Android 101:

Build an app from start to finish

Android Views and Using the Camera

Review - Declaring

```
int i = 0;

String hello = "Hi, there";

Button login = (Button)
this.findViewById(R.id.login);

Giraffe geoff = new Giraffe("Geoff", 10,
"brown");
```

Review - using methods

```
float random = Math.random()

int length = hello.length();

login.setText("Login!");

geoff.walk(3);
```

Welcome Activity

```
public void login(View view){
  if(usernameField.getText().toString().equals("")){

    AlertDialog.Builder builder = new AlertDialog.Builder(this);

    builder.setMessage("Please enter your username")
          .setCancelable(false)
          .setNegativeButton("OK", new
           DialogInterface.OnClickListener() {
             public void onClick(DialogInterface dialog, int id) {
                dialog.cancel();
              }
          });

   AlertDialog alert = builder.create();
    alert.show();
}
```

Opening a new page

WelcomeActivity.java

```
public void openPictureActivity(View view){

  Intent intent = new Intent(WelcomeActivity.this,
  TakePictureActivity.class);

  startActivity(intent);

}
```

Opening a new page

AndroidManifest.xml

```
<activity
android:name=".views.GalleryActivity"
android:configChanges="keyboardHidden|orientation"
android:screenOrientation="portrait"/>
```

takepicture.xml

We need:

TextView that says "Upload an image"

EditText with id titleField

EditText with id descriptionField

Button with onClick takePicture

Button with onClick choosePicture

ImageView with id pictureFromCamera

Button with onClick savePicture

takepicture.xml

```
<TextView android:layout_width="fill_parent"
          android:layout_height="wrap_content"
          android:text="Upload an image!"
          android:textColor="@color/GREEN"/>
```

```
<EditText android:layout_width="fill_parent"
          android:layout_height="wrap_content"
          android:inputType="textCapSentences"
          android:hint="Add title"
          android:id="@+id/titleField"
          android:textColor="@color/PURPLE"/>
```

takepicture.xml

```
<LinearLayout
    android:layout_width="fill_parent"
    android:layout_height="48dip"
    android:orientation="horizontal">

    <Button android:layout_width="fill_parent"
            android:layout_height="fill_parent"
            android:text="Take Picture"
            android:onClick="takePicture"
            android:layout_weight="1"
            />

   <Button android:layout_width="fill_parent"
            android:layout_height="fill_parent"
            android:text="Choose Picture"
            android:onClick="choosePicture"
            android:layout_weight="1"/>

</LinearLayout>
```

takepicture.xml

```
<ImageView android:layout_height="wrap_content"
           android:layout_width="fill_parent"
           android:id="@+id/pictureFromCamera"
           android:layout_weight="1" />

<Button android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="Done"
        android:id="@+id/savePicture"
        android:onClick="save" />
```

