The Keyboard Event Properties

When an event handler that reacts to a keyboard event is called, a Keyboard event argument is passed in. Let's revisit our `dealWithKeyboard` event handler that you saw earlier. In that event handler, the keyboard event is represented by the `e` argument that is passed in:

```
function dealWithKeyboard(e) {
    // gets called when any of the keyboard events are overheard
}
```

This argument contains a handful of properties:

- `keyCode`

  Every key you press on your keyboard has a number associated with it. This read-only property returns that number.

- `charCode`

  This property only exists on event arguments returned by the `keypress` event, and it contains the ASCII code for whatever character key you pressed.

- `ctrlKey`, `altKey`, `shiftKey`

  These three properties return a **true** if the Ctrl key, Alt key, or Shift key are pressed.

- `metaKey`

  The `metaKey` property is similar to the `ctrlKey`, `altKey`, and `shiftKey` properties in that it returns a **true** if the Meta key is pressed. The Meta key is the Windows key on Windows keyboards and the Command key on Apple keyboards.

The Keyboard event contains a few other properties, but the ones you see above are the most interesting ones. With these properties, you can check for which key was pressed and react accordingly. In the next couple of sections, you'll see some examples of this.