Strings and Characters API

Java Strings (always with a capital "S") work almost identically to Python strings. The biggest difference is that in Python you can use square brackets to access each character, but in Java, you must use the charAt() method.

Common instance methods of the String class:

- `char charAt(int index)`
  Returns the char value at the specified index in this String.
- `int compareto(String anotherString)`
  Compares two strings lexicographically. Returns a negative number if this String comes "before" `anotherString`, a positive number if this String comes "after" `anotherString`, and zero if they are equal.
- `int compareToIgnoreCase(String anotherString)`
  Compares two strings lexicographically, ignoring case differences. Returns negative, positive, or zero as above.
- `boolean contains(String s)`
  Returns true if this string contains the other String s.
- `boolean endsWith(String s)`
  Returns true if this String ends with the other String s.
- `boolean equals(String s)`
  Returns true if this String is equal to the other String s.
- `boolean equalsIgnoreCase(String s)`
  Returns true if this String is equal to the other String s, ignoring case differences.
- `int indexOf(char ch)    and    int indexOf(String s)`
  Returns the index within this string of the first occurrence of the specified character or substring.
- `int indexOf(char ch, int fromIndex) and int indexOf(String s, int fromIndex)`
  Returns the index within this String of the first occurrence of the specified character or substring, starting at the specified index. Returns -1 if the char or substring is not found.
- `int lastIndexOf(...)`
  Same as `indexOf`, except searches right to left rather than left to right. Returns -1 if not found.
- `int length()`
  Returns the length of this string.
- `String replace(char oldChar, char newChar)`
  Returns a String resulting from replacing all occurrences of oldChar in this string with newChar.
- `String replace(String oldString, String newString)`
  Returns a String resulting from replacing all occurrences of oldString in this string with newString.
- `String[] split(String delimeter)`
  Splits this String around matches of the given delimiter.
- `boolean startsWith(String s)`
  Returns true if this String starts with the other String s.
- `String strip()`
  Returns a string whose value is this String, with all leading and trailing white space removed.
- `String substring(int beginIndex, int endIndex)`
  Returns a string that is a substring of this string. The substring begins at the specified `beginIndex` and extends to the character at index `endIndex` – 1. `endIndex` is optional; if omitted, the substring ends at the end of the String.
- `String toLowerCase()`
  Returns a copy of this String with all of the characters converted to lowercase.
- `String toUpperCase()`
  Returns a copy of this String with all of the characters converted to uppercase.

Static methods in String:

- `static String valueOf(any primitive type, like char, double, int, etc)`
  Returns the string representation of the argument.

See the `Character` class for many useful static methods equivalent to Python ones, such as isUpper, isLower, etc.