EE447 Lab1 Report

Prof. Luoyi Fu, EE447 Mobile Internet

Spring semester, 2020

Fan Zhou(周凡)
517030910305

1 Android Programming Environment

In this section I will show how I configure Android programming environment. I will mainly cover two parts:Java Environment and Android Studio configuration.

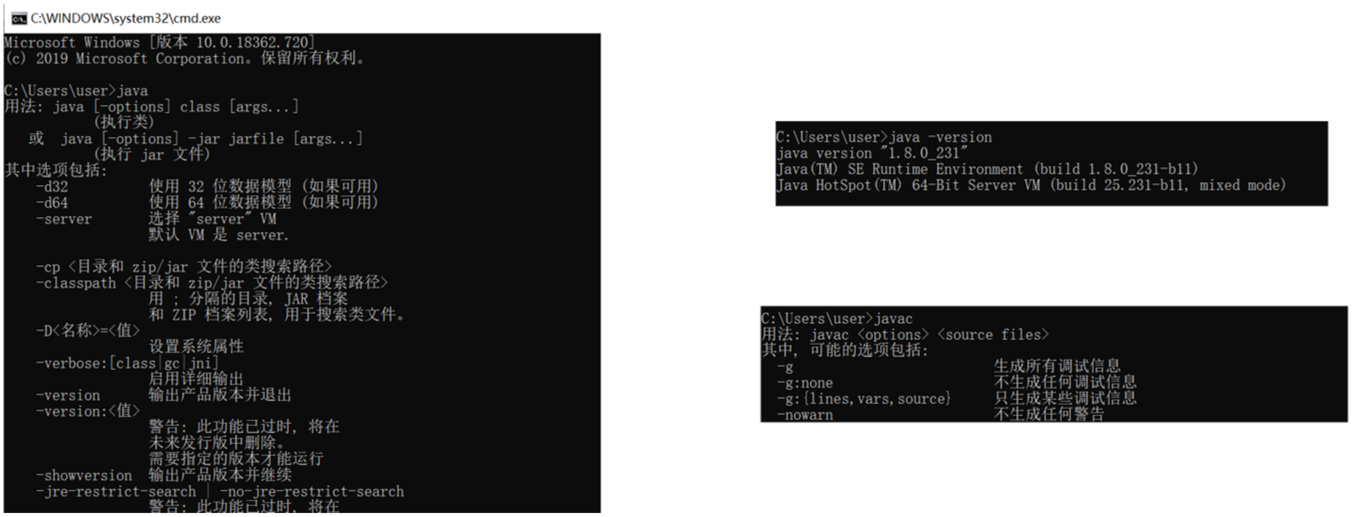
1.1 Java Installation

Several Steps:

1) First go to Oracle website to download jdk package, click next until complete.

2) configure system variable(in windows system).

3) Once Finished, test in cmd shell.



```
Microsoft Windows [版本 10.0.18362.720]
(c) 2019 Microsoft Corporation。保留所有权利。

C:\Users\user>java
用法: java [-options] class [args...]
           (执行类)
   或  java [-options] -jar jarfile [args...]
           (执行 jar 文件)
其中选项包括:
    -d32          使用 32 位数据模型 (如果可用)
    -d64          使用 64 位数据模型 (如果可用)
    -server       选择 "server" VM
                  默认 VM 是 server.

    -cp <目录和 zip/jar 文件的类搜索路径>
    -classpath <目录和 zip/jar 文件的类搜索路径>
                  用 ; 分隔的目录, JAR 档案
                  和 ZIP 档案列表, 用于搜索类文件。
    -D<名称>=<值>
                  设置系统属性
    -verbose:[class|gc|jni]
                  启用详细输出
    -version      输出产品版本并退出
    -version:<值>
                  警告: 此功能已过时, 将在
                  未来发行版中删除。
                  需要指定的版本才能运行
    -showversion  输出产品版本并继续
    -jre-restrict-search | -no-jre-restrict-search
                  警告: 此功能已过时, 将在
```

