|  |  |  |
| --- | --- | --- |
| Method | Arguments | Return Type |
| charAt | int index | Char |
| length | () | Int |
| format | (Format, Object… args)  (Locale l, String format, Object… Args) | String |
| substring | (Int beginIndex)  (int beginIndex,int endIndex) | String |
| split | (String regex)  (String regex,int limit) | String |
| equals | (Object anObj) | boolean |
| equalsIgnoreCase | (Object anObj) | boolean |
| getBytes | () (Charset charset)(String charsetname) | byte[] |
| hashcode | () | int |
| indexOf | (int ch) (int ch, int fromIndex) (String str) | int |
| lastIndexOf | (String str) (String str, int fromIndex) | int |
| isEmpty | () | boolean |
| replaceFirst | (String regex,String replacement) | String |
| replaceAll | (String regex,String replacement) | String |
| split | (String regex) (String regex, int limit) | String[] |

2. I tried out all the above String methods in Java

10. tiny

11. 6 3

12. -2

2

13. No, and the single public class in a Java file must have the same name as the file.

14. Yes, if we use some other static sections to run code.

15. >> uses the sign bit to fill bits from the left. >>> is unsigned and always uses 0 to fill from the left. \

16. Numeric characters, since first character must be alphabetical in identifier.

17. Java only uses pass by Value, so this won’t matter if we use Java. However, it could be in other languages such as C++ we don’t want to make a copy of a large object that takes up a large amount of memory every time we call a function.