**Assessment Task 1: Challenge Question.**

Due Week 6, Friday 7th April (before class).

Submit your assignment as a PDF document and Processing sketch in a zip file via email to [joshua.batty@rmit.edu.au](mailto:joshua.batty@rmit.edu.au)

This task is designed to test your knowledge around designing an interactive real-time drawing tool. You will create custom “paintbrushes” where the mouse is your brush and your code defines how the brush makes marks on the screen.

**Design a Processing based solution that does the following:**

- facilitates a real-time drawing tool via mouse interaction in Processing

- incorporates various 2D shapes, colours, mouse and keyboard events

- makes use of global and local variables for controlling your sketch

For additional points respond to the following bonus design problems:

+:Write at least one custom function.

++:Host the projects code on Github and openProcessing.

+++:Create a new class for saving images to the data folder.

++++:Create custom brushes from .png image files.

+++++:Create a welcome screen that greets you on launch of the program.

**Deliverables**

The submission should include:a written description of the design/original Processing program code with descriptive comments/images and illustrations of artistic influences/references to sources.

**Planning**

Abandon the scree shots – just use what we have been shown

Use an array to add shapes across the screen – this can be cleared, and even use an undo function

Rotate and translate can be turned on or off

Welcome screen – Boolean flip-flop

Gets checked before the background/canvas redraw function

Take the paint program and add a welcome screen,

and a class system for saving and loading images

Include aspects of the rectangle drawing tool that saves the images and shows a preview of what’s being drawn

Take the interesting brush you made using loops and circles, create a class

Add a way to use png files as the object’s brush

**Presentation**

I wanted a preview of what would be drawn – but because the background was being redrawn on each loop it would erase what I had done and I couldn’t add anything new

I found a way to save the background – which could be displayed on a separate screen – and then use that instead of the background() function.