Intro to Java Week 3 Research Assignment

**Points possible:** 30

|  |  |  |
| --- | --- | --- |
| Category | Criteria | % of Grade |
| Accuracy | Is the information accurate? | 25 |
| Organization | Is the essay clean and organized? Ideas are presented in a logical order. | 25 |
| Citations | Students reference and cite at least 5 sources. | 25 |
| Completeness | All requirements of the assignment are complete. | 25 |

**Instructions:** In however many words necessary, write a thorough essay response to each of the below prompts. Be sure to include at least 5 references for this assignment. Do not copy and paste text from the internet or any other source; use the information you find in your research, summarize, in your own words, the concepts. Plagiarism will result in a zero for the assignment as well as disciplinary actions. Push this document to your GitHub repository for this week. Add the URL for this week’s repository to this document where instructed and submit this document to your instructor when complete.

**Select five methods from the String JavaDocs and describe the following for each: 1) what the method signature is, 2) what the method does, and 3) why would this method be useful (how could you use it)?**

**compareTo:** This string compares two strings lexicographically. So if returns<0 then the string comes first, if the returns==0 then the two strings are equivalent but if returns>0 then the parameter that was passed in the comareTo will come first (1). This feature allows collections of strings to be stored and it compares the order in which string comes first in the dictionary (lexicon)(2).

**getBytes(Charset charset):** this one encodes string into a sequence of bytes using the given charset, storing the result into a new byte array (1). This is use when you want the resultant of an array.

**hasCode:** Returns a hash code for this string. The hash code for a String object is computed as

s[0]\*31^(n-1) + s[1]\*31^(n-2) + ... + s[n-1] (1). We use the has code to return a value for the object.

**indexOf (int ch):** This method evaluates the string to determing if the ch is true or -1. Ch- a character(Unicode code point) (1). It shows if the character shoed or did not show but putting a true of a -1.

**Matches:** This one is a Boolean expression and it tells weather of not the string matches to the given regular expression (1). Can use this to match two items.

**Select five methods from the Array JavaDocs and describe the following for each: 1) what the method signature is, 2) what the method does, and 3) why would this method be useful (how could you use it)?**

**asList:** Returns a fixed-size list backed by the specified array. (Changes to the returned list "write through" to the array.) This method acts as bridge between array-based and collection-based APIs, in combination with Collection.toArray(). The returned list is serializable and implements RandomAccess. This method also provides a convenient way to create a fixed-size list initialized to contain several elements (3). So this one holds a predetermind amount of objects and returns it in a list view of the specific array.

**binarySearch(byte[] a,byte key):** this one will search for a specific array of byte for a certain value. This one uses a binary search algorithm to do all of its work. This one returns an index of the search key.

**copyOf:** this one takes an array and copppiess it. But it only does it for a predetermined time. This one returns , a copy of the original array, truncated or padded with false elements to obtain the specified length (3). Can use this when we need to use multiples of the same array.

**deepEquals:** we use this one when we use nesting arrays, it returns true value when the two specific arrays are deeply equal to one another. We use this when we want to nest arrays that are a lot alike.

**toString:** this takes an arrays content and returns a string that is a representation of the content inside of it. We can use this when we need to bring information into another part of the code.

**What is your favorite thing you learned this week?**

Finding out that you can use an array to make your life easier when you code by making a cheat sheet kinda deal.

**References:**

1. [**https://docs.oracle.com/javase/7/docs/api/java/lang/String.html#compareTo(java.lang.String)**](https://docs.oracle.com/javase/7/docs/api/java/lang/String.html#compareTo(java.lang.String))
2. [**https://stackoverflow.com/questions/4064633/string-comparison-in-java**](https://stackoverflow.com/questions/4064633/string-comparison-in-java)
3. [**https://docs.oracle.com/javase/7/docs/api/java/util/Arrays.html#asList(T...)**](https://docs.oracle.com/javase/7/docs/api/java/util/Arrays.html#asList(T...))

**URL to GitHub Repository:**

[**https://github.com/samsquanch27/WeekThreeHomework**](https://github.com/samsquanch27/WeekThreeHomework)