

## Echo

*an applet*

This assignment will have roughly the same behavior as our previous Echo assignment; however, this will have a graphic interface in the form of an applet as opposed to a command line interface. This means you will write a class called `Echo` which is a subclass of `Applet`. To get a feel for how this applet should work, you can see my implementation here.

### Behavior

Your applet will take in a `String` and then display that `String` after the user has performed a button click. The user should also be able to click a checkbox which will signify that escape characters are being interpreted. You will be responsible for the same escape characters as in the previous assignment:

- `\n` becomes a new line character.
- `\t` becomes a tab character.

Notice that there is no concern for inserting or removing the trailing newline as in the previous assignment. This only really makes sense in the context of a command line interface.

### Interface

Your applet should minimally have the following:

- (1) An area devoted to input that only takes a single line of text. You will want to use the `TextField` class here.
- (2) A submit button which, upon being clicked, inputs the text in the `TextField` to the applet. You will want to use the `Button` class here.
- (3) A checkbox which signifies whether or not escape characters will be interpreted. If the checkbox is checked, then you will replace each `\n` with a newline and each `\t` with a tab. You will want to use the `CheckBox` class here.
- (4) An area devoted to displaying the output after it has been processed. You should use the `TextArea` class here.

Make sure that the user cannot input text into this area, except by clicking the submit button or pressing enter while in the `TextField`. Note that pressing enter in a `TextField` generates an `ActionEvent` by default, so you will not have to implement this yourself.

Additionally, use something other than the default layout manager – the default is `FlowLayout`, by the way.

### Implementation

You will minimally need to implement the `ActionListener` and the `ItemListener` interfaces. Make sure that your `Echo` class does not implement the interfaces itself, but that you have some other class doing all the listener logic. You may either use an inner class or an anonymous class for your listener logic. Below is a rough skeleton of the `Echo` class. Note that yours may vary, there are many ways to write this applet.

```
public class Echo extends Applet {
    TextArea output;
    TextField input;
    Checkbox escape;
    Button submit;

    public void init();

    private class Listener implements ActionListener, ItemListener {
        public void actionPerformed(ActionEvent e);
        public void itemStateChanged(ItemEvent ie);
    }
}
```

Notice that there is no `paint()` method. This is because all of the elements of the applet are components and so, they paint themselves. Of course, you may implement things differently than I have and so may require a `paint()` method.

Be sure that you perform appropriate actions in each of the methods (incl. `init()` and any other methods you may add).

---

**Remember** that the name of the file must be the same as the name of the class with `.java` appended to it. In other words, your file should be named `Echo.java`.

Submit your assignment by creating an appropriate html file in your `public.html` directory and copying the `Echo.class` file into the same directory. **Be sure that you only copy the class file, otherwise your source code will be available to everyone else!** Put the source file – i.e. `Echo.java` – in a directory called `applets` in your submissions directory. There is no need to place the `Echo.class` file in the directory.