\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**API METHODS – FEEL FREE TO USE THIS FOR YOUR EXAM**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

All methods here are instance methods unless otherwise stated. The packages required for the methods/classes included here are javax.swing, java.awt and java.util

**JTextArea(int width,int height)** – creates a text-area component whose width and height in pixels is indicated. Note that the component will “grow” to accommodate the information it will display in any case.

**void setText(String txt)** – this sets up the text associated with a particular component with the value contained in txt.

**Font(String type,int style,int point) –** creates a font and gives it an associated type e.g. “TimesRoman” or “monospaced”, a particular style e.g. Font.BOLD or Font.PLAIN and a particular size (point).

**void setFont(Font f)** – sets the font that a particular component will display to a particular value, which is referenced by the variable f

**void append(String txt)** – this will add on (append) the information stored in the variable txt to the existing text in a text-area component.

**double random()** – this method returns a pseudo-random value between 0 and up to, but not including, 1. It is a static method from the Math class.

**double pow(double x, double y)** – this method returns the value of x raised to the power of b. It is a static method from the Math class.

**double sqrt(double x)** – this method returns the square root of x. It is a static method from the Math class.

**double cbrt(double x)** – this method returns the cubed root of x. It is a static method from the Math class.

**double sin(double x)** – this method returns the trigonometric sine of x, where x is an angle in radians. It is a static method from the Math class.

**double cos(double x)** – this method returns the trigonometric cosine of x, where x is an angle in radians. It is a static method from the Math class.

**double tan(double x)** – this method returns the trigonometric tangent of x, where x is an angle in radians. It is a static method from the Math class.

**double parseDouble(String num)** – this method returns a double whose value has been parsed from the String value num. It is a static method of the Double class.

**float parseFloat(String num)** – this method returns a float whose value has been parsed from the String value num. It is a static method of the Float class.

**int parseInt(String num)** – this method returns an int whose value has been parsed from the String value num. It is a static method of the Integer class.

**String format(String formatString, Object… args)** – this method takes a format string specification (which includes format specifiers) along with zero or more arguments linked to the format string and then returns the resulting formatted string. It is a static method of the String class.

**boolean equals(String str)** – method that can be called on a String object to determine whether or not the string contents exactly matches the value specified by the argument str. If it does it returns the boolean value true, otherwise it returns false.

**boolean equalsIgnoreCase(String str)** – method that can be called on a String object to determine whether or not the string contents exactly matches the value specified by the argument str, ignoring case considerations e.g. with this method the string “BaNaNa” would be considered equal to the string “banana”. If the strings are equal (ignoring case considerations) it returns the boolean value true, otherwise it returns false.

**int length()** – method that can be called on a String object to determine the number of characters it currently contains.

**String substring(int a, int b)** – method that can be called on a String object to return a smaller string whose characters will be those from position a in the original string to position b-1 inclusive

**boolean startsWith(String str)** – method that can be called on a String object to determine whether or not the string begins with the letters specified by the argument str. If it does it returns the boolean value true, otherwise it returns false.

**boolean endsWith(String str)** – method that can be called on a String object to determine whether or not the string ends with the letters specified by the argument str. If it does it returns the boolean value true, otherwise it returns false.

**int indexOf(char ch)** – method that can be called on a String object to determine the position of the first occurrence within the string of the character specified by the argument ch . If the character does not exist within the string, then the method returns the value -1 to indicate this, otherwise it returns the position of the first occurrence of the character, noting that the value 0 would indicate the first position etc

**char charAt(int p)** – method that can be called on a String object to determine the value of a character at a particular position p within the String, beginning at position 0 for the first character.

**boolean isDigit(char ch)** – a static method from the Character class which checks to see whether the char argument ch is a digit character in the range ‘0’ to ‘9’ inclusive. If it is it will return the boolean value true, false otherwise

**boolean isLetter(char ch)** – a static method from the Character class which checks to see whether the char argument ch is a letter character in the range ‘a’ to ‘z’ or ‘A’ to ‘Z’ inclusive. If it is it will return the boolean value true, false otherwise

**boolean isUpperCase(char ch)** – a static method from the Character class which checks to see whether the char argument ch is an uppercase letter character in the range ‘A’ to ‘Z’ inclusive. If it is it will return the boolean value true, false otherwise

**boolean isLowerCase(char ch)** – a static method from the Character class which checks to see whether the char argument ch is a lowercase letter character in the range ‘a’ to ‘z’ inclusive. If it is it will return the boolean value true, false otherwise

**String toLowerCase(String str)** – a method from the String class which converts the String argument str to its lowercase equivalent and returns this value

**String toUpperCase(String str)** – a method from the String class which converts the String argument str to its uppercase equivalent and returns this value

**int getNumericValue(char ch)** – a method from the Character class which can be used to convert the character value ch to its integer equivalent and return this value, but only as long as the character value is in the range ‘0’ to ‘9’ inclusive. So, for example, it can be used to convert the character ‘8’ to the integer 8 quickly. It is a static method from the Character class.

**String showInputDialog(Object message)** – a static method of the class JOptionPane that allows a small dialog box to appear which has a text-box in it allowing input to be supplied. You can take it that the argument will be the *message* you wish to issue to the user.

**void showMessageDialog(Component parentComponent, Object message, String title, int messagetype)** – a static method of the class JOptionPane that allows a small dialog box to appear. You can take it that the first argument will be “**null**”, the second one will be the *message* you wish to issue, the third will give the dialog box a *title* in its title-bar and the last will indicate the *messagetype*, which may be one of **JOptionPane.INFORMATION\_MESSAGE, JOptionPane.PLAIN\_MESSAGE, JOptionPane.WARNING\_MESSAGE** or **JOptionPane.ERROR\_MESSAGE**

**int showConfirmDialog(Component parentComponent, Object message)** – a static method of the class JOptionPane that allows a small dialog box to appear with 3 buttons called “Yes”, “No” and “Cancel”. You can take it that the first argument will be “**null**”, the second one will be the *message* you wish to issue. If the “Yes” button is pressed, the method returns the integer value JOptionPane.YES\_OPTION, it returns JOptionPane.NO\_OPTION for the “No” button, JOptionPane.CANCEL\_OPTION for the “Cancel” button and JOptionPane.CLOSE\_OPTION if the user hits the close button on the dialog title bar.

**int compareTo(String str)** – a method of the class String which compares alphabetically the string the method is called on with the string argument it is passed. The method returns a value < 0 if the string the method is called on is “alphabetically less than” the string argument, it returns a value > 0 if the string the method is called on is “alphabetically greater than” the string argument and it returns exactly 0 if the strings are identical

**String toString(Object[] a)** – a static method of the class Arrays which returns a stringified representation of the contents of the array argument, a, passed to the method. The individual array values are comma-separated and contained within square brackets.

**void sort(Object[] a)** – a static method of the class Arrays which sorts the contents of the array argument, a, passed to the method, into ascending order.

**int binarySearch(Object[] a, Object key)** – a static method of the class Arrays which performs a binary search on the array argument, a, passed to the method, using the search key value key, also passed to the method. If the search key value is found in the array, then the method returns the subscript number associated with the slot in the array containing this value. If the search key is not found, then the method returns a negative integer value.

**Object[] copyOf(Object[] a, int newLength)** – a static method of the class Arrays which makes a copy of the array specified by the argument a. The new array will have a length as specified by the second argument, newLength.