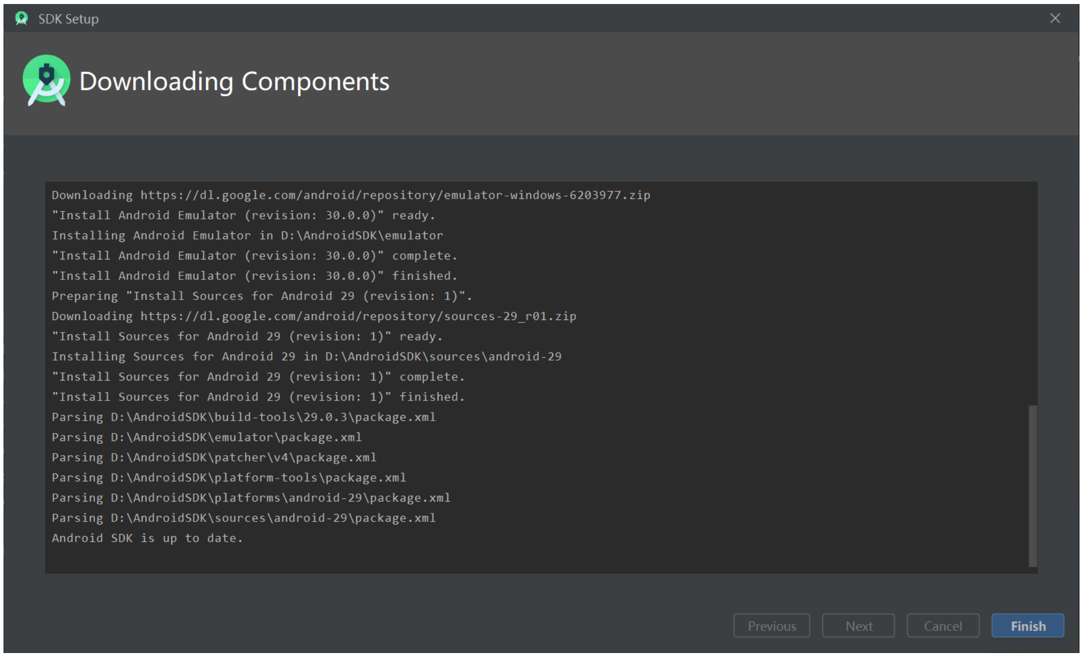
```
C:\Users\user>java -version
java version "1.8.0_231"
Java(TM) SE Runtime Environment (build 1.8.0_231-b11)
Java HotSpot(TM) 64-Bit Server VM (build 25.231-b11, mixed mode)
```

```
C:\Users\user>javac
用法: javac <options> <source files>
其中, 可能的选项包括:
  -g                         生成所有调试信息
  -g:none                    不生成任何调试信息
  -g:{lines,vars,source}     只生成某些调试信息
  -nowarn                    不生成任何警告
```

图 1: cmd shell, testing if jdk has been sucessfully installed.Type ”java” ”java -version” ”javac” to check.

1.2 Android Studio and SDK

Android Studio can be downloaded in the official website. After installation, using SDK manager to set up SDK.

