# 03 User Interface

## Keyboard/Mouse Input

**Mouse variables** – Processing maintains the several useful global variables for tracking mouse position and the state of the mouse’s buttons: mouseX, mouseY, mousePressed, pmouseX, pmouseY, etc. This allows several useful constructs, e.g.

// Draw a line if the left mouse button is pressed

if (mousePressed && (mouseButton == LEFT))

line(pmouseX, pmouseY, mouseX, mouseY);

**Mouse events** – Processing defines special functions can be called if the mouse is clicked, moved or dragged, e.g.:

void mousePressed() {

println("mouse pressed");

}

**Keyboard variables/events** – similarly, global variables and functions such as keyDown and keyPressed() can allow keyboard input to be managed:

void keyPressed() {

println("a key was pressed, code "+keyCode);

}

## Designing UI Controls

**Control design** – controls such as buttons, lists and textboxes are perfect examples of objects. They have fields – e.g. state of the button (pressed/not pressed), contents of the textbox – and methods – e.g. click, type a character.

|  |  |  |
| --- | --- | --- |
|  | ImageButton | Textbox |
| Fields | text  position  size  textColor | text  position  size  focused  textColor  selected |
| Methods | draw  clickStart  clickEnd | draw  keyTyped |