

Practical 1

Create a “Hello World” application that will display “Hello World” in the middle of the Screen in the Emulator.

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:fontFamily="serif-monospace"
        android:text="Hello World!"
        android:textColor="@color/black"
        android:textSize="25sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

Output



Hello World!



Practical 2

Create an application to display various Android Activity Lifecycle Phases.

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/counter"
        android:layout_width="0dp"
        android:layout_height="0dp"
        android:layout_marginStart="16dp"
        android:layout_marginTop="16dp"
        android:layout_marginEnd="16dp"
        android:layout_marginBottom="16dp"
        android:background="@android:color/holo_blue_light"
        android:gravity="center"
        android:text="@string/counter"
        android:textAlignment="center"
        android:textColor="@color/black"
        android:textSize="250sp"
        app:layout_constraintBottom_toTopOf="@+id/decrement"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/increment" />

    <Button
        android:id="@+id/increment"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:layout_marginTop="16dp"
        android:layout_marginEnd="16dp"
        android:onClick="count"
        android:text="@string/increment"
        android:textSize="30sp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
```

```

<Button
    android:id="@+id/decrement"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="16dp"
    android:layout_marginEnd="16dp"
    android:layout_marginBottom="16dp"
    android:onClick="count"
    android:text="@string/decrement"
    android:textSize="30sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent" />

</androidx.constraintlayout.widget.ConstraintLayout>

```

MainActivity.java

```

package com.example.counter;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {

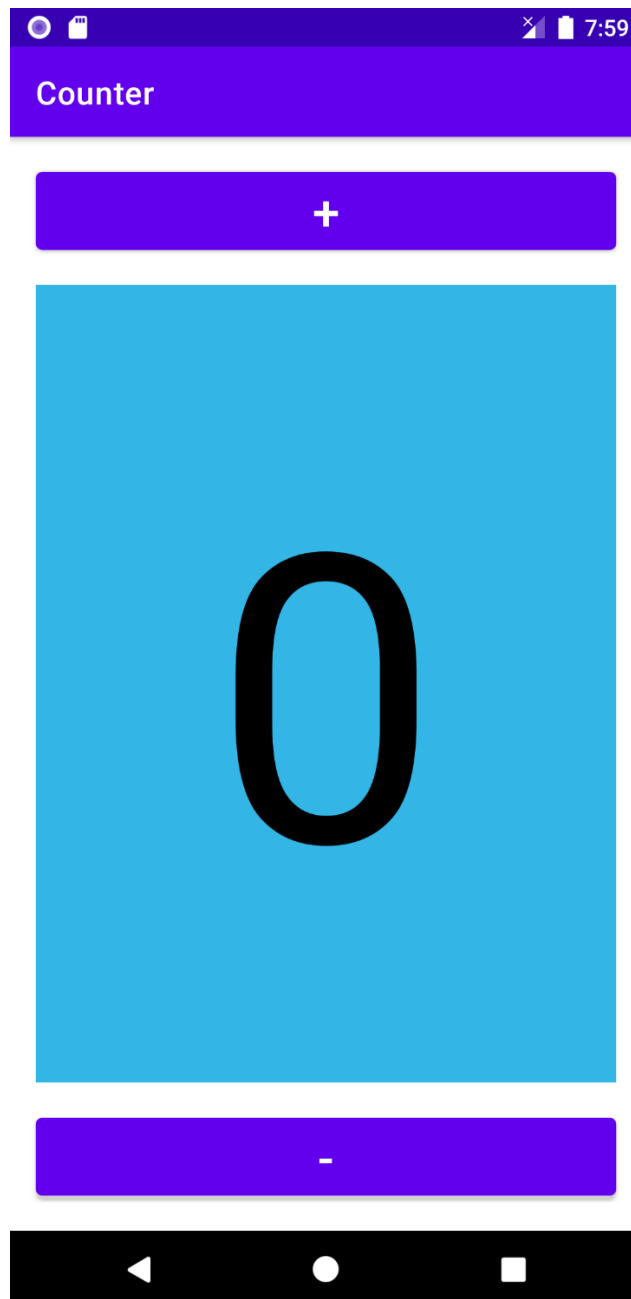
    TextView counter;
    String counterText;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        /* getting the TextView */
        counter = (TextView) findViewById(R.id.counter);
    }

    public void count(View view) {
        Button button = (Button) (view);
        String type = button.getText().toString();
        switch (type) {
            case "+":
                counterText = counter.getText().toString();
                counter.setText(Integer.toString(Integer.parseInt(counterText) + 1));
                break;
            case "-":
                counterText = counter.getText().toString();
                if (!counterText.equals("0")) {
                    counter.setText(Integer.toString(Integer.parseInt(counterText) -
1));

```

