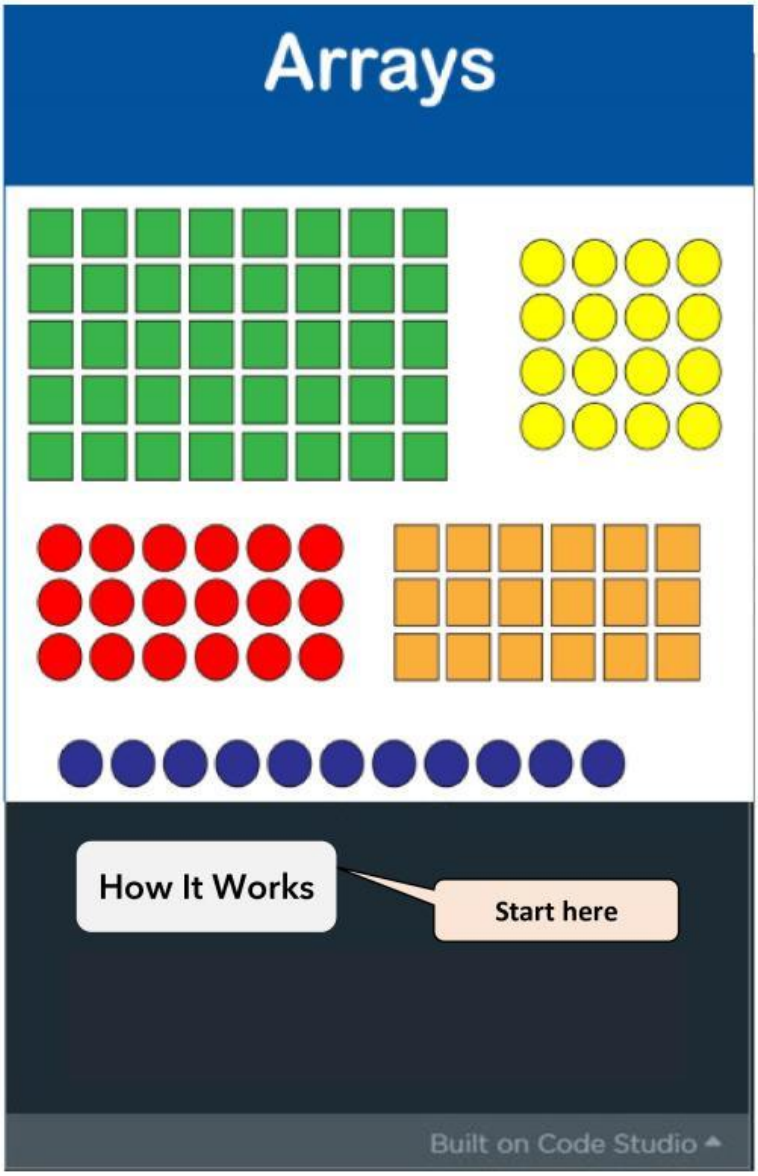
Project 79

DP EDUCATION

Coding School

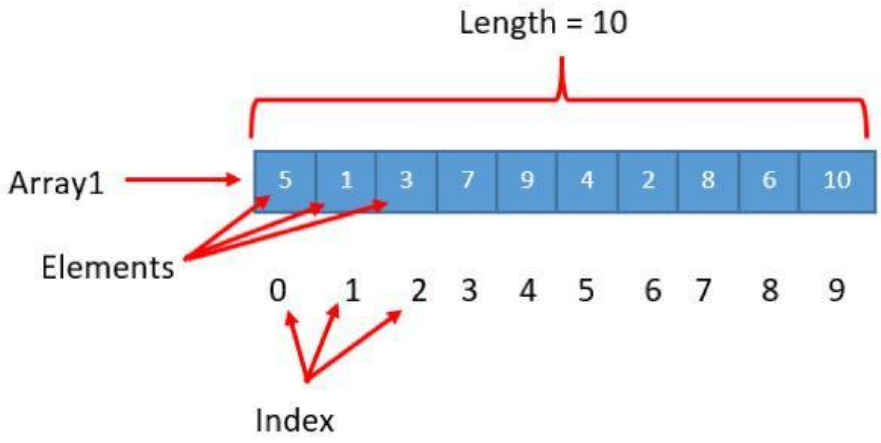


LIVEWORKSHEETS

- Arrays are used if a data set of the same data type needs to be stored in the same place.

- An array can store only one data set of the same data type. And an Array has a fixed length.

