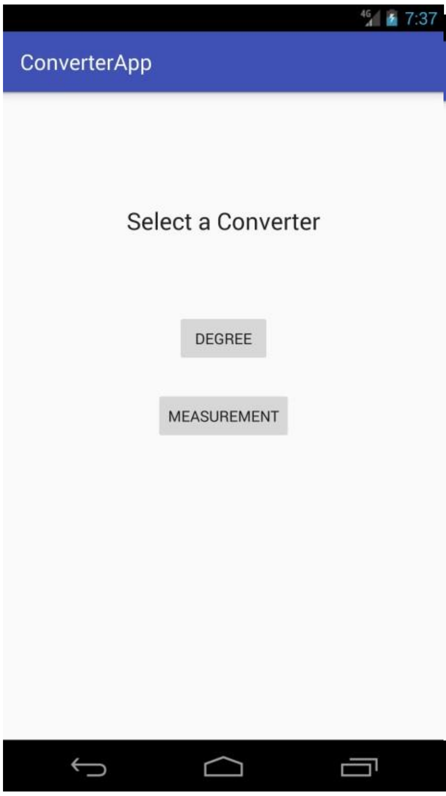
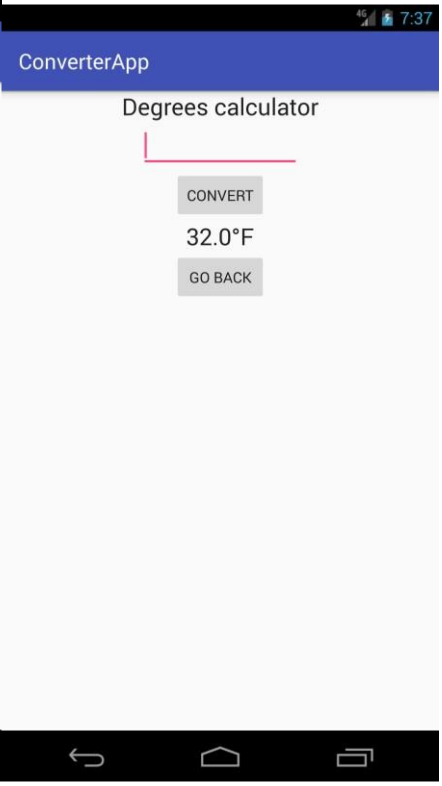
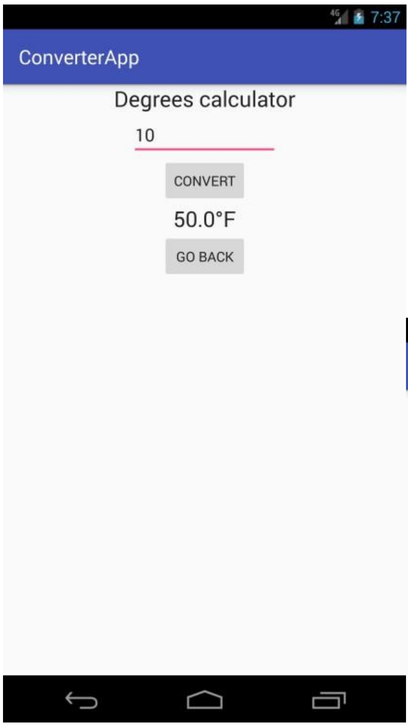
Cos30017 Steven Efthimiadis 1627406

