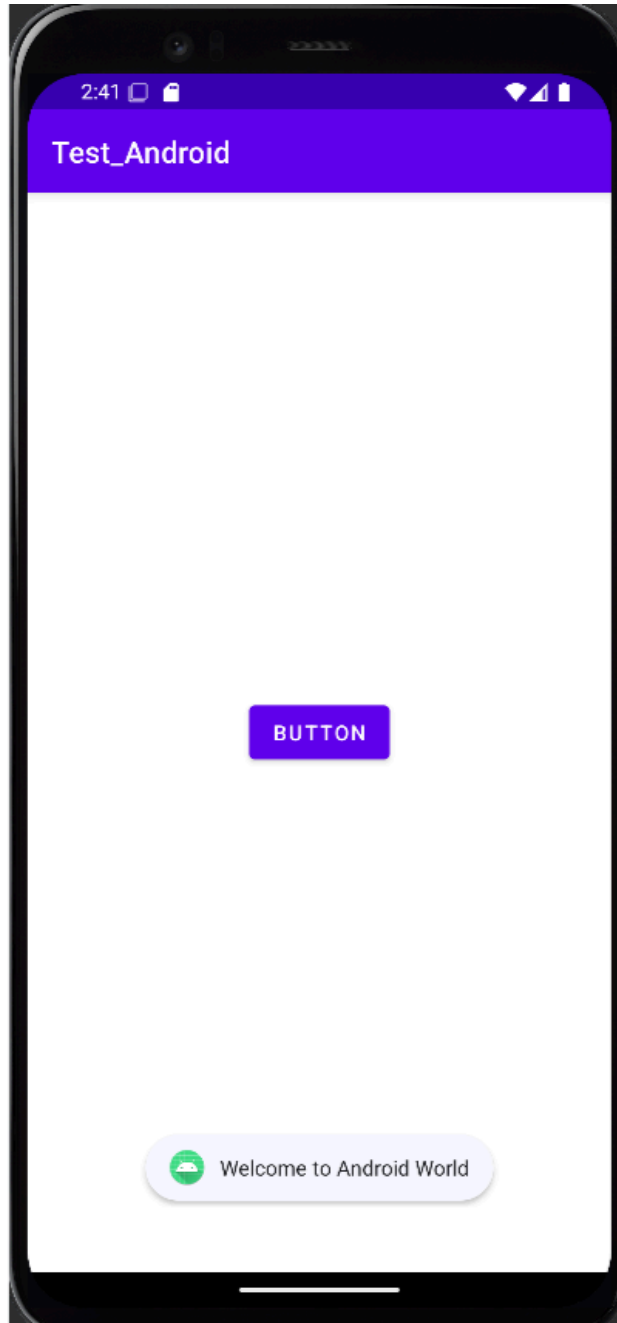


Practical Assignments – 2

1. Write a Kotlin program in Android to display a welcome message to the user on a click of a Button using Toast.



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:text="Button"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

</androidx.constraintlayout.widget.ConstraintLayout>

```

Mainactivity.kt:

```

package com.example.test_android

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Button
import android.widget.Toast