```
        }  
        break;  
    default:  
        break;  
}  
}  
}
```

Output





Practical 3

Create an application with first activity having an EditText and a Send Button. Make use of Explicit Intent to send the Text of the EditText from first activity to second activity and display it in a TextView there on clicking the Button.

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <Button
        android:id="@+id/send"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp"
        android:layout_marginEnd="16dp"
        android:text="@string/send"
        android:onClick="sendMessage"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toEndOf="@+id/message"
        app:layout_constraintTop_toTopOf="parent" />

    <EditText
        android:id="@+id/message"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:layout_marginTop="16dp"
        android:layout_marginEnd="16dp"
        android:ems="10"
        android:inputType="textPersonName"
        android:hint="@string/message"
        app:layout_constraintEnd_toStartOf="@+id/send"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

MainAcitivty.java

```
package com.example.explicitintent;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;

public class MainActivity extends AppCompatActivity {

    public static final String EXTRA_MESSAGE = " /::|::/";
    EditText message;
    String messageText;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        message = (EditText)findViewById(R.id.message);
    }

    public void sendMessage(View view) {
        messageText = message.getText().toString();
        Intent intent = new Intent(this,MessageActivity.class);
        intent.putExtra(EXTRA_MESSAGE,messageText);
        startActivity(intent);
    }
}
```

activity_message.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MessageActivity">

    <TextView
        android:id="@+id/getMessage"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:fontFamily="monospace"
        android:text="TextView"
        android:textSize="30sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent">
```



```

        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
    </androidx.constraintlayout.widget.ConstraintLayout>

```

MessageActivity.java

```

package com.example.explicitintent;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.widget.TextView;

public class MessageActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_message);
        Intent intent = getIntent();
        String recievedMessage = intent.getStringExtra(MainActivity.EXTRA_MESSAGE);
        TextView message = (TextView)findViewById(R.id.getMessage);
        message.setText(recievedMessage);
    }
}

```

AndroidManifest.xml

```

<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.explicitintent">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.ExplicitIntent">
        <activity android:name=".MessageActivity"
            android:label="Message"
            android:parentActivityName=".MainActivity">
            <meta-data
                android:name="android.support.PARENT_ACTIVITY"
                android:value=".MainActivity"/>
        </activity>
        <activity android:name=".MainActivity">
            <intent-filter>

```

```
        <action android:name="android.intent.action.MAIN" />

        <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
</activity>
</application>
</manifest>
```

Output





Hello World

SEND



Hello World



Practical 4

Create an application with first activity having an EditText and a Send Button. Make use of Implicit Intent that uses ACTION_SEND action and let user select an App from App Chooser and Navigate to it on clicking the Button.

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:layout_marginTop="16dp"
        android:layout_marginEnd="16dp"
        android:text="@string/button"
        android:onClick="sendAction"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toEndOf="@+id/message"
        app:layout_constraintTop_toTopOf="parent" />

    <EditText
        android:id="@+id/message"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:layout_marginTop="16dp"
        android:ems="10"
        android:inputType="textPersonName"
        android:hint="@string/message"
        app:layout_constraintEnd_toStartOf="@+id/button"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

MainActivity.java

```
package com.example.implicitintent;

import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ShareCompat;

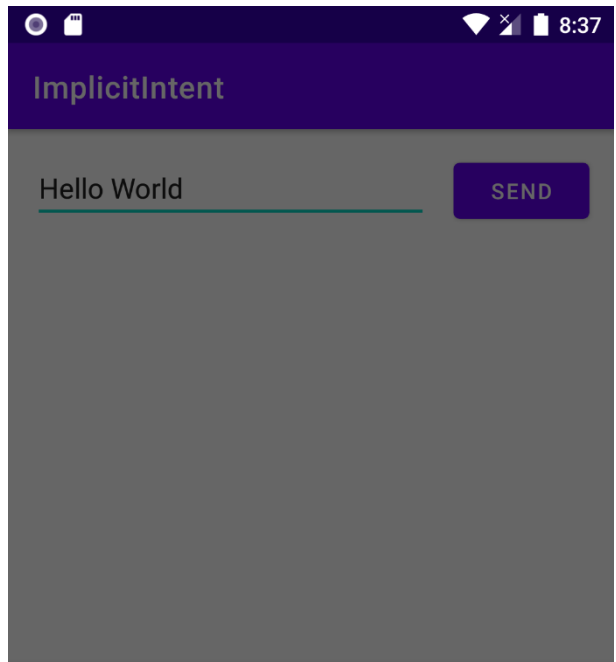
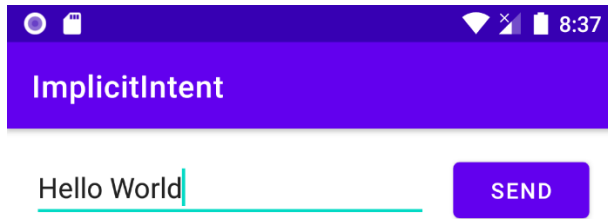
import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.EditText;

public class MainActivity extends AppCompatActivity {

    EditText message;
    String messageText;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        message = (EditText)findViewById(R.id.message);
    }

    public void sendAction(View view) {
        messageText = message.getText().toString();
        Intent intent = new Intent();
        intent.setAction(Intent.ACTION_SEND);
        intent.setType("text/plain");
        intent.putExtra(Intent.EXTRA_TEXT,messageText);
        startActivity(Intent.createChooser(intent,
        getResources().getText(R.string.app_name)));
    }
}
```