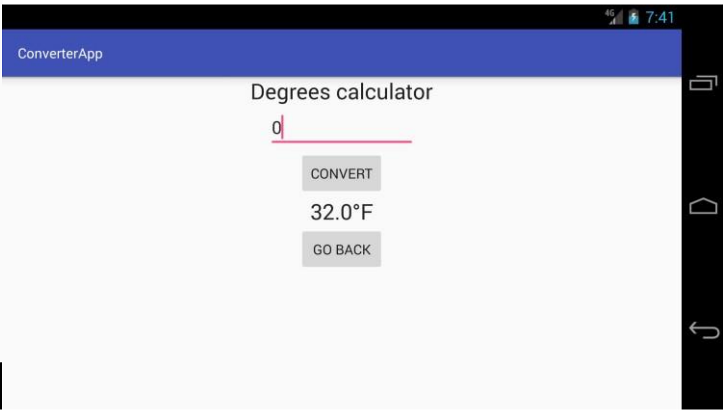
Assignment 3

Task 1

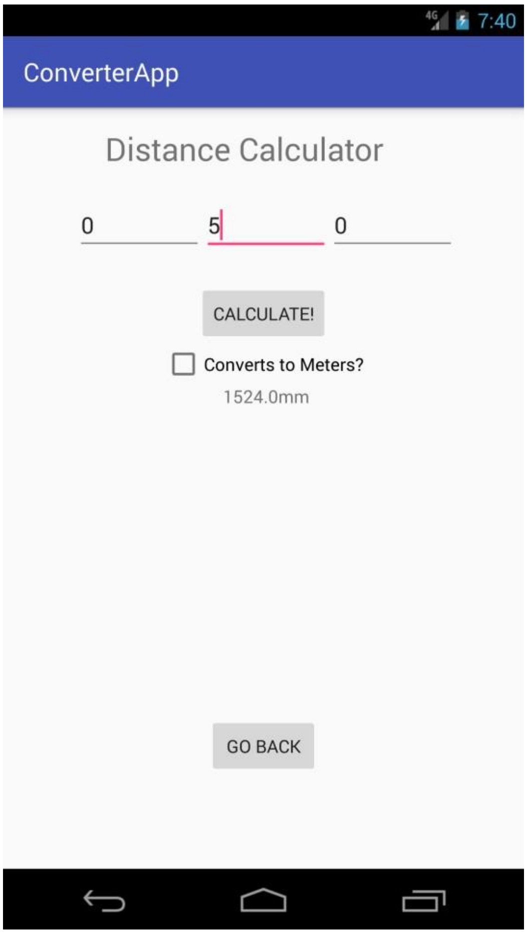


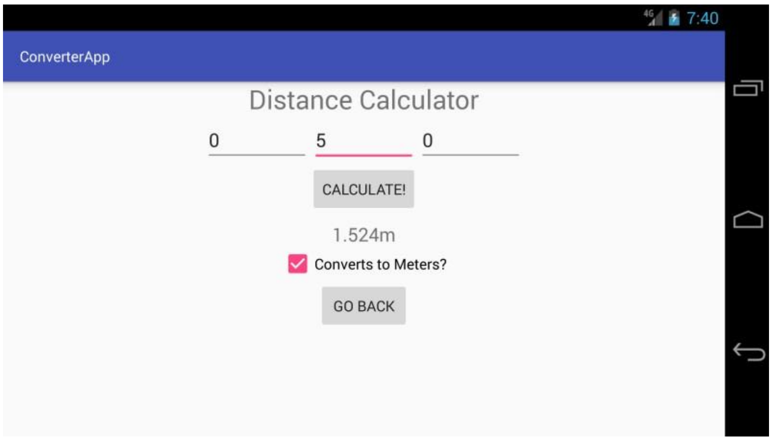


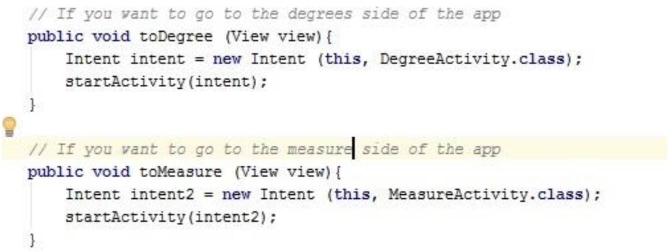




Cos30017 Steven Efthimiadis 1627406



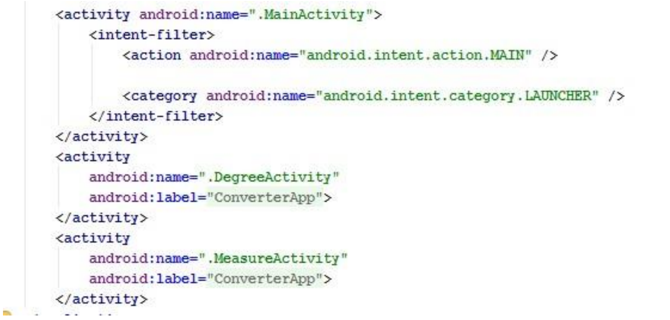




```
// If you want to go to the degrees side of the app
public void toDegree (View view){
    Intent intent = new Intent (this, DegreeActivity.class);
    startActivity(intent);
}

// If you want to go to the measure side of the app
public void toMeasure (View view){
    Intent intent2 = new Intent (this, MeasureActivity.class);
    startActivity(intent2);
}
```

If you have multiple activities you need to reach. You need to set up new intents so you can cleanly change app activities and when you change activities you don't return to the home activity

