StringBuilder and StringBuffer Methods

Common Methods in String, StringBuilder, and StringBuffer Classes:

1. length():
   The `length()` method is used to determine the number of characters in a sequence. It returns the length of the sequence represented by the object. For example, if you call `length()` on a `StringBuilder` object, it will return the number of characters currently stored in it.
2. charAt(index):
   This method is used to fetch the character present at a specific index in the sequence. The index should be between `0` and `length()-1`. For instance, if the string is "Geeks", `charAt(1)` will return 'e'.
3. indexOf(str):
   The `indexOf()` method searches for the first occurrence of a specified substring within the sequence. If the substring exists, the method returns the index of its first occurrence; otherwise, it returns `-1`.
4. indexOf(str, fromIndex):
   This is a variation of the `indexOf()` method, which starts searching for the substring from a specified index. If the substring is found, it returns the index of the first occurrence; otherwise, it returns `-1`.
5. lastIndexOf(str):
   This method searches for the last occurrence of a specified substring within the sequence. If the substring exists, it returns the index of its last occurrence; otherwise, it returns `-1`.
6. lastIndexOf(str, fromIndex):
   Similar to `lastIndexOf(str)`, this method begins the search from the

specified `fromIndex` and moves backward to find the last occurrence of the substring.

7. compareTo(sb):
   The `compareTo()` method compares two sequences lexicographically. It returns `0` if the sequences are equal, a positive value if the current sequence is greater, and a negative value if it is smaller.
8. substring(beginIndex):
   This method returns a new sequence starting from the specified `beginIndex` to the end of the sequence.
9. substring(beginIndex, endIndex):
   This variation of `substring()` allows you to specify both the starting and ending indices. It returns a new sequence starting at `beginIndex` and ending at `endIndex - 1`.
10. chars():
    The `chars()` method provides a stream of integer values representing the characters in the sequence.

Methods Exclusive to StringBuilder and StringBuffer Classes:

1. append(x):
   The `append()` method adds the specified data to the end of the sequence. This data can be of various types, such as `boolean`, `int`, `char`, `String`, `float`, or `Object`. For example, you can append "World" to "Hello" using `append()`.
2. insert(offset, x):
   This method allows you to insert data at a specified offset position in the sequence. For instance, you can insert "Java" at index `4` in the string "GeeksFor" to make it "GeeksJavaFor".

3. setCharAt(index, c):

   The `setCharAt()` method modifies the character at the specified index in the sequence. For example, you can change the first character of "Geeks" to 'T', resulting in "Teeks".

4. reverse():

   The `reverse()` method reverses the characters in the sequence. For instance, applying `reverse()` on "Geeks" will give "skeeG".

5. deleteCharAt(index):

   This method removes the character at the specified index in the sequence. For example, deleting the character at index `3` in "Geeks" will result in "Gees".

6. delete(start, end):

   This method removes all characters from the specified `start` index to `end-1`. For instance, deleting characters from index `1` to `4` in "Geeks" will result in "Gs".

7. capacity():

   The `capacity()` method returns the current storage capacity of the sequence. It indicates how much data can be stored before the sequence needs to resize itself.

8. replace(start, end, str):

   This method replaces the characters between the specified `start` and `end` indices with a new string. For example, replacing characters from index `1` to `4` in "GeeksForGeeks" with "Hello" will result in "GHelloGeeks".

Here is an example program to demonstrate working of the key functions:

```
class GfG

{

    public static void main (String[] args) {

        // StringBuilder or StringBuffer class

        StringBuilder sb = new StringBuilder("dcba");
    }



        // Reversing the StringBuilder
        sb.reverse();
        System.out.println(sb);

        // Appending to sb
        sb.append("efg");
        System.out.println(sb);

        // Replacing the character at 1
        // with h
        sb.setCharAt(1, 'h');
        System.out.println(sb);

        // Delete the characters at 0, 1
        sb.delete(0, 2);
        System.out.println(sb);

        // Inserts "efg" at 1
        sb.insert(1, "efg");
        System.out.println(sb);
    }
```

```
}
```

Output

```
abcd

abcdefg

ahcdefg

cdefg

cefgdefg
```