Getting ready for the camera

Creating a sdcard for an emulator

In your terminal, navigate to where you saved the Android sdk

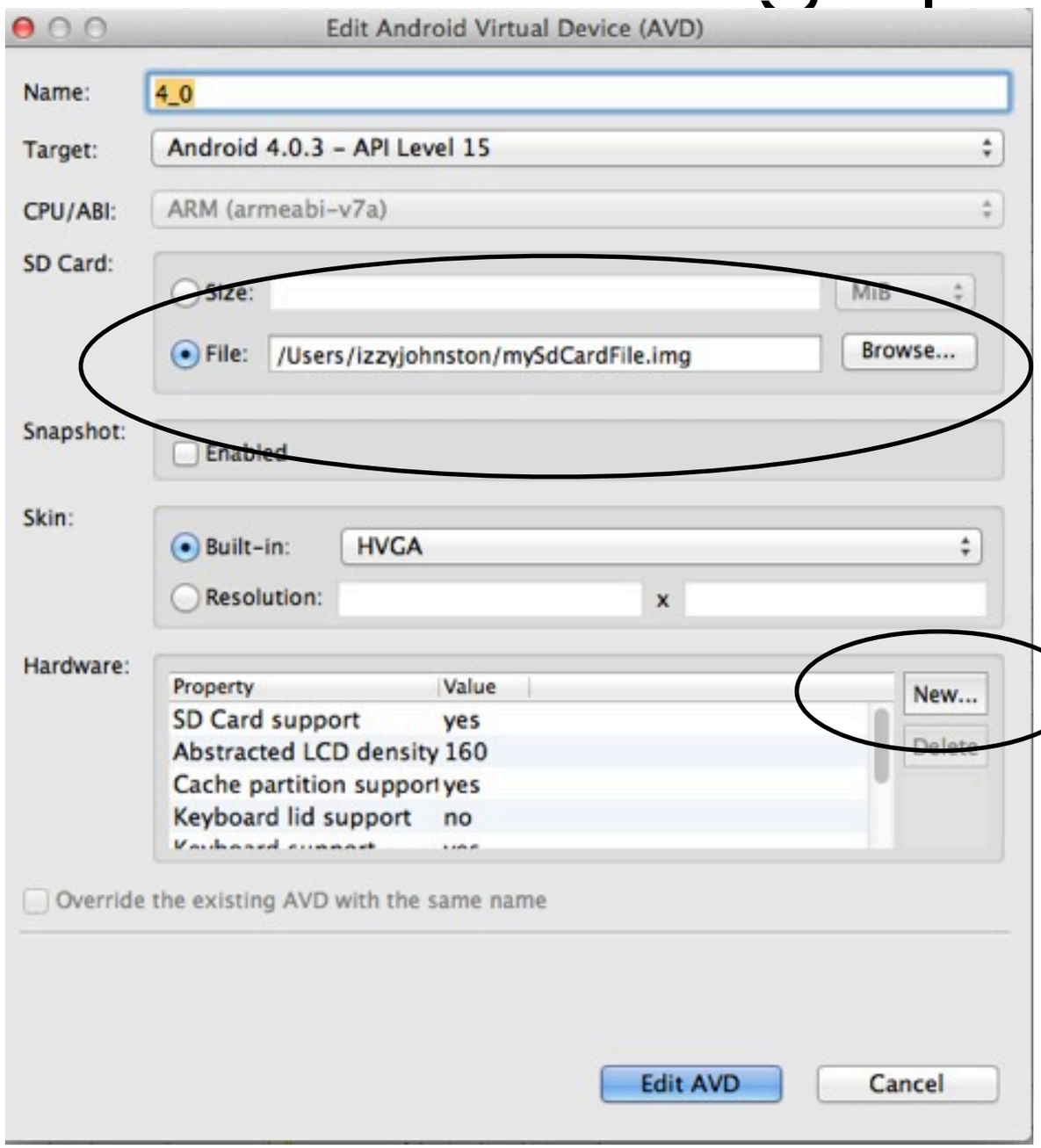
Go into the folder "tools"

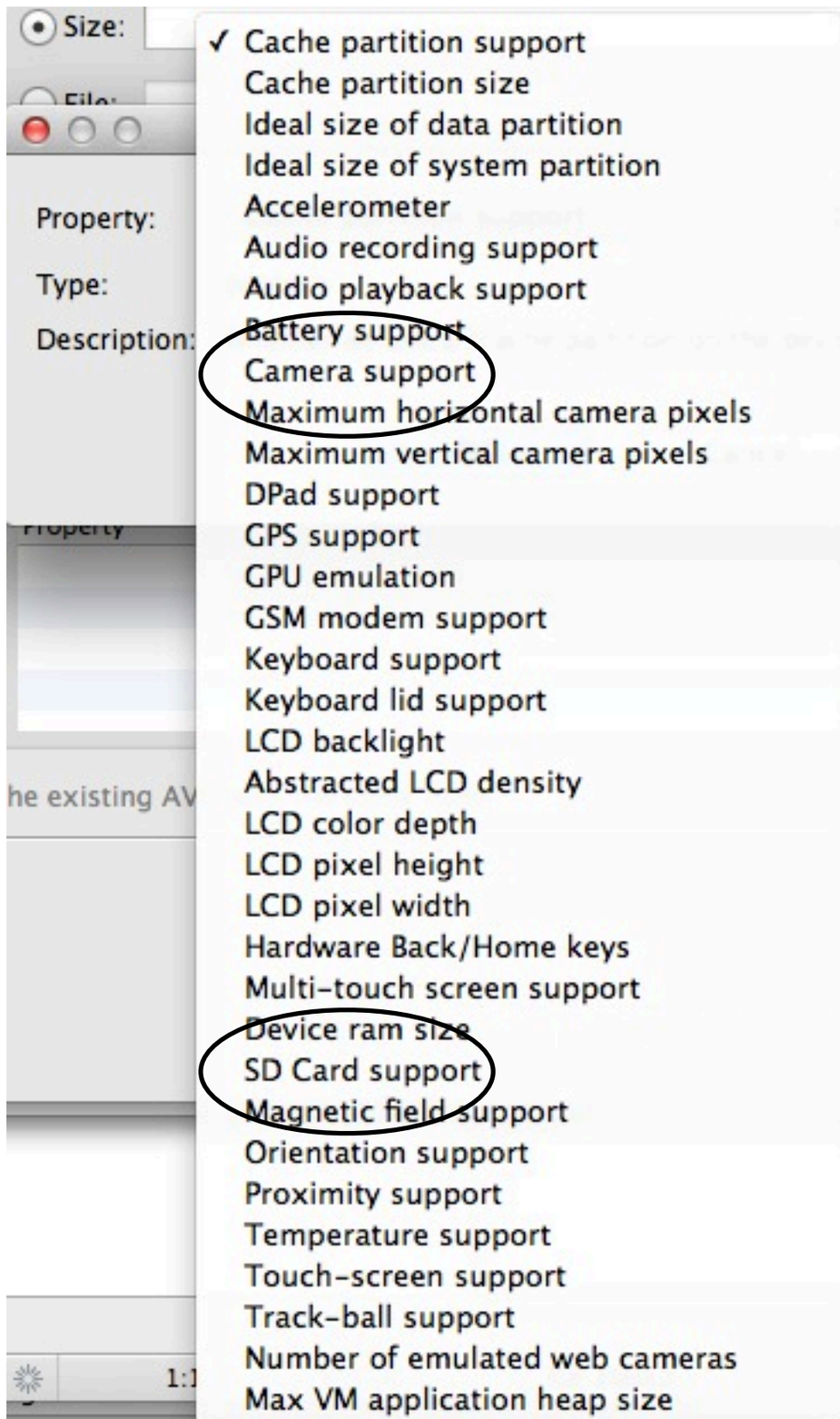
Run this script

```
mksdcard -l mySdCard 1024M mySdCardFile.img
```