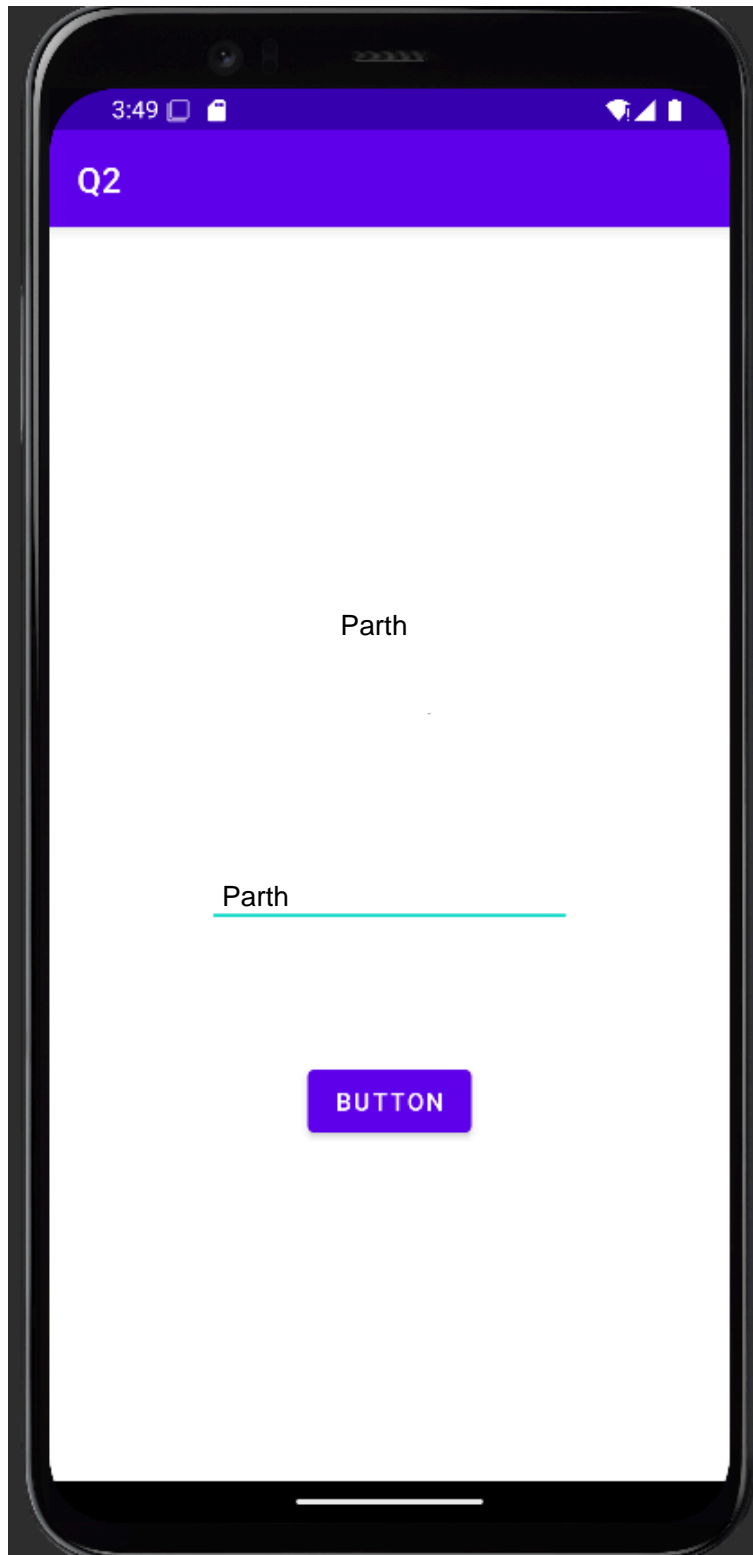
class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        var btnToast: Button = findViewById(R.id.button)

        btnToast.setOnClickListener(){
            Toast.makeText(this, "Welcome to Android World", Toast.LENGTH_LONG).show()
        }
    }
}

```

2. Write a Kotlin program in Android to accept a text from the user using Edit Text control and display it on the screen on the click of a 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=".SecondQuestion">

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="TextView"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.498"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.377" />

    <Button
        android:id="@+id/button2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Button"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textView" />

    <EditText
        android:id="@+id/editTextTextPersonName"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="textPersonName"
        android:hint="Your Name"
        app:layout_constraintBottom_toTopOf="@+id/button2"
```

```
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/textView" />
```

