Reacting to the Mouse and Keyboard

The `mousePressed`, `mouseX`, and `mouseY` Variables

```java
void draw(  )
{
    stroke( 0, 0, 0 );
    fill( 255, 50, 50  );
    if( mousePressed )
    {
        ellipse( mouseX, mouseY, 50, 50 );
    }
}
```

- `mousePressed` is a built-in variable that is always telling you if a mouse button is currently pressed.
- `mouseX` and `mouseY` are built-in variables that are always telling you where the mouse cursor is.
The `mousePressed`, `mouseX`, and `mouseY` Variables

```java
void draw( )
{
    if( keyPressed )
    {
        switch( key )
        {
            case 'r':
                fill( 255, 50, 50 );
                break;
            case 'g':
                fill( 50, 255, 50 );
                break;
            case 'b':
                fill( 50, 50, 255 );
                break;
        }
    }
    if( mousePressed )
    {
        ellipse( mouseX, mouseY, 50, 50 );
    }
}
```

The `keyPressed` and `key` Variables

- `keyPressed` is a built-in variable that tells you if a keyboard key has been pressed.
- `key` is a built-in variable that tells you what key has been hit.

The `switch/case` statements are Processing’s way of checking many values without having a whole slew of if-statements.

The `stroke()` and `fill()` calls have been moved to `setup()`.
What if you want to read the Special Keys?

```java
... 
if (keyPressed )
{
  if ( key == CODED )
  {
    switch( keyCode )
    {
      case UP: // up-arrow
      ...
      break;
    }
  }
}
```

Values for `keyCode` can be:
- UP
- DOWN
- LEFT
- RIGHT
- ESC
- DELETE
- BACKSPACE
- TAB
- ENTER
- RETURN
void mousePressed( )
{
if( Debug )
    println( "mouse button = " + mouseButton );
}

void mouseMoved( )
{
if( Debug )
    println( "mouse has been moved: " + mouseX + ", " + mouseY );
}

void mouseDragged( )
{
if( Debug )
    println( "mouse has been dragged: " + mouseX + ", " + mouseY );
}