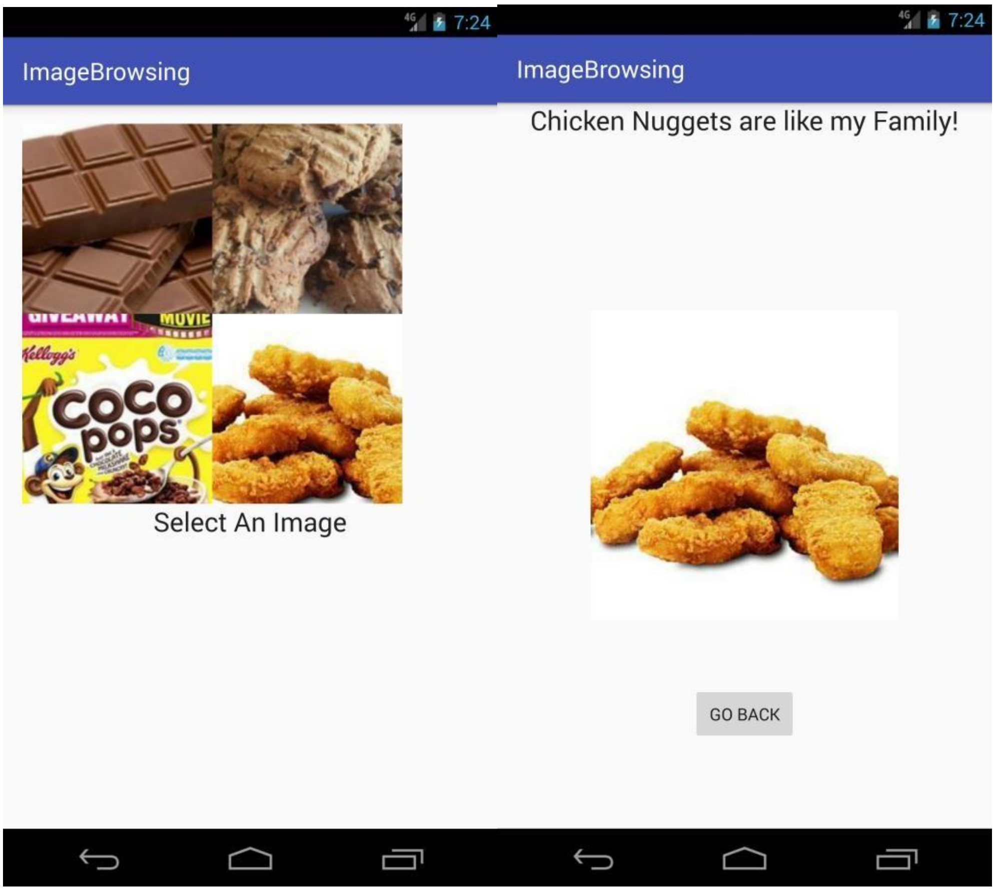


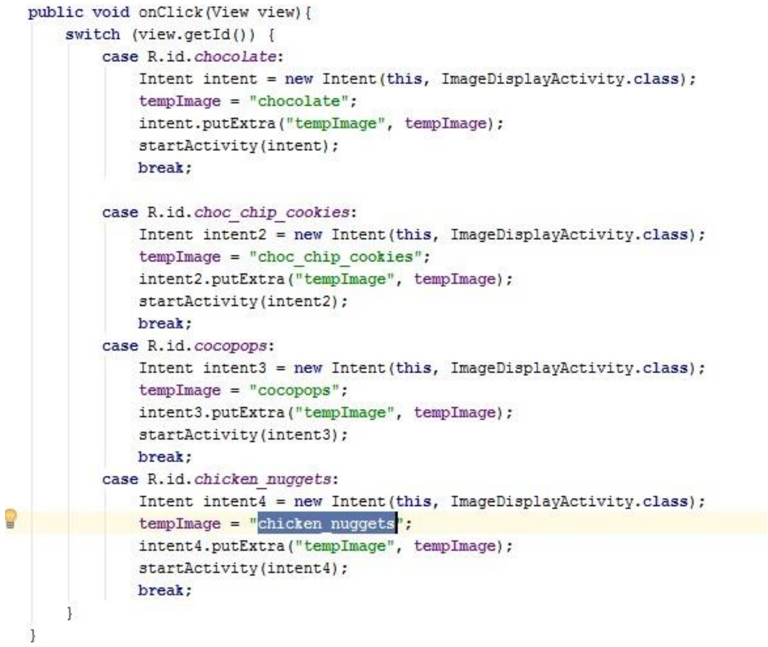
```
<activity android:name=".MainActivity">
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />

        <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
</activity>
<activity
    android:name=".DegreeActivity"
    android:label="ConverterApp">
</activity>
<activity
    android:name=".MeasureActivity"
    android:label="ConverterApp">
</activity>
```

When you're in the manifest.xml, you need to setup multiple activities so when you are changing intents the app can grab the activity from the manifest file.

