Tkinter is a Python library that provides a GUI toolkit for building graphical user interfaces (GUIs) using the Tk GUI toolkit. Tkinter is included with most Python installations and is therefore readily available on most platforms.

TK, on the other hand, is the underlying toolkit that Tkinter is built upon. It is a graphical user interface toolkit originally developed for the Tcl programming language, but now also used with other languages such as Python. TK provides a set of widgets and graphics primitives that can be used to create GUIs.

TTK (Themed Tk) is an extension module for Tkinter that provides a set of additional widgets and functionality beyond what is available in Tk. TTK widgets are designed to be more modern and visually appealing than the standard Tk widgets, and they are often styled to match the native look and feel of the underlying operating system.

`curselection()` is a method available in some graphical user interface (GUI) frameworks, such as Tkinter in Python, that allows you to retrieve the currently selected item or items in a listbox or other widget that allows for selection.

The `curselection()` method returns a tuple of integers that represent the index positions of the currently selected items. In most GUI frameworks, the first item in a list or widget is at index 0, the second item is at index 1, and so on.

For example, if you have a listbox containing five items and the user selects the first and third items, the `curselection()` method would return a tuple with the values `(0, 2)`.

You can then use these index values to retrieve the actual items from the listbox using the `get()` method, which takes one or more index values as arguments and returns the corresponding items.

The `get()` method in Tkinter takes one index as its argument and returns the text of the item at that index.

In Tkinter, the `delete()` and `insert()` methods are used to manipulate the contents of a widget, such as a `Listbox`.

The `delete()` method is used to remove one or more items from the widget. The first argument specifies the starting index of the range to be deleted (in this case, 0), and the

second argument specifies the ending index of the range to be deleted (in this case, `'end'`, which is a special string that represents the end of the widget). So, `delete(0, 'end')` removes all the items in the `Listbox` widget.

The `insert()` method is used to add an item to the end of the widget. The first argument specifies the index where the new item should be inserted (in this case, `'end'` again), and the second argument specifies the text of the new item. So, `insert('end', item)` adds the value of the `item` variable to the end of the `Listbox`.

`askokcancel()` is a method in the Tkinter module of Python that displays a pop-up dialog box with an OK and Cancel button, along with a message. This method is often used to confirm an action with the user, such as confirming whether they really want to delete a file.

```
add_btn = Button(window , text="Add Medicine" , bg="light
blue",fg="black",font="helvetica 10 bold",command=add_medicine ) #command is what
responds to the action or the event of the button when the button is clicked.
```

`bind()` is a method in the Tkinter module of Python that is used to associate a function with a particular event that occurs on a widget. The event could be something like a mouse click, a key press or a window resize.

The general syntax of the `bind()` method is as follows:`widget.bind(event, handler)`