- For example, consider the data set 5, 1, 3, 7, 9, 4, 2, 8, 6, 10..



- First give the array a name. We can store that data set in the order we want. An array has a fixed length. The length of this array is 10.

- After the array is defined, it cannot add new data to the array. That is, suppose that 12 should be added as the 11th data in the above data set. But here the array is defined as length 10. Therefore, the 11th data cannot be set in the array.

- The values stored in the array are called elements. The place where the data is stored is called the index.

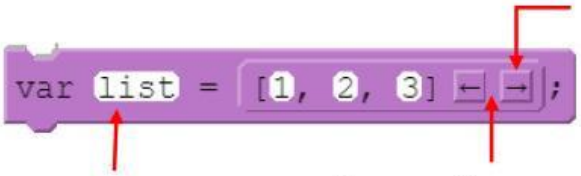
- Index numbering is done from 0 to increasing order. Answers are numbered from 0 to 1, 2, 3. If the length of the array is 10, the maximum value of its index is 9.

- Here we will not create an app and learn more about arrays using the console of the app lab..

LIVEWORKSHEETS

❖ Creating an array in App lab is as follows.

- If the data type of the array is numbers, click variables in the toolbox and use the code block there.
- If the data type of the array is string, click variables in the toolbox and use the code block there



If you need to add data while creating an array, you can click this arrow and add data.

Give the name to be given to the Array here.

If you need to remove data while creating an array, you can click this arrow and remove the

❖ To get the length of the array, click the variables in the toolbox and use the code block there. How to assign the length of the array as "len" to a variable is as follows.

❖ The following code block is used to get each element of the array.

Name of Array The index of the required

Now let's see how to find the largest number in an array.

❖ Let's use the data set 12, 17, 15, 11, 8 for this.

❖ First, let's store the data set in the given array. Let's name the Array as “arr”.

❖ Let's assign the length of the array to the "len" variable.

LIVEWORKSHEETS

```
var len = arr.length;
```

- Now create a variable as "highest" and assign the first element of the array i.e. zero index to it.

```
var highest = arr[0];
```

- Create a for loop and increment the value of variable i from zero until the length of the array is not greater than.

- In the for loop, use an if condition to check if the i-th index is greater than the highest.

```
for( var i = 0; i < len; i++ ){
  if( highest < arr[i] ){
    highest = arr[i];
  }
}
```

Now let's consider the first case in the for loop.
When i = 0 then highest = arr[0] = 12.
Now consider the condition in the for loop.

```
if( highest < arr[i] ){
```

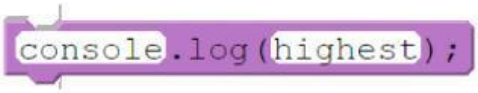
When i = 0 arr[i] is arr[0]. arr[0] = 12. Since arr[0] is smaller than Highest, the condition is false.

When i = 1, highest = 12. arr[1] = 17. Since arr[1] is greater than Highest, the condition is true. What happens in the if condition. ie highest = arr[1]. Then highest = 17.

When i = 2, highest = 17. arr[2] = 15. Since arr[2] is smaller than Highest, the condition is false. So i = 4 in the for loop. When i = 5, i is greater than length, so the for loop ends.

LIVEWORKSHEETS

❖ Now let's display the highest in the console.



Now let's see how to find the largest number and second largest number in an array.

❖ Let's use the data set 6, 9, 7, 1, 4, 3, 2 for this..

❖ First, let's store the data set in the given array. Let's name the Array as “arr2”.

❖ Let's assign the length of the array to the "len" variable.

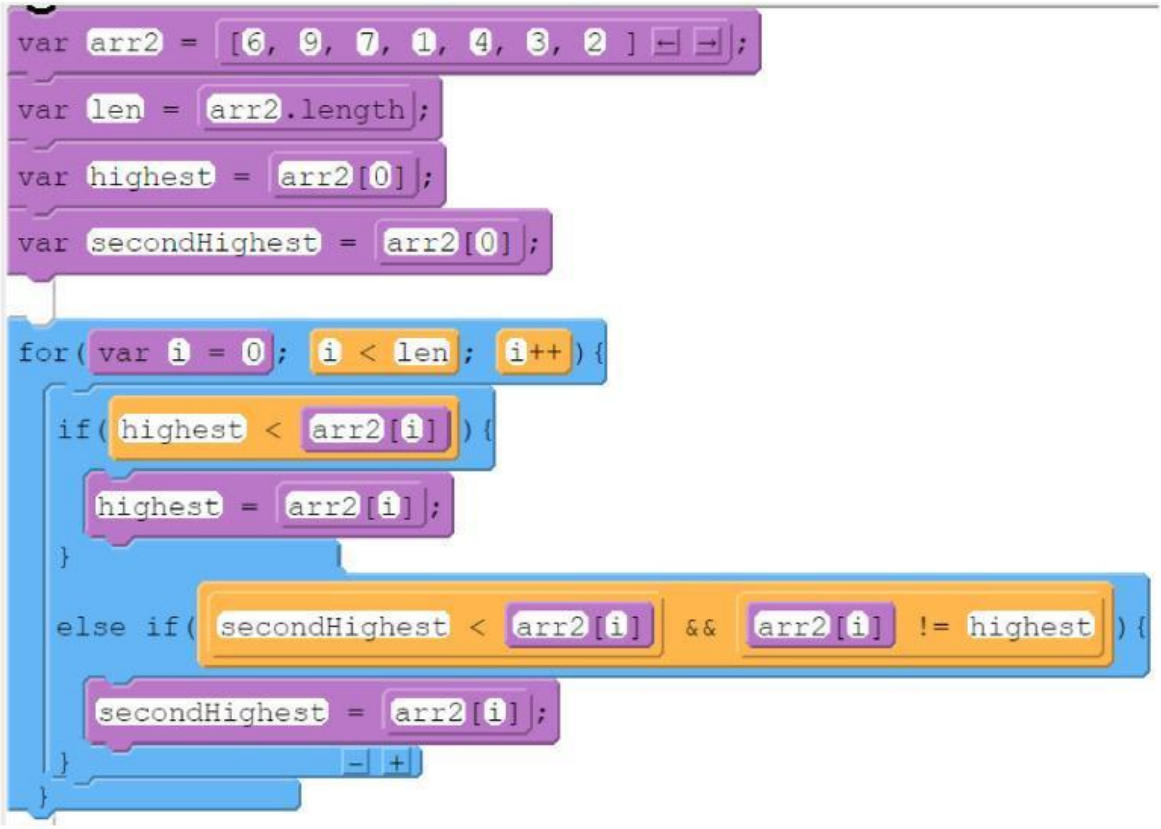
❖ Create a variable as "highest" and assign the first element of the array i.e. zero index to it.

❖ Also create a variable as "secondHighest" and assign the first element of the array i.e. zero index to it.

❖ Create a for loop and increment the value of variable i from zero until the length of the array is not greater than.

❖ By if conditions in the for loop
- Check the cases where the i-th index is greater than the highest.
- In cases where the i-th index is not greater than the highest, check cases greater than the secondHighest.

LIVEWORKSHEETS



6, 9, 7, 1, 4, 3, 2

Now let's consider the first case in the for loop.

When i = 0 then highest = arr2[0] = 6 and secondHighest = arr2[0] = 6.

Now consider the conditions in the for loop.

First condition

Second condition

When i = 0 arr2[i] is arr2[0]. arr2[0] = 6. Since arr2[0] is smaller than Highest, the first condition is false. Since arr2[0] is smaller than secondHighest, the second condition is also false.

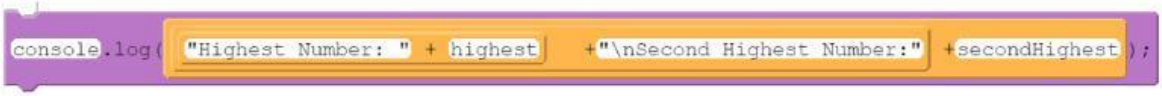
LIVEWORKSHEETS

When i = 1, highest = 6 and secondHighest = 6. arr2[1] = 9. Since arr2[1] is greater than Highest, the first condition is true. The first if condition happens. That is, highest = arr2[1]. Then highest = 9. Since the first condition is true, the second condition does not run

When i = 2, highest = 9 and secondHighest = 6. arr2[2] = 7. Since arr2[2] is smaller than Highest, the first condition is false. The second condition is true because arr2[2] is greater than secondHighest and arr2[2] is not equal to highest.
What happens in the second if condition happens. ie secondHighest = arr2[2]. Then secondHighest = 7.

In this way, i = 6 in the for loop. When i = 7, i is greater than length, so the for loop ends.

❖ Now let's display the highest in the console.



LIVEWORKSHEETS