Output



ImplicitIntent



Gmail



Messages



Copy to
clipboard



Save to Drive



Practical 5

Create a Spinner with the Strings taken from the resource folder (res>>values) and on changing the Spinner Selection or Value Images will change on a ImageView.

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <Spinner
        android:id="@+id/planets"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:layout_marginEnd="16dp"
        app:layout_constraintBottom_toTopOf="@+id/imageView"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <ImageView
        android:id="@+id/imageView"
        android:layout_width="275dp"
        android:layout_height="400dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/planets"
        />
</androidx.constraintlayout.widget.ConstraintLayout>
```

MainActivity.java

```
package com.example.imagesandspinner;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
```

```

import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView;
import android.widget.AdapterView;
import android.widget.AdapterView;
import android.widget.AdapterView;

import java.util.HashMap;
public class MainActivity extends AppCompatActivity {

    /* defining Spinner and HashMap for drawables*/
    Spinner cardsSpinner;
    HashMap<String,String> drawableMap = new HashMap<String, String>();

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        //getting the ImageView
        ImageView img = (ImageView)findViewById(R.id.imageView);

        //getting the String Array resources from strings.xml
        String[] cards = getResources().getStringArray(R.array.cards);
        //populating map
        drawableMap.put(cards[0],"ace_of_club");
        drawableMap.put(cards[1],"ace_of_diamond");
        drawableMap.put(cards[2],"ace_of_heart");
        drawableMap.put(cards[3],"ace_of_spade");

        //getting spinner and adding Items to it
        cardsSpinner = (Spinner)findViewById(R.id.planets);
        ArrayAdapter<String> adapter = new
        ArrayAdapter<String>(this,android.R.layout.simple_spinner_item, cards);

        adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
        cardsSpinner.setAdapter(adapter);

        // adding Listener to the cardsSpinner Object for the callback method
        onItemSelected() using anonymous class
        cardsSpinner.setOnItemSelectedListener(new
        AdapterView.OnItemSelectedListener() {
            @Override
            public void onItemSelected(AdapterView<?> adapterView, View view, int i,
            long l) {
                String card =
                drawableMap.get(adapterView.getItemAtPosition(i).toString());
                int resId =
                getResources().getIdentifier(card,"drawable",getPackageName());
                img.setImageResource(resId);
            }

            @Override
            public void onNothingSelected(AdapterView<?> adapterView) {

            }
        }
    }
}

```

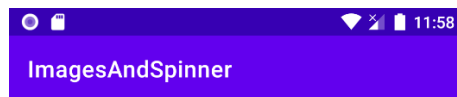


```
});  
  
}  
  
}
```

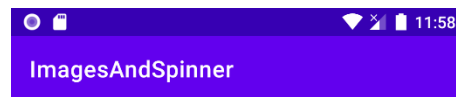
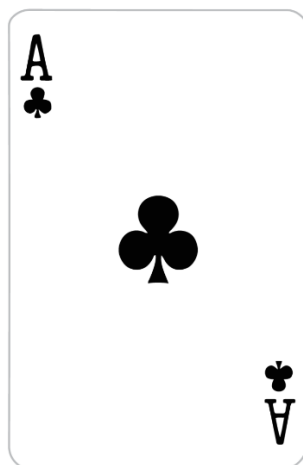
strings.xml