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

Mainactivity.kt:

```
package com.example.test_android

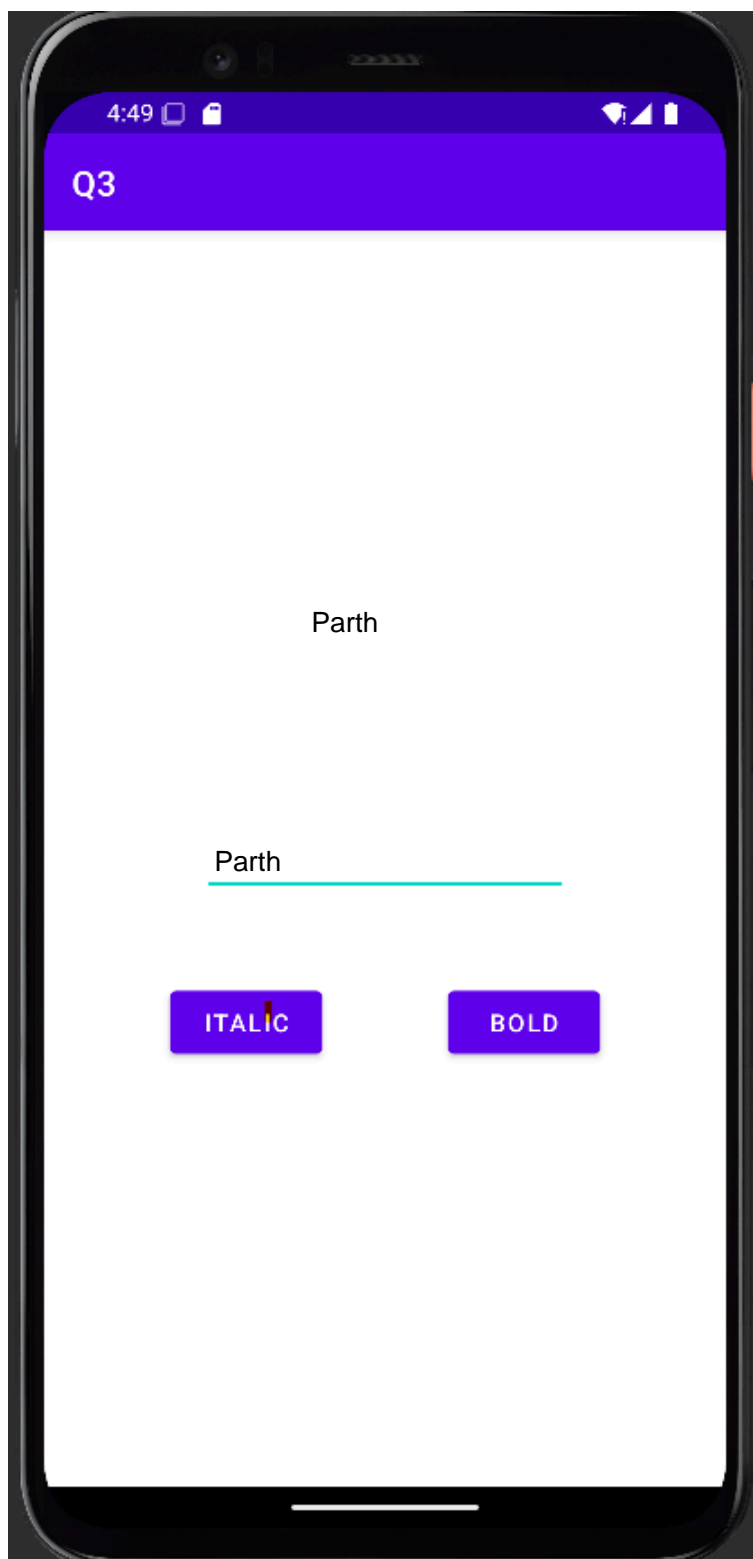
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Button
import android.widget.EditText
import android.widget.TextView

class SecondQuestion : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_second_question)

        var actionBtn:Button = findViewById(R.id.button2)
        var text:TextView = findViewById(R.id.textView)
        var displayText:EditText = findViewById(R.id.editTextTextPersonName)

        actionBtn.setOnClickListener {
            val inputText = displayText.text.toString()
            text.text = inputText
        }
    }
}
```

3. Write a Kotlin program in Android to accept a text from the user using Edit Text control and provide 2 buttons to the user “Bold” and “Italic”. Implement the Bold or Italics formatting using toggle functionality for these 2 buttons. Display the user text in a Text View Control after formatting it.



