Cloning

TODO

TODO

A game with just one platform isn't very impressive, so we need to add some more. But we don't want to copy/paste the code every time we make a platform–that would be no good at all!

What we want to do is to move the code for making a platform into a **function**. A function (aka. procedure, aka. "custom block" from Scratch) lets you give a **name** to a piece of code, so you can refer to it by name.

- Move the platform code into a function, by adding `function makePlatform( ) {` at the beginning of that section, and `}` at the end.

  ```
  function makePlatform() {
    //...[all your platform code]...
  }
  ```

  *Put the `function` bits **around** your existing code.*

  You'll want to re-indent your code at this point–try selecting the code inside the function, and pressing the Tab key.

Save. You'll notice the platform disappears. This is because we haven't **called** our function; we've defined it, but not used it yet! Let's do that now.

- Make a platform by calling your new function.

  ```
  makePlatform()
  ```

  Add this at the end of your program, before the final `});`.

Now we can easily make more platforms!

- Add some more platforms.

```
makePlatform()
makePlatform()
makePlatform()
// ...
```

The platforms are all being made at the same height - we really want to make each platform higher than the last. Let's make a variable at the beginning of our program, to keep track of the height of the last one…

- Create a `lastY` variable.

```
var lastY = 100
```

Add this before the `makePlatform` function.

Make sure not to put this *inside* `makePlatform`, or you'll get a **different** version of the `lastY` variable each time. We need to remember it between calls to our function, so it has to go outside.

Now let's use it.

- Use `lastY` inside `makePlatform`.

```
    platform.posY = lastY
```

- **Challenge:** increase `lastY` every time you make a platform.

  *Make sure you do this **inside** `makePlatform`!*

Destroying clones

Finally, if we don't destroy the bullets, eventually the game will get really slow! Let's fix that, by destroying them once they're completely off the screen:

- Destroy the bullets once they’re completely off-screen.

```
        if (!bullet.isOnScreen()) {
            bullet.destroy()
        }
```

This should go inside the bullet’s `forever`, just after the code to move it.

The `destroy()` function attached to a Sprite removes it from the screen permanently. This also stops any `forever` loops attached to it.