10/26/2021

Mobile Application Development

Lab3 Manual

Lab Manual 3

Safdar Hussain, BSCS-VII-A

CMS: 023-18-0059

PRACTICE ACTIVITIES:

Activity 1 : Create an app that explores the life-cycle of an activity.

∙ The main objective of this app is to experience callback methods first hand. Write definition of all these methods and observe the results.

∙ onCreate()

∙ onStart()

∙ onResume()

∙ onPause()

∙ onStop()

∙ onRestart()

∙ onDestroy()

1. What callbacks are called when an app is first launched?

onCreate() , onStart(), onResume() callbacks are called.

1. What callbacks occur when Home is pressed?

onPause and onStop callbacks are called.

1. What callbacks occur when an app is restarted from the launcher?

onRestart, onStart, and onResume callbacks are called.

1. What callbacks occur when the device is rotated?

onPause()

onStop()

onDestory()

onCreated()

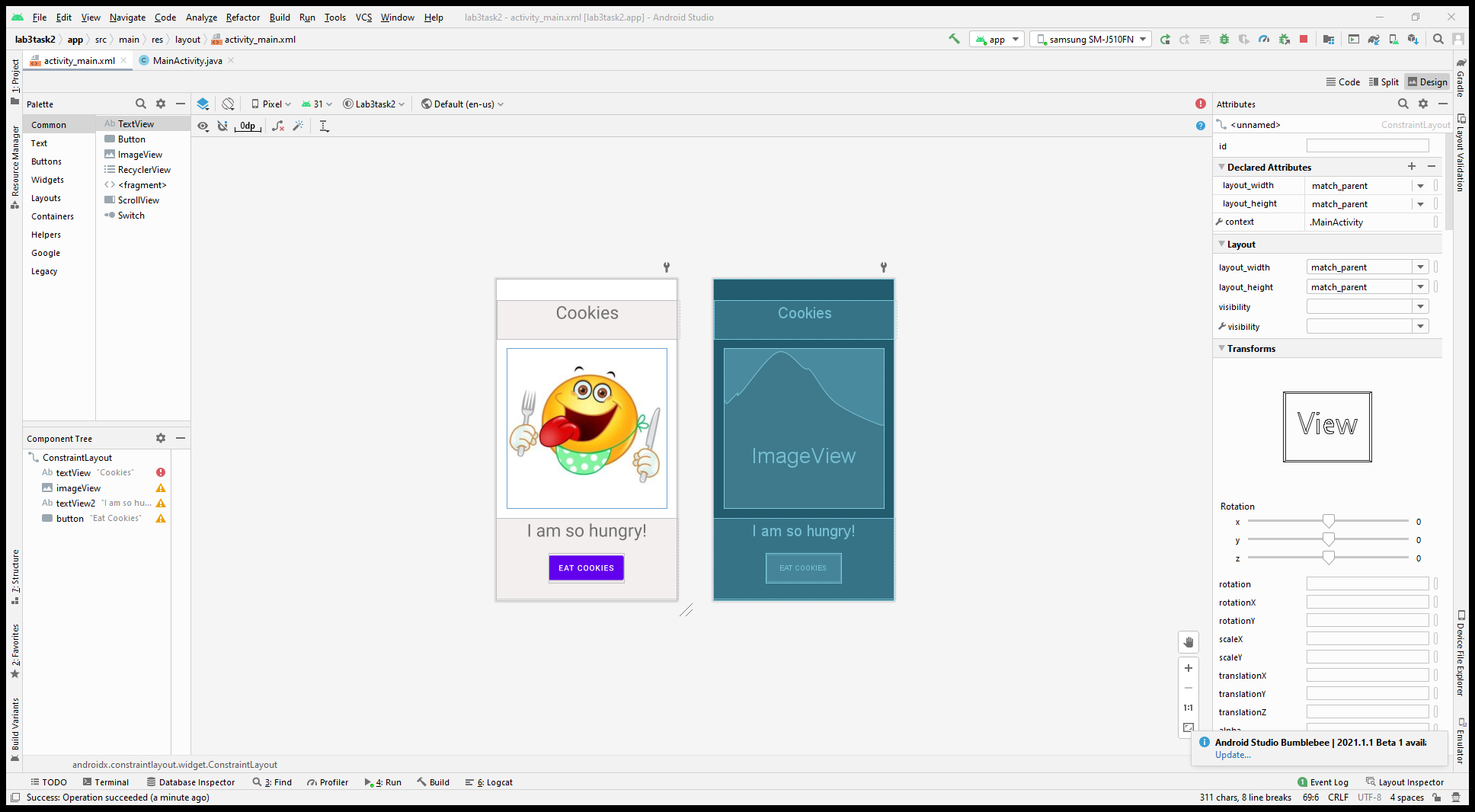
onStarted()