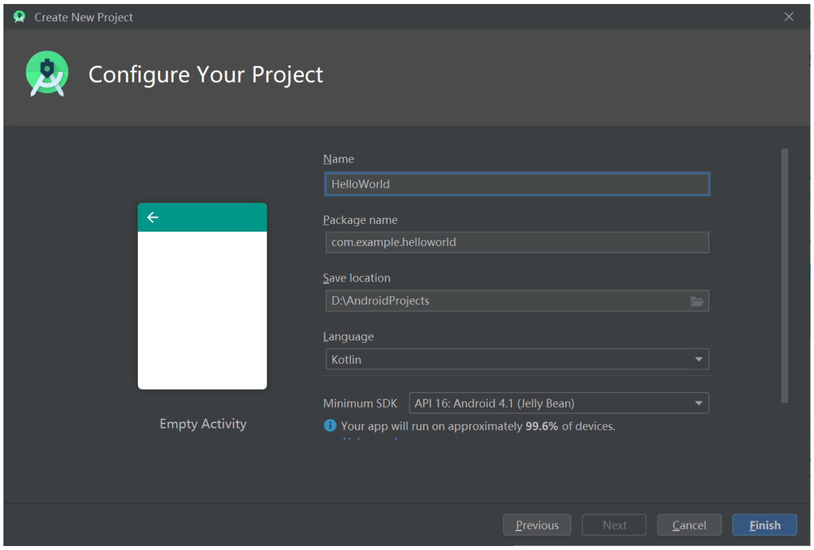


```
Downloading https://dl.google.com/android/repository/emulator-windows-6203977.zip
"Install Android Emulator (revision: 30.0.0)" ready.
Installing Android Emulator in D:\AndroidSDK\emulator
"Install Android Emulator (revision: 30.0.0)" complete.
"Install Android Emulator (revision: 30.0.0)" finished.
Preparing "Install Sources for Android 29 (revision: 1)".
Downloading https://dl.google.com/android/repository/sources-29_r01.zip
"Install Sources for Android 29 (revision: 1)" ready.
Installing Sources for Android 29 in D:\AndroidSDK\sources\android-29
"Install Sources for Android 29 (revision: 1)" complete.
"Install Sources for Android 29 (revision: 1)" finished.
Parsing D:\AndroidSDK\build-tools\29.0.3\package.xml
Parsing D:\AndroidSDK\emulator\package.xml
Parsing D:\AndroidSDK\patcher\v4\package.xml
Parsing D:\AndroidSDK\platform-tools\package.xml
Parsing D:\AndroidSDK\platforms\android-29\package.xml
Parsing D:\AndroidSDK\sources\android-29\package.xml
Android SDK is up to date.
```

图 2: SDK setup window

2 First Project–Hello World

Create a new project with an empty activity and you will find this in the MainActivity.java: If you run this APP in the virtual machine or in a real Android phone, you can see the word ”hello world” in the screen.



Name: HelloWorld

Package name: com.example.helloworld

Save location: D:\AndroidProjects

Language: Kotlin

Minimum SDK: API 16: Android 4.1 (Jelly Bean)

Your app will run on approximately **99.6%** of devices.

图 3: Create a new Project, ”Hello world” as default

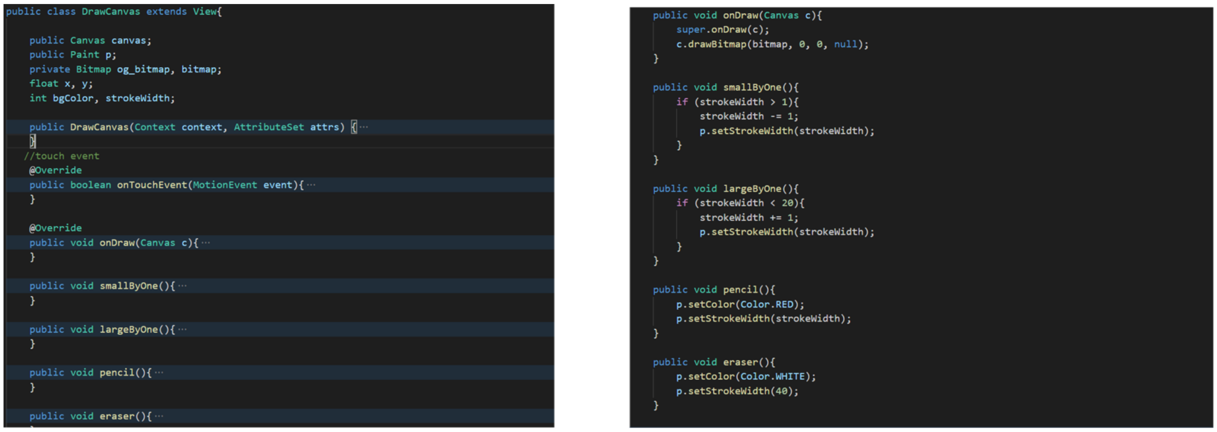
About how to build and run the APP, I learn that building project is a must. Then you can choose whether to use a virtual machine or run in a real phone.

3 Drawline Example and some exploration

I choose DrawLineSample for this lab. First I run the provided code and tries to grasp its main idea. I also read some blogs to know about Android proejct's structure.

