Interview Cake

Write a method to reverse a string in-place.

> Since strings in Java are **immutable** , first convert the string into *an array of characters*, do the in-place reversal on that array, and re-join that array into a string before returning it. This isn't technically "in-place" and the array of characters will cost $O(n)$ additional space, but it's a reasonable way to stay within the spirit of the challenge. If you're comfortable coding in a language with mutable strings, that'd be even better!

Breakdown

In general, an in-place algorithm will require swapping elements.

Solution

We swap the first and last characters, then the second and second-to-last characters, and so on until we reach the middle.

Java ▼

```
public static String reverse(String str) {

    char[] strChars = str.toCharArray();

    int startIndex = 0;
    int endIndex = strChars.length - 1;

    while (startIndex < endIndex) {

        // swap characters
        char temp = strChars[startIndex];
        strChars[startIndex] = strChars[endIndex];
        strChars[endIndex] = temp;

        // move towards middle
        startIndex++;
        endIndex--;
    }

    return new String(strChars);
}
```

Complexity

$O(n)$ time and $O(1)$ space.