Rio Salado College

Key Terms and Concepts – Lesson 7

CIS163AA – Java Programming – Level 1

Term	Definition
Accessor methods	**Accessor methods** return information about an object.
Anonymous object String variable	An **anonymous object** is an unnamed object. A **String variable** is a named object of the String class.
Append() method	The StringBuilder class **append() method** lets you add characters to the end of a StringBuilder object.
Buffer	A **buffer** is a block of memory.
Capacity	The **capacity** of a StringBuilder object is the actual length of the buffer, as opposed to that of the string contained in the buffer.
Capacity() method	The StringBuilder class **Capacity() method** returns the actual length, or capacity, of the StringBuilder object.
Character class	The **Character class** is one whose instances can hold a single character value. This class also defines methods that can manipulate or inspect single-character data.
Charat () method	The String class **charAt () method** requires an integer argument that indicates the position of the character that the method returns.
Charat() method	The StringBuilder class **charAt() method** accepts an argument that is the offset of the character position from the beginning of a String and returns the character
Compareto() method	The String class **compareTo() method** is used to compare two Strings; the method returns zero only if the two Strings refer to the same value. If there is any difference between the Strings, a negative number is returned if the calling object is “less than” the argument, and a positive number is returned if the calling object is “more than” the argument.
Concatenation	**Concatenation** is the process of joining a variable to a string to create a longer string.
Double class	The **Double class** is a wrapper class that contains a simple double and useful methods to manipulate it.
Endswith() method	The String class **endsWith() method** takes a String argument and returns true or false if a String object does or does not end with the specified argument.
Equals() method	The String class **equals() method** evaluates the contents of two String objects to determine if they are equivalent.

Term	Definition
Equalsignorecase() method	The String class **equalsIgnoreCase() method** is similar to the equals() method. As its name implies, it ignores case when determining if two Strings are equivalent.
Immutable	**Immutable** objects cannot be changed.
Index0f() method	The String class **index0f() method** determines whether a specific character occurs within a String. If it does, the method returns the position of the character; the first position of a String begins with zero. The return value is -1 if the character does not exist in the String.
Startswith() method	The String class **startsWith() method** takes a String argument and returns true or false if a String object does or does not start with the specified argument.
String class	The **String class** is for working with fixed-string data-that is, unchanging data composed of multiple characters.
Stringbuilder stringbuffer classes	The **StringBuilder** and **StringBuffer classes** are for storing and manipulating changeable data composed of multiple characters. It is an alternative to the String class when you know a String will be modified.
Substring()	The **substring()** method allows you to extract part of a String.
Tolowercase()	The String class **toLowerCase()** method converts any String to its lowercase equivalent.
Tostring() method	The **toString() method** converts any object to a String.
Touppercase()	The String class **toUpperCase()** method converts any String to its uppercase equivalent.
Wrapper	A **wrapper** is a class or object that is “wrapped around” a simpler element.

Note: Please see key terms in the textbooks for examples of some of the terms.