- AndroidManifest.xml: records activities used in the APP, and state some privilege to be used in the APP.
- res Folder: some front end files, including icons, figures and xml files.
- java files: back end, different java activity classes refers to user's activity. Generate different response and show in the front end.

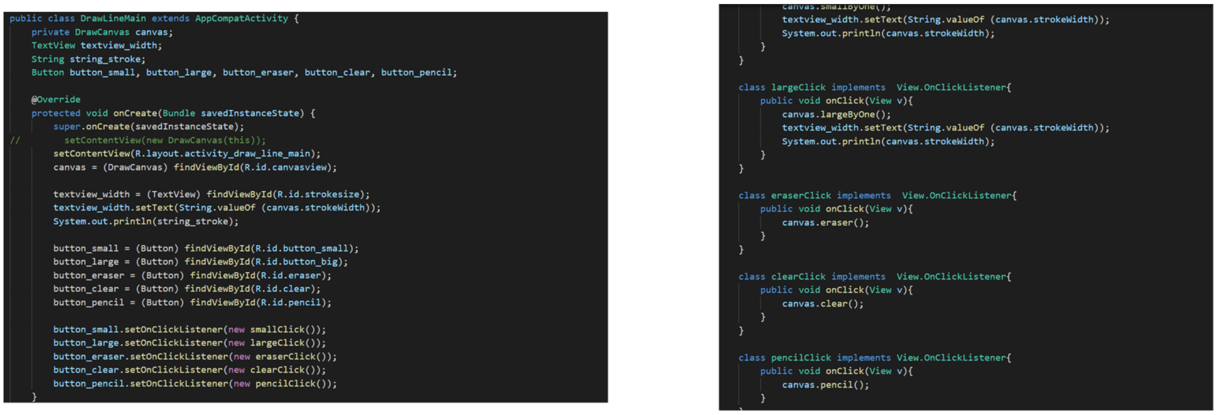
Here is the main code of DrawLineSample Project: As shown in the figure, Drawcanvas class is mainly used to implement the canvas and the Main Class is used to call the canvas and monitor user event. Once user touch or click somewhere in the canvas, the event listener will find such action(if such listener exists) and carry on corresponding tasks.



```java
public class DrawCanvas extends View{

    public Canvas canvas;
    public Paint p;
    private Bitmap og_bitmap, bitmap;
    float x, y;
    int bgColor, strokeWidth;

    public DrawCanvas(Context context, AttributeSet attrs) {···
    }
   //touch event
    @Override
    public boolean onTouchEvent(MotionEvent event){···
    }

    @Override
    public void onDraw(Canvas c){···
    }

    public void smallByOne(){···
    }

    public void largeByOne(){···
    }

    public void pencil(){···
    }

    public void eraser(){···
```

```java
    public void onDraw(Canvas c){
        super.onDraw(c);
        c.drawBitmap(bitmap, 0, 0, null);
    }

    public void smallByOne(){
        if (strokeWidth > 1){
            strokeWidth -= 1;
            p.setStrokeWidth(strokeWidth);
        }
    }

    public void largeByOne(){
        if (strokeWidth < 20){
            strokeWidth += 1;
            p.setStrokeWidth(strokeWidth);
        }
    }

    public void pencil(){
        p.setColor(Color.RED);
        p.setStrokeWidth(strokeWidth);
    }

    public void eraser(){
        p.setColor(Color.WHITE);
        p.setStrokeWidth(40);
    }
```