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=".ThirdQuestion"
    tools:layout_editor_absoluteX="0dp"
    tools:layout_editor_absoluteY="-71dp">

    <EditText
        android:id="@+id/input"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="Your Name"
        android:inputType="textPersonName"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <Button
        android:id="@+id/boldBtn"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="48dp"
        android:text="Bold"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toEndOf="@+id/ItalicBtn"
        app:layout_constraintTop_toBottomOf="@+id/input" />

    <Button
        android:id="@+id/ItalicBtn"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="48dp"
        android:text="Italic"
        app:layout_constraintEnd_toStartOf="@+id/boldBtn"
        app:layout_constraintHorizontal_bias="0.5"
```

```
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/input" />
```

```
<TextView
    android:id="@+id/msgDisplay"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:padding="10px"
    android:text="TextView"
    app:layout_constraintBottom_toTopOf="@+id/input"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

Mainactivity.kt:

```
package com.example.test_android

import android.graphics.Typeface
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Button
import android.widget.EditText
import android.widget.TextView

class ThirdQuestion : AppCompatActivity() {
    private var isBold = false
    private var isItalic = false

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_third_question)

        val boldBtn: Button = findViewById(R.id.boldBtn)
        val italicBtn: Button = findViewById(R.id.ItalicBtn)
        val msgDisplay: TextView = findViewById(R.id.msgDisplay)
        val input: EditText = findViewById(R.id.input)

        fun updateStyle() {
            msgDisplay.text = input.text
            msgDisplay.setTypeface(null,
```