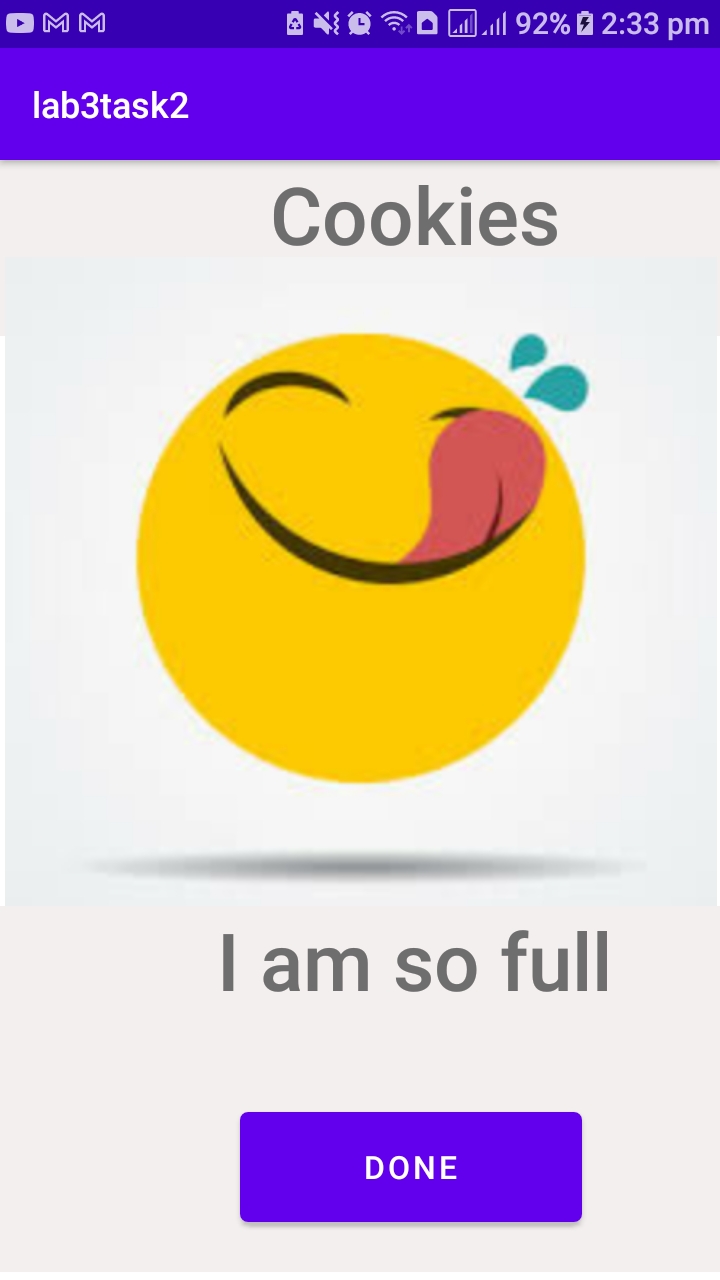
onResume()

.

Activity 2 : Develop an app to describe the mood.

public void eatCookies(View view){  
 TextView text = findViewById(R.id.*textView2*);  
 Button btn = findViewById(R.id.*button*);  
 ImageView img = findViewById(R.id.*imageView*);  
  
 text.setText("I am so full");  
 btn.setText("Done");  
 img.setImageResource(R.drawable.*full*);  
}





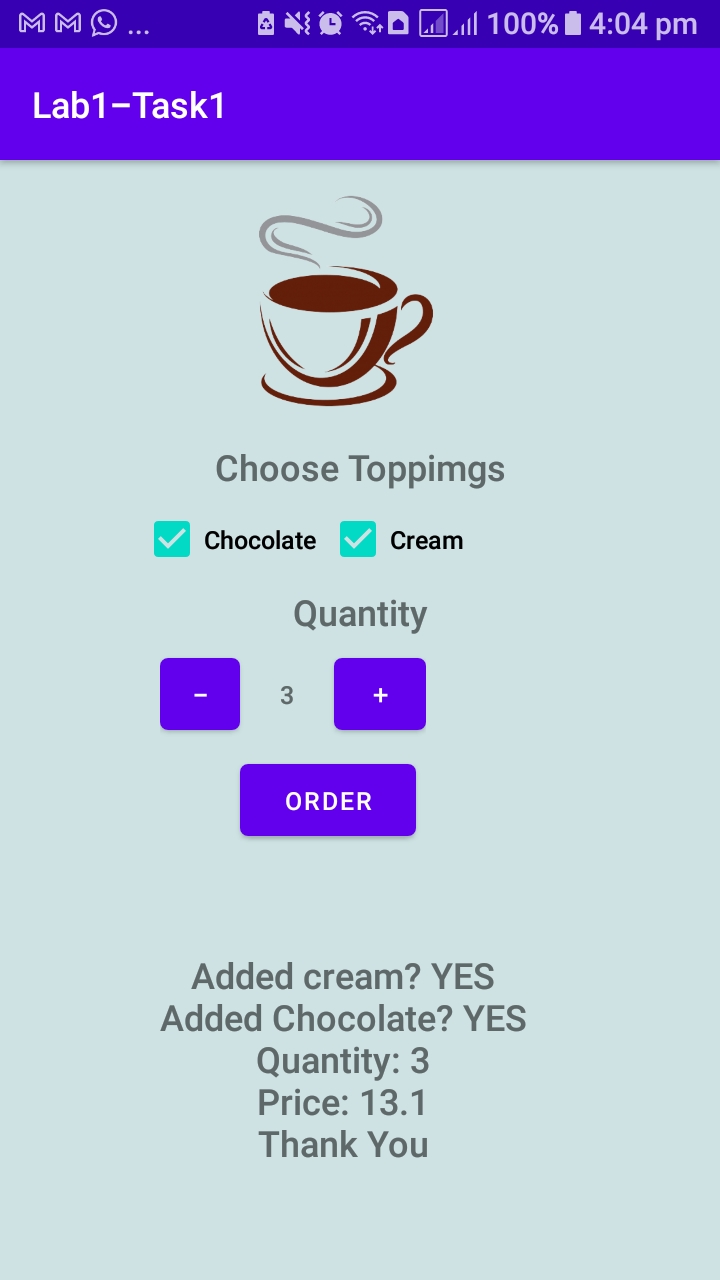
Activity 3 : Build the Coffee Ordering app shown below.