图 4: canvas class for paiting



```java
public class DrawLineMain extends AppCompatActivity {
    private DrawCanvas canvas;
    TextView textview_width;
    String string_stroke;
    Button button_small, button_large, button_eraser, button_clear, button_pencil;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
//        setContentView(new DrawCanvas(this));
        setContentView(R.layout.activity_draw_line_main);
        canvas = (DrawCanvas) findViewById(R.id.canvasview);

        textview_width = (TextView) findViewById(R.id.strokesize);
        textview_width.setText(String.valueOf (canvas.strokeWidth));
        System.out.println(string_stroke);

        button_small = (Button) findViewById(R.id.button_small);
        button_large = (Button) findViewById(R.id.button_big);
        button_eraser = (Button) findViewById(R.id.eraser);
        button_clear = (Button) findViewById(R.id.clear);
        button_pencil = (Button) findViewById(R.id.pencil);

        button_small.setOnClickListener(new smallClick());
        button_large.setOnClickListener(new largeClick());
        button_eraser.setOnClickListener(new eraserClick());
        button_clear.setOnClickListener(new clearClick());
        button_pencil.setOnClickListener(new pencilClick());
    }
```

```java
            canvas.smallByOne();
            textview_width.setText(String.valueOf (canvas.strokeWidth));
            System.out.println(canvas.strokeWidth);
        }
    }

    class largeClick implements  View.OnClickListener{
        public void onClick(View v){
            canvas.largeByOne();
            textview_width.setText(String.valueOf (canvas.strokeWidth));
            System.out.println(canvas.strokeWidth);
        }
    }

    class eraserClick implements  View.OnClickListener{
        public void onClick(View v){
            canvas.eraser();
        }
    }

    class clearClick implements  View.OnClickListener{
        public void onClick(View v){
            canvas.clear();
        }
    }

    class pencilClick implements View.OnClickListener{
        public void onClick(View v){
            canvas.pencil();
        }
```

图 5: Main class calling the canvas class and making response to user events

I have added some functions to this toy app. The drawlinesample projects provides a canvas where user can draw something on it. I have the following functions:

- changing stroke width, using "+" or "-" and you can find the stroke width is changed.
- eraser, use white color to cover the place. User can switch from options including "eraser" and "pencil".
- clear, clean the canvas, implemented by using a new map.

Below are some screenshots from the APP run on a HUAWEI phone.

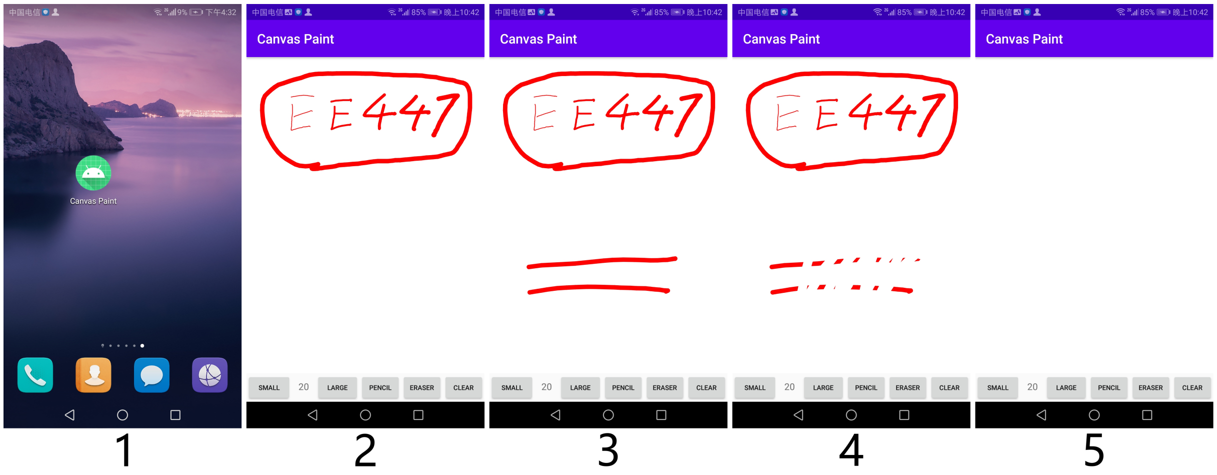


图 6: A view of how the app work.1)the icon 2)and 3)using different strokes to paint 4)using eraser to clean some areas 5)clear the whole canvas

4 Reference

- you can visit my github repository https://github.com/koalazf99/EE447Lab to view my lab codes revised from original provided code.
- Android introduction book《第一行代码》郭霖
- CSDN blog about how to use canvas class: https://blog.csdn.net/MofukYXM/article/details/89044636