Chapter 5: Java Strings

- In Java, there is **no primitive type** to store the **string** data.
- To handle string data in Java, we require **objects of the String class**.
- **String class** is included in the **java.lang package**. Therefore, each class can use the String class without importing any package.
- String objects are **immutable**, that means the value of the objects can never be changed.

String & Operators:

- With String + and += performs concatenation.
- == and != checks for the object references
- equals() and equalsIgnoreCase() will check for the data.

String Methods:

Method	Description
int length()	The length of String is the number of characters present in it.
String concat(String str)	The str is concatenated with the invoking object's string and the resultant string is returned.
String substring(int start_index) String substring(int start_index, int end_index)	The first form extracts a substring from invoking string object from start_index to the end of the string and returns it The second form extracts a substring from invoking string object from start_index to the end_index-1 and returns it
char charAt(int position)	The charAt() will obtain a character from the position and will return that character.
String replace(char original, char replacement) String replace(String original_str, String new_str)	The original char/original_str is replaced by the replacement char/new_str.
String trim()	the method trim() is used to remove leading and trailing blank white spaces.
String toLowerCase()	method toLowerCase() converts all the characters in a string from uppercase to lowercase.
String toUpperCase()	method toUpperCase() converts all the characters in a string from lowercase to uppercase. The non-alphabetical characters such as digits, blank spaces, punctuation marks etc. remains unaffected.