Text Boundaries

Text can be nicely constrained to a boundary.

This fixed zone can be set using `orxText_SetSize()` or the Object's `Size` property. The zone could be defined either with a fixed width and/or a fixed height. The combination of settings will change the behaviour of how the text flows within the boundary or zone.

Well work though some examples, but first, make a project.

Setting up a new project

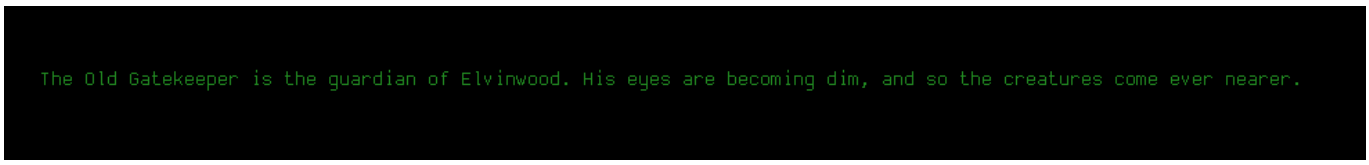
To help you work through this tutorial, first create a new blank project using the init script.

Setting up some demo text

Let's change the default logo object into a text object. We will use the same piece of text for each demo to clearly show the effect of each setting.

```
[Object]
Graphic   = @
Text      = @
String    = The Old Gatekeeper is the guardian of Elvinwood. His eyes are
becoming dim, and so the creatures come ever nearer.
Smoothing = false
Pivot     = center
Color     = ForestGreen
```

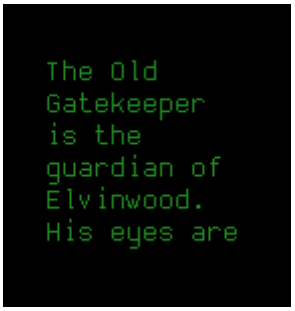
In it's plain form above, this will format like this:



Let's add a fixed size to the object with:

```
Size    = (100, 100)
```

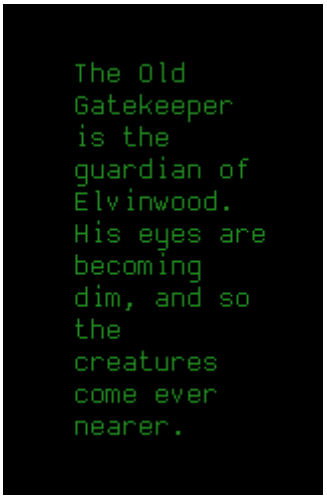
If you re-run the game, the 100×100 rectangle is not large enough to contain the whole text, and any words that don't fit become truncated.



It is possible to not supply a fixed height. This is done by using a `0` value:

```
Size    = (100, 0)
```

What this is saying is that we want a fixed width of 100, but the height should be automatically sized to whatever is needed to fit all of the text.



Setting the size using Commands

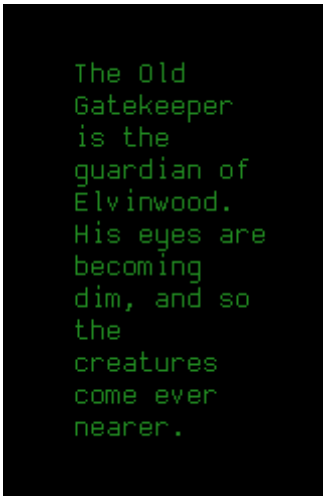
Before running, set the size back to 100,100:

```
Size    = (100, 100)
```

Run the game.

Press the ~ key (tilde) to open the Orx Console while the application is running.

```
>Object.FindNext Object
Object.SetSize < (100,0)
```



Setting the size using Code

We can use the `orxText_SetSize` in code in order to automatically resize the text and re-render.

In order to be able to affect the text object, we'll need to remove the creation of the Scene:

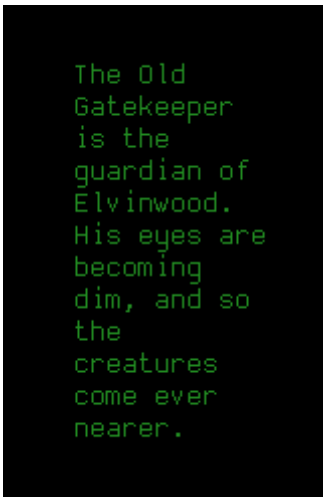
```
orxObject_CreateFromConfig("Scene");
```

And replace it with:

```
orxOBJECT *textObject = orxObject_CreateFromConfig("Object");
```

Then add the following underneath to change the size:

```
orxVECTOR newSize = { 100, 0 };
orxObject_SetSize(textObject, &newSize);
```



You can also see a nicer example of text being resized both horizontally and vertically here:

Text Boundaries

From:
https://orx-project.org/wiki/ - **Orx Learning**

Permanent link:
https://orx-project.org/wiki/en/tutorials/text-fonts/text_boundaries

Last update: **2025/09/30 17:26 (13 months ago)**