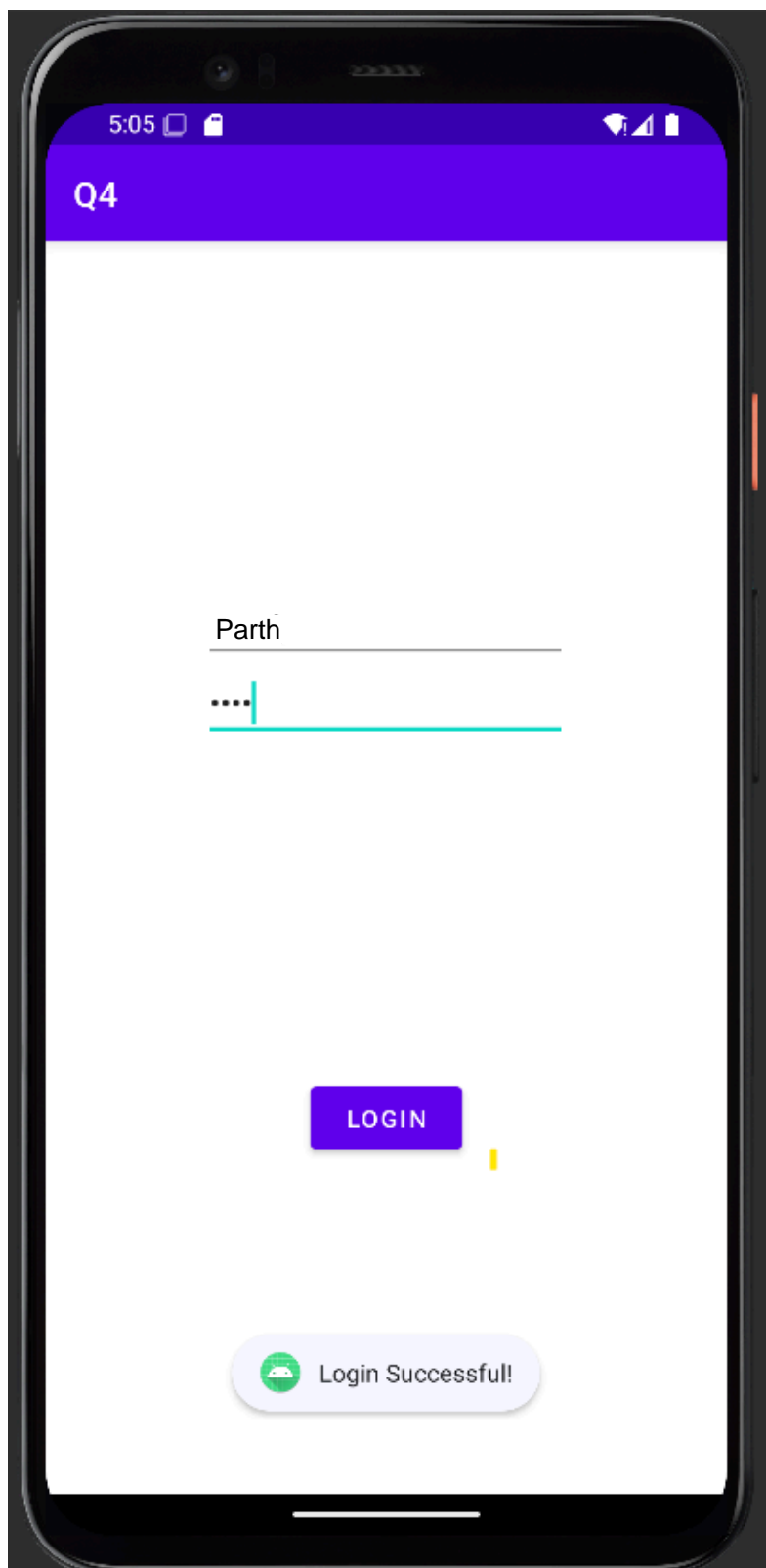
```
        if (isBold && isItalic) Typeface.BOLD_ITALIC
        else if (isBold) Typeface.BOLD
        else if (isItalic) Typeface.ITALIC
        else Typeface.NORMAL)
    }

    boldBtn.setOnClickListener { isBold = !isBold; updateStyle() }
    italicBtn.setOnClickListener { isItalic = !isItalic; updateStyle() }

}

}
```

4. Write a Kotlin program in Android to create a LOGIN PAGE to demonstrate the use of Floating Label, Numeric keyboard and Password field.



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=".FourthQuestion">

<EditText
    android:id="@+id/Username"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="200dp"
    android:ems="10"
    android:inputType="textPersonName"
    android:hint="Username"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.497"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />

<Button
    android:id="@+id/Login"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Login"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/Password" />

<EditText
    android:id="@+id/Password"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:ems="10"
    android:hint="Password"
    android:inputType="textPassword"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.497"
    app:layout_constraintStart_toStartOf="parent"
```

```
        app:layout_constraintTop_toBottomOf="@+id/Username" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

