Coding workshop: Add GUI widgets to your app

Introduction

In this worksheet, you will learn how to add some GUI widgets to your JUCE application.

Objectives

At the end of this worksheet, you will be able to:

- Use the JUCE framework to create and compile a starter application
- Explain what a GUI library is and give examples of common GUI widgets

Add a button

We are now going to add a TextButton widget to the JUCE application.

Add the class data member to the MainComponent header

Open the MainComponent.h file in your IDE. Add a data member to the private section of the MainComponent class in MainComponent.h. The type is TextButton, and the name can be 'playButton'. This code should suffice:

```
juce::TextButton playButton;
```

Note - you do not need to prefix the type with 'juce::', but it makes it explicit that this class is in the juce namespace (and therefore part of the juce library).

Set up the button in the MainComponent constructor

Open up the MainComponent.cpp file in your IDE. We are going to add some code to set up the button to the constructor (MainComponent::MainComponent()). Call the addAndMakeVisible function, passing it the playButton.

addAndMakeVisible is part of the Component class and has the following function prototype:

```
void addAndMakeVisible (Component& child, int zOrder = -1);
```

There are a few things to note about this prototype:

- It uses pass by reference.
- It has a default value for the second parameter.

Add this code to the end of your constructor:

```
addAndMakeVisible(playButton);
```

Set the dimensions of the button in the resize function

Next, we have to set the dimensions of the button. We do this in our component's MainComponent::resized function. This function is called when the app window first opens and whenever it is resized (e.g. by the user). Add this to resized:

```
playButton.setBounds(0, 0, getWidth(), getHeight()/5);
```

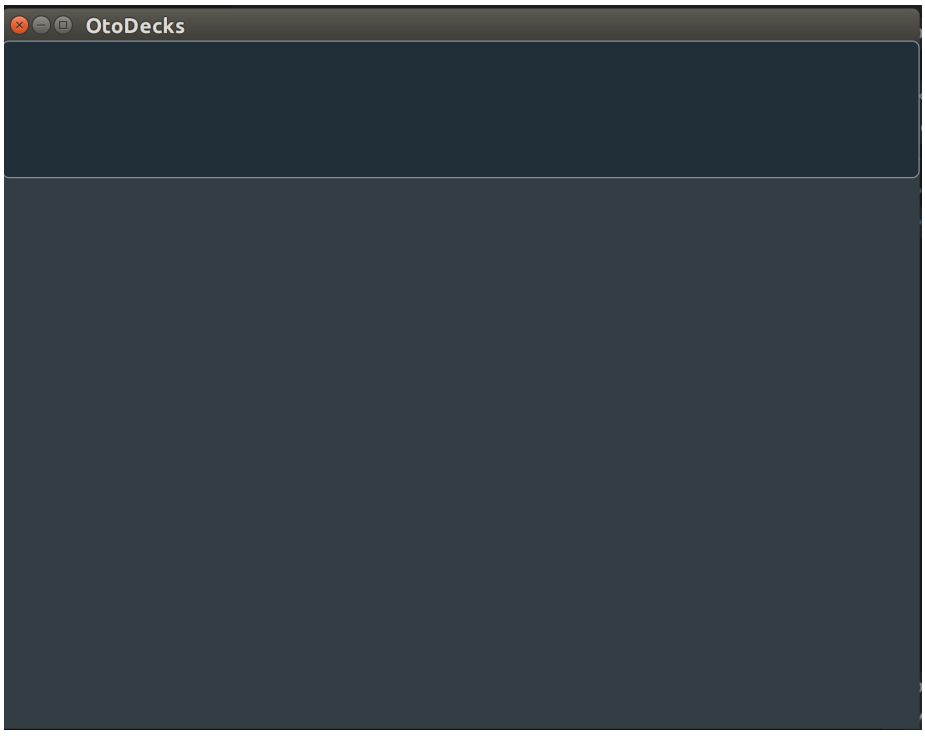
The prototype for setBounds is:

```
void setBounds (int x, int y, int width, int height);
```

Note that we use getWidth and getHeight to set the width and height - these functions tell us the size of the parent component. We set the height to 1/5 of the total height of the main component.

Test it out

Now build and run the project. You should see something like this:



When you mouseover the button and click it, it should animate.

Next steps

The next step is to experiment with the widget set available in JUCE. Have a look at the documentation the GUI basics module:

https://docs.juce.com/master/group___juce___gui___basics.html

Try adding some different components to your app.

- Think about the DJ applications you have looked at - what kind of components did they have? Are any of these available in JUCE?
- Think about how you might control the layout - what happens if you use numerical values for the position and size? Can you make the layout resize when the user resizes the window?
- Can you create a GUI that looks a bit like a DJ app you have seen?