

Lab Task – 4

UML to JAVA Code (**Solve Any 2**)

1. Create the **Student** Class. Then create two **objects** of this class. Take User input for the instance variables.

Student
-name: String -id: int -cgpa: double
<u>+main(String[]) :void</u>

2. Create the **Student** Class. Then create **N Number objects** of this class. Take User input for the instance variables.
3. Create the **Mobile** Class. Then create N number of **objects** of this class. Take User input for the number of objects and the instance variables [**Array of Objects**].

Mobile
-modelName: String -brand: String -code: int -price: double
<u>+main(String[]) :void</u>

String Class in Java (**Solve Any 3**)

1. Concat (**Merge**) Three Strings together.

Input Samples	Output Samples
String – 1: Hello String – 2: My String – 3: Class	Hello My class

2. Input **Three Strings** and find out which one is **Longer**. You have to compares Three strings **lexicographically**

Input Samples	Output Samples
String – 1: Hello String – 2: Class String – 3: School	String 3 is greatest.

3. Input **Three Strings** and find out which two Strings are **Equal**.

Input Samples	Output Samples
String – 1: Hello String – 2: Class String – 3: Hello	String – 1 is Equals to String- 3

4. Store **Some Strings** and Display them in **Alphabetical Order**

Input Samples	Output Samples
Babul Kamal Alam Antara	Alam Antara Babul Kamal

5. Input **Two Strings** and Convert them in both **Lower** case and **Upper** case

Input Samples	Output Samples
Babul Kamal	Lower Case: bakul kamal Upper Case: BAKUL KALA

For Details about String Class, please visit the link:

<https://www.javatpoint.com/java-string>

https://www.w3schools.com/java/java_ref_string.asp

Java String Methods

Reference: https://www.w3schools.com/java/java_ref_string.asp

The String class has a set of built-in methods that you can use on strings.

Method	Description	Return Type
charAt()	Returns the character at the specified index (position)	char
codePointAt()	Returns the Unicode of the character at the specified index	int
codePointBefore()	Returns the Unicode of the character before the specified index	int
codePointCount()	Returns the Unicode in the specified text range of this String	int
compareTo()	Compares two strings lexicographically	int

<u>compareToIgnoreCase()</u>	Compares two strings lexicographically, ignoring case differences	int
<u>concat()</u>	Appends a string to the end of another string	String
<u>contains()</u>	Checks whether a string contains a sequence of characters	boolean
<u>contentEquals()</u>	Checks whether a string contains the exact same sequence of characters of the specified CharSequence or StringBuffer	boolean
<u>copyValueOf()</u>	Returns a String that represents the characters of the character array	String
<u>endsWith()</u>	Checks whether a string ends with the specified character(s)	boolean
<u>equals()</u>	Compares two strings. Returns true if the strings are equal, and false if not	boolean
<u>equalsIgnoreCase()</u>	Compares two strings, ignoring case considerations	boolean
<u>format()</u>	Returns a formatted string using the specified locale, format string, and arguments	String
<u>getBytes()</u>	Encodes this String into a sequence of bytes using the named charset, storing the result into a new byte array	byte[]
<u>getChars()</u>	Copies characters from a string to an array of chars	void
<u>hashCode()</u>	Returns the hash code of a string	int
<u>indexOf()</u>	Returns the position of the first found occurrence of specified characters in a string	int
<u>intern()</u>	Returns the canonical representation for the string object	String
<u>isEmpty()</u>	Checks whether a string is empty or not	boolean
<u>lastIndexOf()</u>	Returns the position of the last found occurrence of specified characters in a string	int
<u>length()</u>	Returns the length of a specified string	int
<u>matches()</u>	Searches a string for a match against a regular expression, and returns the matches	boolean

<code>offsetByCodePoints()</code>	Returns the index within this String that is offset from the given index by <code>codePointOffset</code> code points	int
<code>regionMatches()</code>	Tests if two string regions are equal	boolean
<code>replace()</code>	Searches a string for a specified value, and returns a new string where the specified values are replaced	String
<code>replaceFirst()</code>	Replaces the first occurrence of a substring that matches the given regular expression with the given replacement	String
<code>replaceAll()</code>	Replaces each substring of this string that matches the given regular expression with the given replacement	String
<code>split()</code>	Splits a string into an array of substrings	String[]
<code>startsWith()</code>	Checks whether a string starts with specified characters	boolean
<code>subSequence()</code>	Returns a new character sequence that is a subsequence of this sequence	CharSequence
<code>substring()</code>	Returns a new string which is the substring of a specified string	String
<code>toCharArray()</code>	Converts this string to a new character array	char[]
<code>toLowerCase()</code>	Converts a string to lower case letters	String
<code>toString()</code>	Returns the value of a String object	String
<code>toUpperCase()</code>	Converts a string to upper case letters	String
<code>trim()</code>	Removes whitespace from both ends of a string	String
<code>valueOf()</code>	Returns the string representation of the specified value	String