```
<resources>  
    <string name="app_name">ImagesAndSpinner</string>  
    <string-array name="cards">  
        <item>Ace Of Clubs</item>  
        <item>Ace Of Diamonds</item>  
        <item>Ace Of Hearts</item>  
        <item>Ace Of Spades</item>  
    </string-array>  
</resources>
```

Output

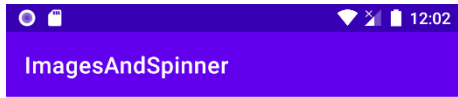


Ace Of Clubs ▼

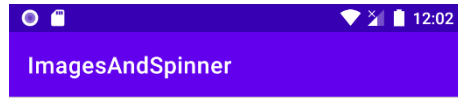
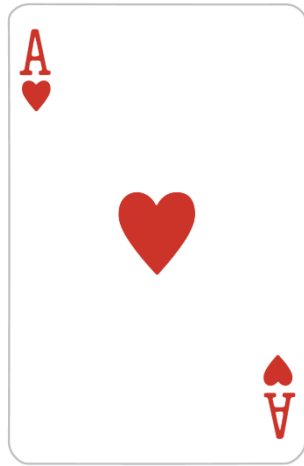


Ace Of Spades ▼

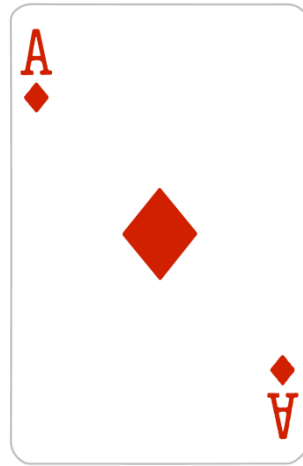




Ace Of Hearts ▼



Ace Of Diamonds ▼



Practical 6

Create a Menu with 5 Options and on selecting any Option the Option Title/Text should appear in a TextView in Main Activity in Uppercase.

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/selectedOption"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/selected_option_text"
        android:textAlignment="center"
        android:textSize="36sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

strings.xml

```
<resources>
    <string name="app_name">Menu</string>
    <string name="menu_option_1">Option 1</string>
    <string name="menu_option_2">Option 2</string>
    <string name="menu_option_3">Option 3</string>
    <string name="menu_option_4">Option 4</string>
    <string name="menu_option_5">Option 5</string>
    <string name="selected_option_text">Select An Option from Menu</string>
</resources>
```

options_menu.xml

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:app="http://schemas.android.com/apk/res-auto"
      xmlns:android="http://schemas.android.com/apk/res/android">
    <item
        android:id="@+id/menu_option_1"
        android:title="@string/menu_option_1"
        app:showAsAction="never" />
    <item
        android:id="@+id/menu_option_2"
        android:title="@string/menu_option_2"
        app:showAsAction="never" />
    <item
        android:id="@+id/menu_option_3"
        android:title="@string/menu_option_3"
        app:showAsAction="never" />
    <item
        android:id="@+id/menu_option_4"
        android:title="@string/menu_option_4"
        app:showAsAction="never" />
    <item
        android:id="@+id/menu_option_5"
        android:title="@string/menu_option_5"
        app:showAsAction="never" />
</menu>
```

MainActivity.java

```
package com.example.menu;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.util.Log;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {

    TextView selectedOption;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }

    // This method inflates the Menu defined as XML in the res/menu Resource to an
    Menu Object that is added to this activity.
```

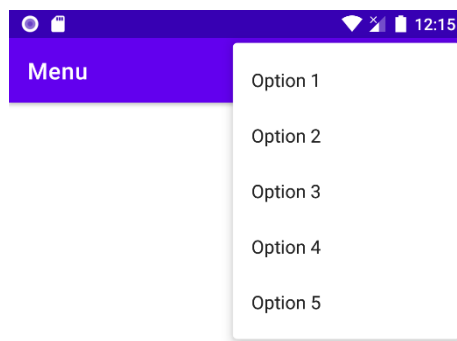
```

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    MenuInflater inflater = getMenuInflater();
    inflater.inflate(R.menu.options_menu, menu);
    return true;
}

//This method listens for the Change in the Menu Object i.e is called whenever a
Selection is made in the Menu Object.
@Override
public boolean onOptionsItemSelected(MenuItem item) {
    // Handle item selection
    String selectedOptionText = item.getTitle().toString().toUpperCase();
    selectedOption = (TextView)findViewById(R.id.selectedOption);
    selectedOption.setText(selectedOptionText);
    return super.onOptionsItemSelected(item);
}
}

```

Output



Select An Option from
Menu



OPTION 3



