Lab Session 3 – Introduction to Android Studio

.Objectives

The objectives for this week are:

- Understand the structure of Android Studio projects.
- Create a basic Android Studio project.
- Start the construction of an Arkanoid clone game within Android.

.Tasks

Task 1

- Create a new project in your H: drive following the same process as was described in Lab 1. This should give you a basic outline of an Android application.
- Follow the videos below the lab sheet. These will assist you in creating a simple project in Android.
- Additionally, they will give you an understanding of the structure of an Android Studio project.
- To test the project and understand more about the virtual devices, create a number of different virtual devices using the AVD manager.
- Test the project on:
    - A different phone type / screen size to one that you created.
    - A tablet.
    - A wear device.
- If the emulator is not in a good position, use Alt+Space+M to move it to a better location / minimise.

Task 2

- The game Arkanoid starts with a passage:

"THE TIME AND ERA OF THIS STORY IS UNKNOWN. AFTER THE MOTHERSHIP ARKANOID WAS DESTROYED, A SPACECRAFT "VAUS" SCRAMBLED AWAY FROM IT. BUT ONLY TO BE TRAPPED IN SPACE WARPED BY SOMEONE……."

- Create a new string item in the Strings.xml file within the Res folder that depicts this passage. Name the string "overview".
- Replace the string that is currently being used for the "Hello World" string with the newly created string. If you are unsure of this process, discuss it with your lab tutor.
- The string can use HTML tags to alter its format when compiled. Use the below tags to enhance the string to emphasise key parts:

Supported HTML elements include:

- <b> for **bold** text.
- <i> for *italic* text.
- <u> for <u>underline</u> text.

Task 3

- There is need to be able to debug code at various points throughout your games development.
- Android Studio has many of the features that you will have used in other IDE's such as Visual Studio and Eclipse.
- To aid the debugging process, set the line numbering to on. This is done by:
  - Pressing "SHIFT – SHIFT" and searching for "Show line numbers".
  - Click the slider in the appropriate line that relates to line numbers to "ON".

Task 4

- Placing text to the console in Android Studio requires that we use a different approach to other programming formats. It is best practice to use the Log method to display feedback and errors in Log Cat.
- Log is a logging class that you can utilize in your code to print out messages to the LogCat. Common logging methods include:
  - v(String, String) (verbose)
  - d(String, String) (debug)
  - i(String, String) (information)
  - w(String, String) (warning)

- e(String, String) (error)

- Add a log message to your MainActivity so that it replicates the below:

```
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    //Output a string to the Log Cat
    Log.v("Message", "Output from here");
    setContentView(R.layout.activity_main);
}
```

- Log will need to added as an import below package. Add in the below code to stop the error.

```
import android.util.Log;
```

- This can also be achieved using Alt+Enter when you are promoted by Android Studio.
- Run the application and check Log Cat to see the output. Your lab tutor can assist you by showing you how to view Log Cat.