

Prompts

Strings

1. Select five (5) methods from the [String Java Documentation](#) and describe the following for each:

- (a) What is the method signature for each method?
- (b) What does each method do?
- (c) Why would this method be useful (how could you use it)? Give an example.

Arrays

2. Select five (5) methods from the [Array Java Documentation](#) and describe the following for each:

- (a) What is the method signature for each method?
- (b) What does each method do?
- (c) Why would this method be useful (how could you use it)? Give an example.

Collections

Resource: [Java Tutorials on Interfaces](#)

- 3. What are the differences between Lists, Sets, and Maps in Java?
- 4. List at least two different implementations for each collection (List, Set, and Map). When would you use one of the implementations over the other?
- 5. Write Java code that does the following:
 - (a) Write a line of code that shows how you would instantiate an ArrayList of String.
 - (b) Write a line of code that shows how you would instantiate a HashSet of String.
 - (c) Write a line of code that shows how you would instantiate a HashMap of String, String.

Week 4 Research

Rachel VanHorn

- 1. 5 String methods: charAt, contains, indexOf, length, repeat
 - a. String.charAt(index);
 - a. String.contains(charSequence);
 - a. String.indexOf(int or String); can also include (int or String ,fromIndex) if you want to search from a specific index on.
 - a. String.length();
 - a. String.repeat(int);
 - b. .charAt() returns the character at the specified index position in the String.
 - b. .contains() returns true if the String contains the specified set of chars.
 - b. .indexOf() returns the index in the String of the first instance of the int or substring. You can also specify where in the index to start searching.

- b. `.length()` returns the length of the String.
- b. `.repeat()` returns the String concatenated to itself the number of times you specify in the parameter.
- c. `.charAt()` can be useful if you are trying to find out the character at a specific position in a String, like if you are trying to check that a user name starts with only an alphabet character and not a number or special character.
- c. `.contains()` could be useful if you are checking to make sure that a password contained the required elements, such as a number and a special character.
- c. `.indexOf()` can be helpful if you are looking for a specific integer, char or substring and you could put it into a loop to keep checking through the whole string starting with the index of where the instance occurred last.
- c. `.length()` is used to determine the length of a string, it is used to help determine how many iterations of a loop we need or how large of an array to create. It can also help check password lengths to see if the password is a certain required length.
- c. `.repeat()` I'm honestly not sure how this is useful other than just to concatenate words to themselves without writing a whole for loop.

Sources:

<https://docs.oracle.com/en/java/javase/17/docs/api/java.base/java/lang/String.html>
<https://www.geeksforgeeks.org/string-class-repeat-method-in-java-with-examples/>
<https://blog.udemy.com/java-charat/>

5.

- a. `List<String> strList = new ArrayList<String>();`
- b. `Set<StringBuilder> sbSet = new HashSet<StringBuilder>();`
- c. `Map<String, String> strMap = new HashMap<String, String>();`

Source:

My notes from the weekly videos