Mainactivity.kt:

```
package com.example.test_android

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Button
import android.widget.EditText
import android.widget.Toast

class FourthQuestion : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_fourth_question)

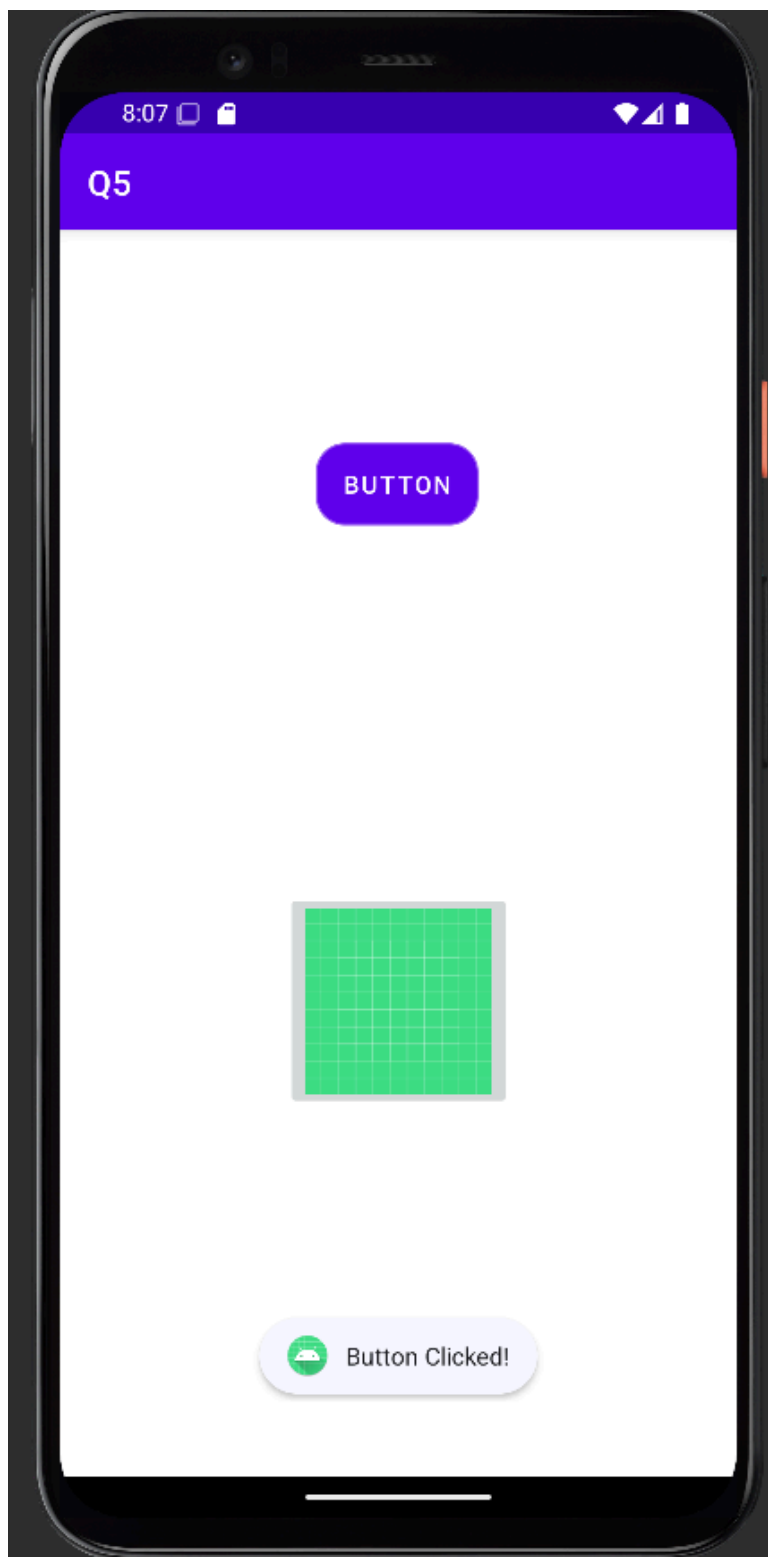
        var user: EditText = findViewById(R.id.Username)
        var pass: EditText = findViewById(R.id.Password)
        var loginBtn: Button = findViewById(R.id.Login)

        loginBtn.setOnClickListener{

            val username = user.toString()
            val password = pass.toString()

            if(username.isEmpty() || password.isEmpty()){
                Toast.makeText(this, "Invalid Input !!", Toast.LENGTH_LONG).show()
            }
            else{
                Toast.makeText(this, "Login Successful!", Toast.LENGTH_SHORT).show()
            }
        }
    }
}
```

5. Write a Kotlin program in Android to demonstrate the use of background property of Button Control as well as Image 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=".FifthQuestion">

    <Button
        android:id="@+id/button3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:background="@drawable/custom_btn"
        android:text="Button"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.498"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.183" />

    <ImageButton
        android:id="@+id/imageButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/button3"
        app:srcCompat="@drawable/icon_svg_background" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