- Assume a single coffee costs $4.00. Charge an additional $1.00 for chocolate and $.50 for whipped cream, per cup.

- Background Color : #f7eac1

package com.example.lab1\_task1;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.os.Bundle;  
import android.view.View;  
import android.widget.CheckBox;  
import android.widget.TextView;  
  
public class MainActivity extends AppCompatActivity {  
  
 public TextView summary;  
 public TextView quantity;  
 public CheckBox cream;  
 public CheckBox chocolate;  
  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*coffe\_order*);  
 quantity = findViewById(R.id.*textViewQuantity*);  
 summary= findViewById(R.id.*textViewSummary*);  
 cream = findViewById(R.id.*checkBoxCream*);  
 chocolate = findViewById(R.id.*checkBoxChocolate*);  
 }  
  
 public void onIncrease(View view){  
 quantity.setText((Integer.*parseInt*(quantity.getText().toString()) + 1) + "");  
 }  
  
 public void onDecrease(View view){  
 quantity.setText((Integer.*parseInt*(quantity.getText().toString()) - 1) + "");  
 }  
  
 public void onOrder(View view){  
 float bill = 0;  
 String creem = "NO";  
 String choco = "NO";  
  
 if(Integer.*parseInt*(quantity.getText().toString()) != 0){  
  
 if(cream.isChecked()){  
 bill += 0.1;  
 creem = "YES";  
 }  
 if(chocolate.isChecked()){  
 bill += 1;  
 choco = "YES";  
 }  
  
 bill += Integer.*parseInt*(quantity.getText().toString()) \* 4;  
 summary.setText("\nAdded cream? " + creem + "\nAdded Chocolate? " + choco +  
 "\nQuantity: " + quantity.getText().toString() + "\nPrice: " + bill + "\nThank You");  
 }  
  
 }  
}

Output:



Activity 4: Calculator

Create the simple calculator shown below. This app should use a TableLayout.  
  
public class MainActivity extends AppCompatActivity {  
  
 TextView textView;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity7);  
 System.*out*.println("Activity Created");  
 textView = findViewById(R.id.textbox);  
  
  
 }  
public void numbers(View view) {  
 Button btn = (Button) view;  
 textView.setText(textView.getText() + btn.getText().toString());  
 }  
  
 public void clear(View view) {  
 textView.setText("");  
 }  
  
 public void operation(View view) {  
 Button btnOp = (Button) view;  
 String text = textView.getText().toString();  
 if (text.contains("+") || text.contains("-") || text.contains("\*") || text.contains("/") || text.contains("%"))  
 Toast.makeText(this, "You can not use multiple operators", Toast.LENGTH\_SHORT).show();  
 else  
 textView.setText(textView.getText() + btnOp.getText().toString());  
 }  
  
 public void equal(View view) {  
 String[] numbers = textView.getText().toString().split("-|\\+|x|/|%");  
 if (textView.getText().toString().contains("+"))  
 textView.setText(textView.getText() + "\n" + (Integer.*parseInt*(numbers[0]) + Integer.*parseInt*(numbers[1])) + "");  
 else if (textView.getText().toString().contains("-"))  
 textView.setText(textView.getText() + "\n" + (Integer.*parseInt*(numbers[0]) - Integer.*parseInt*(numbers[1])) + "");  
 else if (textView.getText().toString().contains("/"))  
 textView.setText(textView.getText() + "\n" + (Float.*parseFloat*(numbers[0]) / Float.*parseFloat*(numbers[1])) + "");  
 else if (textView.getText().toString().contains("x"))  
 textView.setText(textView.getText() + "\n" + (Integer.*parseInt*(numbers[0]) \* Integer.*parseInt*(numbers[1])) + "");  
 else if (textView.getText().toString().contains("%"))  
 textView.setText(textView.getText() + "\n" + (Integer.*parseInt*(numbers[0]) % Integer.*parseInt*(numbers[1])) + "");  
 }  
  
}

Output:

