Array Helper Methods

A byte size lesson in Java

Most Popular Array Methods

- push( )
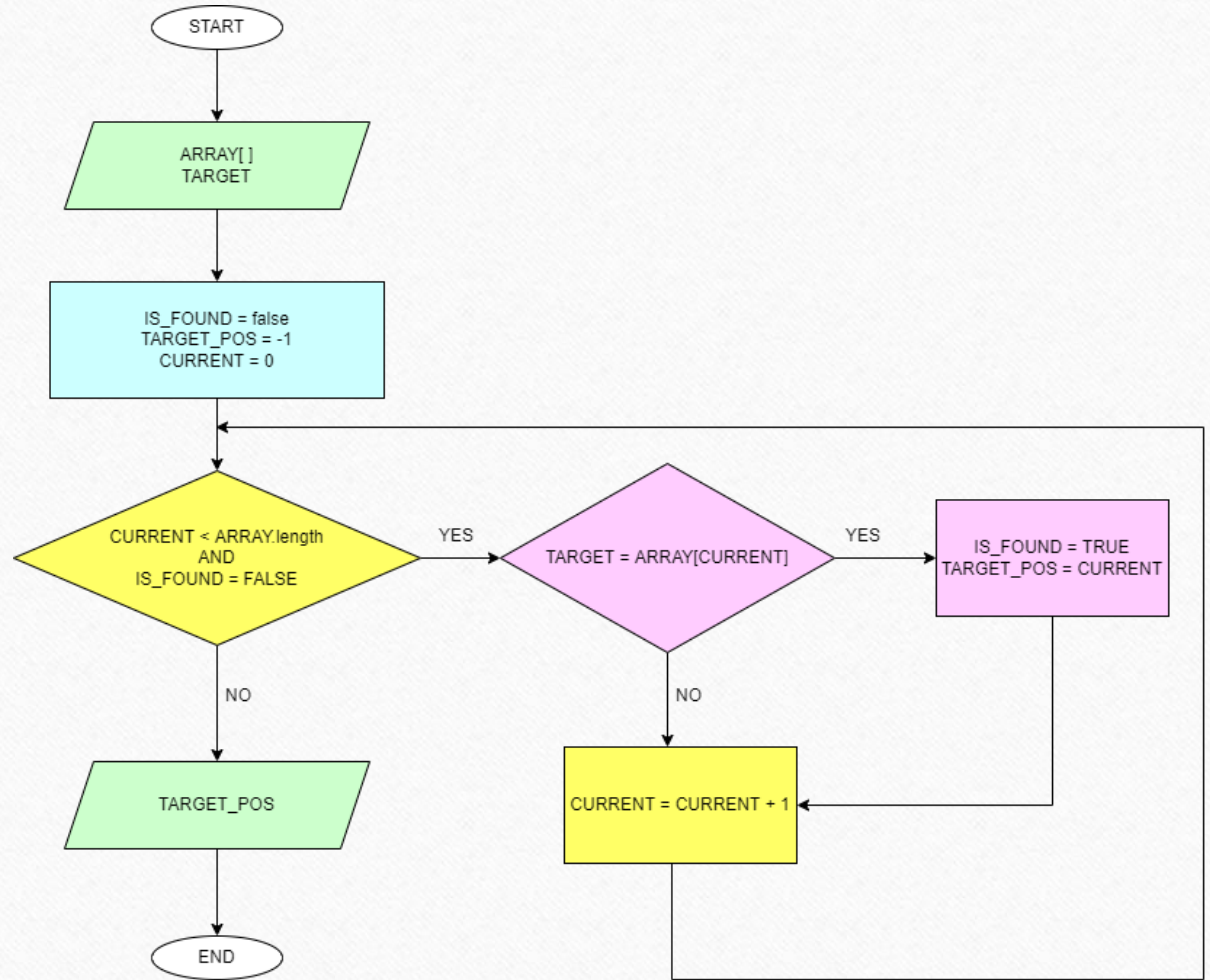- pop( )
- shift( )
- splice( )
- slice( )

- **findIndex( )**
- min( )/max( )
- **sum( )**
- **getFirst( )/getLast( )**
- shuffle( )

Find Index

This method gives the index of a particular element within an array. If the element is not found then -1 is usually returned.

```
int[] numbers = {1, 3, 4, 7, 9};
ArrayHelpers.findIndex(numbers, 1); // gets back 0
ArrayHelpers.findIndex(numbers, 2); // gets back -1
```

Draw a flowchart in this space.

Get First/Last

A method to get the first or last element in an array.

```
int[] numbers = {1, 3, 4, 7, 9};
ArrayHelpers.getFirst(numbers); // gets back 1
ArrayHelpers.getLast(numbers); // gets back 9
```

Sum

This method works for numerical arrays to calculate the total sum of all the elements.

```java
int[] numbers = {1, 3, 4, 7, 9};
int sum = ArrayHelpers.sum(numbers); // gets back 24
System.out.println(sum);
```

Other methods explained

Method Name	Description
push	Place a new element at the next available place in the array.
pop	Remove the last element in the array.
shift	Shift all elements by a number of positions.
slice	Gives back part of an array.
splice	Add or remove multiple array elements in place.
min/max	Find the lowest/largest value in an array.
shuffle	Jumble up all the elements within an array.

Why go through the trouble?

- When you are solving a complex problem (for your Coursework perhaps) and you will use arrays, you will most likely need **at least one** of the most common methods.

- It does not make sense for everyone to write the same code over and over again. We can **share work we did at school**, and you can focus on the original aspects of your project!

- This will save you time and will **reduce the chances of making silly mistakes**.

Example: A Car Park

- We are being entrusted to write a program for a car park in the capital used mostly by workers.

- Customers want a way to plan ahead so they would like a way to know before arriving how much their stay will cost.

Day of the week	Arrival time			
	From 08:00 to 15:59		From 16:00 to Midnight	
	Max stay in hours	Price per hour	Hours	Price
Sunday	8	2.00	Up to Midnight	2.00
Monday	2	10.00	Up to Midnight	2.00
Tuesday	2	10.00	Up to Midnight	2.00
Wednesday	2	10.00	Up to Midnight	2.00
Thursday	2	10.00	Up to Midnight	2.00
Friday	2	10.00	Up to Midnight	2.00
Saturday	4	3.00	Up to Midnight	2.00

Example: Car Park

- Common questions by customers:
  - "When is the cheapest day to park?"
  - "Can I park for more than 4 hours on Friday?"
  - "Are all off-peak rates the same?"

```
String[] daysOfWeek = {"Mon", "Tue", "Wed", "Thu", "Fri", "Sat", "Sun"};
int[] maxHrs = {2, 2, 2, 2, 2, 4, 8};
double[] peakPrice = {10.00, 10.00, 10.00, 10.00, 10.00, 3.00, 2.00};
double[] offPeakPrice = {2.00, 2.00, 2.00, 2.00, 2.00, 2.00};
```

findIndex( )

min( )/max( )

sum( )

getFirst( )/getLast( )

shuffle( )

Circle any methods you think can be useful for the Car Park program.