Mainactivity.kt:

```
package com.example.test_android

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
```

```
import android.widget.Button
import android.widget.ImageButton
import android.widget.Toast

class FifthQuestion : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_fifth_question)

        var btn:Button = findViewById(R.id.button3)
        var imgBtn: ImageButton = findViewById(R.id.imageButton)

        btn.setOnClickListener {
            Toast.makeText(this, "Button Clicked!", Toast.LENGTH_SHORT).show()
        }

        imgBtn.setOnClickListener {
            Toast.makeText(this, "ImageButton Clicked!", Toast.LENGTH_SHORT).show()
        }
    }
}
```

6. Write a Kotlin program in Android to design a Student Registration Form to implement different widgets available in Android. Provide appropriate fields and 2 buttons “Submit” and “Reset”. Implement the appropriate functionality of the Button.

8:28

Q6

Parth

parth@gmail.com

341241512

☒ Male

☐ Female

☒ I accept the terms

SUBMIT

RESET

Registration Successful

Activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="20dp">

    <EditText
        android:id="@+id/name"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Full Name" />

    <EditText
        android:id="@+id/email"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Email"
        android:inputType="textEmailAddress"/>

    <EditText
        android:id="@+id/phone"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Phone Number"
        android:inputType="phone"/>

    <RadioGroup
        android:id="@+id/gender"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <RadioButton
            android:id="@+id/male"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Male"/>
        <RadioButton
            android:id="@+id/female"
```

```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Female"/>
</RadioGroup>

<CheckBox
    android:id="@+id/terms"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="I accept the terms"/>

<Button
    android:id="@+id/submit"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Submit"/>

<Button
    android:id="@+id/reset"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Reset"/>

</LinearLayout>

```

Mainactivity.kt:

```

package com.example.test_android

import android.os.Bundle
import android.widget.*
import androidx.appcompat.app.AppCompatActivity

class SixthQuestion : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_sixth_question)

        val nameInput: EditText = findViewById(R.id.name)
        val emailInput: EditText = findViewById(R.id.email)
        val phoneInput: EditText = findViewById(R.id.phone)
    }
}

```

```

val genderGroup: RadioGroup = findViewById(R.id.gender)
val termsCheckBox: CheckBox = findViewById(R.id.terms)
val submitButton: Button = findViewById(R.id.submit)
val resetButton: Button = findViewById(R.id.reset)

submitButton.setOnClickListener {
    val name = nameInput.text.toString()
    val email = emailInput.text.toString()
    val phone = phoneInput.text.toString()
    val genderId = genderGroup.checkedRadioButtonId
    val gender = if (genderId != -1) findViewById<RadioButton>(genderId).text.toString() else ""
    val termsAccepted = termsCheckBox.isChecked

    if (name.isEmpty() || email.isEmpty() || phone.isEmpty() || gender.isEmpty() || !termsAccepted) {
        Toast.makeText(this, "Fill all fields & accept terms", Toast.LENGTH_SHORT).show()
    } else {
        Toast.makeText(this, "Registration Successful", Toast.LENGTH_SHORT).show()
    }
}

