\*/

**int** n = a.length - 1;

/\*\*Iterate array in reverse way.\*/

**for** (**int** i = n; i >= 0; i--) {

/\*\*If key found then return index\*/

**if** (a[i] == key) {

**return** i;

}

}

/\*\*Element not found return -1.\*/

**return** ***INDEX\_NOT\_FOUND***;

}

/\*\*

\* This method is for double[]

\*/

**public** **static** **int** lastIndexOf(**final** **double**[] a, **double** key) {

/\*\*

\* if array is null or length is 0

\* then return -1.

\* \*/

**if** (a == **null** || a.length == 0) {

**return** ***INDEX\_NOT\_FOUND***;

}

/\*\*Set n as last index in array.\*/

**int** n = a.length - 1;

/\*\*Iterate array in reverse way.\*/

**for** (**int** i = n; i >= 0; i--) {

/\*\*If key found then return index\*/

**if** (a[i] == key) {

**return** i;

}

}

/\*\*Element not found return -1.\*/

**return** ***INDEX\_NOT\_FOUND***;

}

/\*\*

\* This method is for boolean[]

\*/

**public** **static** **int** lastIndexOf(**final** **boolean**[] a, **boolean** key) {

/\*\*

\* if array is null or length is 0

\* then return -1.

\* \*/

**if** (a == **null** || a.length == 0) {

**return** ***INDEX\_NOT\_FOUND***;

}

/\*\*Set n as last index in array.\*/

**int** n = a.length - 1;

/\*\*Iterate array in reverse way.\*/

**for** (**int** i = n; i >= 0; i--) {

/\*\*If key found then return index\*/

**if** (a[i] == key) {

**return** i;

}

}

/\*\*Element not found return -1.\*/

**return** ***INDEX\_NOT\_FOUND***;

}

/\*\*

\* This method is for char[]

\*/

**public** **static** **int** lastIndexOf(**final** **char**[] a, **char** key) {

/\*\*

\* if array is null or length is 0

\* then return -1.

\* \*/

**if** (a == **null** || a.length == 0) {

**return** ***INDEX\_NOT\_FOUND***;

}

/\*\*Set n as last index in array.\*/

**int** n = a.length - 1;

/\*\*Iterate array in reverse way.\*/

**for** (**int** i = n; i >= 0; i--) {

/\*\*If key found then return index\*/

**if** (a[i] == key) {

**return** i;

}

}

/\*\*Element not found return -1.\*/

**return** ***INDEX\_NOT\_FOUND***;

}

}