Task 2

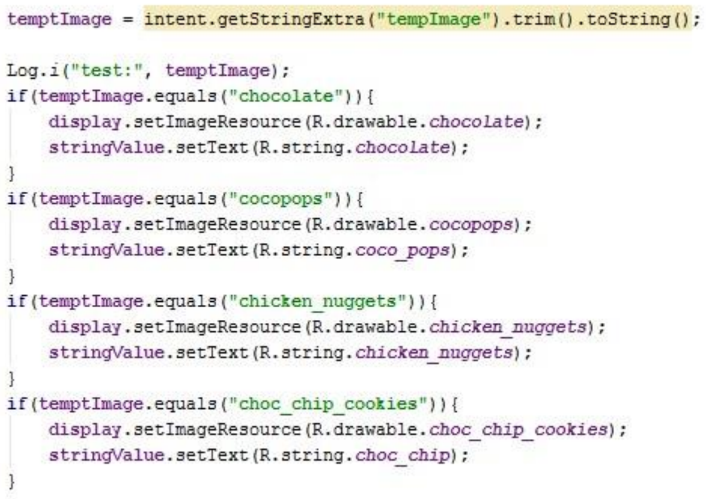




```
public void onClick(View view){
    switch (view.getId()) {
        case R.id.chocolate:
            Intent intent = new Intent(this, ImageDisplayActivity.class);
            tempImage = "chocolate";
            intent.putExtra("tempImage", tempImage);
            startActivity(intent);
            break;

        case R.id.choc_chip_cookies:
            Intent intent2 = new Intent(this, ImageDisplayActivity.class);
            tempImage = "choc_chip_cookies";
            intent2.putExtra("tempImage", tempImage);
            startActivity(intent2);
            break;
        case R.id.cocopops:
            Intent intent3 = new Intent(this, ImageDisplayActivity.class);
            tempImage = "cocopops";
            intent3.putExtra("tempImage", tempImage);
            startActivity(intent3);
            break;
        case R.id.chicken_nuggets:
            Intent intent4 = new Intent(this, ImageDisplayActivity.class);
            tempImage = "chicken_nuggets";
            intent4.putExtra("tempImage", tempImage);
            startActivity(intent4);
            break;
    }
}
```

This is where we send a message to the next activity to enable it to show the correct image and text



```
temptImage = intent.getStringExtra("tempImage").trim().toString();

Log.i("test:", temptImage);
if(temptImage.equals("chocolate")){
    display.setImageResource(R.drawable.chocolate);
    stringValue.setText(R.string.chocolate);
}
if(temptImage.equals("cocopops")){
    display.setImageResource(R.drawable.cocopops);
    stringValue.setText(R.string.coco_pops);
}
if(temptImage.equals("chicken_nuggets")){
    display.setImageResource(R.drawable.chicken_nuggets);
    stringValue.setText(R.string.chicken_nuggets);
}
if(temptImage.equals("choc_chip_cookies")){
    display.setImageResource(R.drawable.choc_chip_cookies);
    stringValue.setText(R.string.choc_chip);
}
```

This is where we retrieve the message and make sure there are no hidden nasties. So we trim and covert it to string to make sure it is returning a string and if the string equals one of these four it will
display the correct image and picture

Task 3

a) Why is the intent messaging facility considered as a late run-time binding be

An intent is basically a message sent between different components. So it’s considered late time late runtime binding because when you want to send a message to the next activity you to send it to the compiler for the next activity to access.

b) There are two primary contents an intent can hold which are passive data structure. The action is basically and general action which the intent needs to perform. The data which is any data required to operate. There are secondary contents which can be used. The category is additional information about the action needs to accomplish. Type stipulates an explicit type of data of the intent data (MIME type). The component is the specific name of the component class you need to use for the intent. Extras is the Bundle which is any additional information you need to send to the next intent so the activity can perform the required task.

c) It’s passive because you generate any data you want send to the next and store it in the compiler for the next activity. The next activity will then search the compiler for any stored data and if it is relevant to its activity. There is no communication between each activity so therefor it is passive.

d)

```
Intent intent = new Intent (this, MeasureActivity.class);
startActivity(intent);
```

This is how you would call a new Intent. You would firstly say what intent object you are currently in. This is achieved by returning “`this`”. Then you would input the next activity you wish to access by placing the class of the activity in the Intent “`MeasureActivity.class`”. Now the intent message to change to a new intent is saved in the java compiler, so all you need to do is access the compiler by using “`startActivity(intent)`” and then you will switch to the new activity.

Reference:

Free practice test , mock test, driving test, interview questions. 2011. Free practice test, mock test, driving test, interview questionsAndroid Tutorial - Understanding Intent » Free practice test , mock test, driving test, interview questions. [ONLINE] Available at: http://www.skill-guru.com/blog/2011/01/18/how-to-move-from-one-screen-to-another-through-intent/. [Accessed 22 September 2016].