

Professional Skills
Agile Fundamentals
Jira
Git
Databases Introduction
Java Beginner
<div><div></div>What is Java?</div>
<div><div></div>Installation</div>
<div><div></div>Hello World Example</div>
<div><div></div>Data Types</div>
<div><div></div>Packages</div>
<div><div></div>Naming Conventions Cheat Sheet</div>
<div><div></div>Flow of Control</div>
<div><div></div>Class Members</div>
<div><div></div>Operators</div>
<div><div></div>Conditionals</div>
<div><div></div>Iteration</div>
<div><div></div>Arrays</div>
<div><div></div>ArrayList</div>
<div><div></div>Enhanced For Loops</div>
<div><div></div>String Manipulation</div>
<div><div></div>Class Constructors</div>
<div><div></div>Access Modifiers</div>
<div><div></div>Installing Java &amp; Maven To PATH</div>
<div><div></div>Object-Oriented Programming Principles</div>
<div><div></div>Encapsulation</div>
<div><div></div>Inheritance</div>
<div><div></div>Polymorphism</div>
<div><div></div>Abstraction</div>
<div><div></div>Interfaces</div>
<div><div></div>Type Casting</div>
<div><div></div>Static</div>
<div><div></div>Final</div>
<div><div></div>Garbage Collection</div>
<div><div></div>Input With Scanner</div>
<div><div></div>Pass by Value/Reference</div>
<div><div></div>JUnit</div>

# ArrayList

## Contents

- [Overview](#)
  - [ArrayList in action](#)
  - [add\(\)](#)
  - [get\(\)](#)
  - [set\(\)](#)
  - [remove\(\), size\(\) and clear\(\)](#)
  - [Iterate through an ArrayList](#)
- [Elements are objects](#)
- [Sorting](#)
- [Tutorial](#)
- [Exercises](#)

## Overview

**ArrayList** is a type of collection in Java.

It is an implementation of the **List** interface, backed by an array (`[]`).

This makes your array *resizeable*, as opposed to just using a standard array (which cannot be resized).

Since **ArrayList** is an implementation of **List**, we should *code to the interface* when initialising one:

► ArrayList

### ArrayList in action

Let's see how an **ArrayList** works by making one for a list of sandwiches:

► Sandwiches

#### add()

Let's add a few elements into our **sandwiches ArrayList**:

► ArrayList add()

#### get()

Each element in an **ArrayList** is assigned a unique *index* (starting at **0**) based on its position, much like in a normal array.

We can access an item in the **ArrayList** by using the **get()** method and passing in the *index* of the element:

► ArrayList get()

#### set()

We can modify an **ArrayList** element by passing the *index* of the element to the **set()** method, along with whatever we want it to be set to:

► ArrayList set()

#### remove(), size() and clear()

We can remove one element from the **ArrayList** with the **remove()** method (passing in the index of the element), or all of them with the **clear()** method.

We can also count the number of elements in the **ArrayList** with the **size()** module, since they automatically resize (unlike normal arrays):

► ArrayList remove(), size() and clear()

<div><div></div><div>Test Driven Development</div></div> <div><div></div><div>UML Basics</div></div> <div><div></div><div>JavaDoc</div></div> <div><div></div><div>Peer Programming</div></div> <div><div></div><div>Code Reviews</div></div>
Maven
Testing (Foundation)
Java Intermediate
HTML
CSS
Javascript
Spring Boot
Selenium
Sonarqube
Advanced Testing (Theory)
Cucumber
MongoDB
Express
NodeJS
React
Express-Testing
Networking
Security
Cloud Fundamentals
AWS Foundations
AWS Intermediate
Linux
DevOps
Jenkins Introduction
Jenkins Pipeline
Markdown
IDE Cheatsheet

## Iterate through an ArrayList

We can use a *for-loop* to iterate through an `ArrayList`, passing in the `size()` method to specify how many times the loop should run:

► ArrayList for-loop

This can also be accomplished with an *enhanced for-loop*:

► ArrayList enhanced for-loop

## Elements are objects

Elements in an `ArrayList` are actually objects, so if we wanted a list of `ints`, we would need to use the wrapper class `Integer`:

► ArrayList objects

## Sorting

Since an `ArrayList` is a type of collection, we can use methods from the Java `Collections` class with it, such as the `sort()` method:

► ArrayList sorting

## Tutorial

There is no tutorial for this module.

## Exercises

Try out using `ArrayList` yourself:

- create a new `ArrayList`
- `add()` several elements
- print out the entire `ArrayList`
- iterate through the `ArrayList` and print out each element (with both normal and enhanced for-loops)
- `get()` specific elements
- `set()` different elements
- `remove()` elements
- `sort()` the `ArrayList` (try this with several object types)
- try using the `reverse()`, `swap()` and `clear()` methods.