Getting ready for the camera

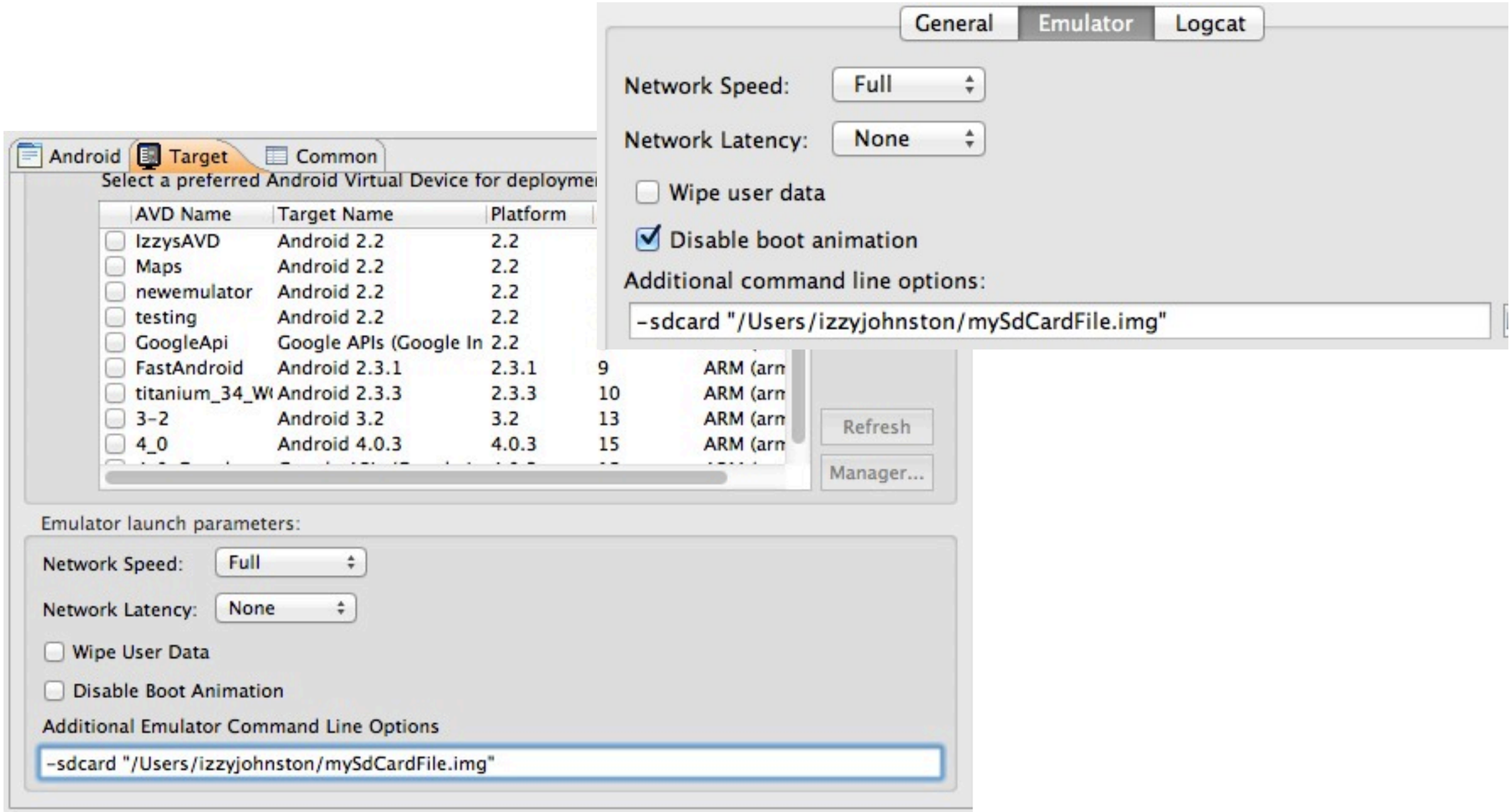
Setting up the emulator





Getting ready for the camera

Setting up the emulator



Getting ready for the camera

AndroidManifest.xml

```
<uses-permission
  android:name="android.permission.CAMERA"/>

<uses-permission
  android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
```

TakePictureActivityyy

Declare variables

```
private GirlDevelopIt app;

private final int ACTIVITY_TAKE_PHOTO = 1;
private final int ACTIVITY_SELECT_PHOTO = 2;

private ImageView pictureFromCamera;
private EditText titleField;
private EditText descriptionField;
private Button savePicture;
private String pathToImage ="";
```

TakePictureActivityyy

onCreate

```
super.onCreate(savedInstanceState);

setContentView(R.layout.takepicture);

this.app = (GirlDevelopIt)getApplicationContext();

initElements();
```

TakePictureActivityyy

initElements()

```
pictureFromCamera=
    (ImageView)this.findViewById(R.id.pictureFromCamera);

titleField = (EditText)this.findViewById(R.id.titleField);
```

```
descriptionField = (EditText)this.findViewById(R.id.descriptionFi

savePicture = (Button) this.findViewById(R.id.savePicture);

if(app.getUsername()==null || app.getUsername().equals("")){
  savePicture.setEnabled(false);
}
```

TakePictureActivityyy

takePicture(View v)

```
final Intent intent =
  new Intent(android.provider.MediaStore.ACTION_IMAGE_CAPTURE);

intent.putExtra(MediaStore.EXTRA_OUTPUT,
  Uri.fromFile(Utils.getTempFile(getApplicationContext())));

startActivityForResult(intent, ACTIVITY_TAKE_PHOTO);
```

TakePictureActivityyy

choosePicture(View v)

```
final Intent intent =
    new Intent(Intent.ACTION_PICK,
      android.provider.MediaStore.Images.Media.INTERNAL_CONTENT_UR

startActivityForResult(intent, ACTIVITY_SELECT_PHOTO);
```

TakePictureActivityyy

onActivityResult(int requestCode, int resultCode, Intent data)

```
super.onActivityResult(requestCode, resultCode, data);

if(resultCode == RESULT_CANCELED){
   return;
}
```

TakePictureActivity

onActivityResult(int requestCode, int resultCode, Intent data)

```
if(requestCode == ACTIVITY_TAKE_PHOTO){

  try{
      pathToImage=Utils.getTempFile(getApplicationContext()).get
              AbsolutePath();

      File takenFile=
              Utils.getTempFile(getApplicationContext());

  Bitmap thumbnailBmp = Utils.decodeFile(takenFile);

  pictureFromCamera.setImageBitmap(thumbnailBmp);

  pictureFromCamera.setScaleType(ImageView.ScaleType.CENTER
          _INSIDE);
```

TakePictureActivity

onActivityResult(int requestCode, int resultCode, Intent data)

```
try
 {
  FileOutputStream ostream =
      new FileOutputStream(takenFile);

  thumbnailBmp.compress(Bitmap.CompressFormat.JPEG, 100,
            ostream);
      ostream.close();
    }
    catch (Exception exp){
      Log.e("TakePhoto", exp.getMessage());
    }
    System.gc();
  }
  catch(Exception exp){
    Log.e("TakePhoto", exp.getMessage());
  }
}
```

TakePictureActivity

onActivityResult(int requestCode, int resultCode, Intent data)

```
if(requestCode == ACTIVITY_SELECT_PHOTO){
  try{
   System.gc();
   pathToImage = Utils.getPath(data.getData(),
       getContentResolver());
   File selFile=new File(pathToImage);
   Bitmap thumbnailBmp = Utils.decodeFile(selFile);
    pictureFromCamera.setImageBitmap(thumbnailBmp);
    pictureFromCamera.setScaleType(ImageView.ScaleType.CENTER_IN
    SIDE);

  }
  catch(Exception exp){
     Log.e("SelectPhoto", exp.getMessage());
  }
}
```

Next Week!

Building an Image class with getters and setters

Saving images and their data

Showing the saved images in a list

Questions?