resetButton.setOnClickListener {
    nameInput.text.clear()
    emailInput.text.clear()
    phoneInput.text.clear()
    genderGroup.clearCheck()
    termsCheckBox.isChecked = false
}
}
}

```

7. Use the Registration Form created in Q6 and implement the click event for the submit button to check whether all the fields are filled or not. If all the form fields are filled, redirect the user to a welcome page, displaying welcome message. On the click of the clear button, reset all the fields.

8:28

Q6

Parth

parth@gmail.com

341241512

☒ Male

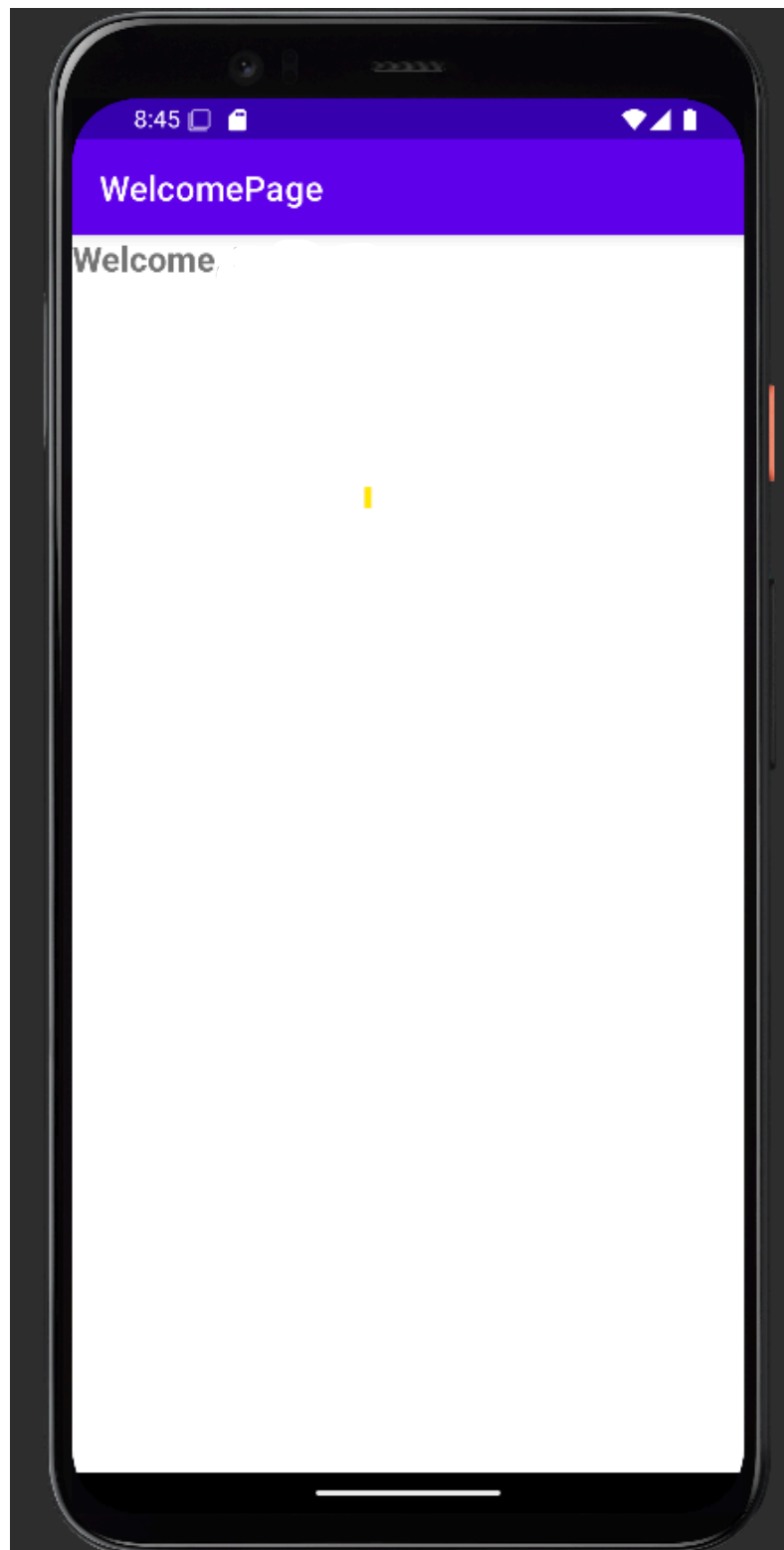
☐ Female

☒ I accept the terms

SUBMIT

RESET

Registration Successful



Activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="20dp">

    <EditText
        android:id="@+id/name"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Full Name" />

    <EditText
        android:id="@+id/email"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Email"
        android:inputType="textEmailAddress"/>

    <EditText
        android:id="@+id/phone"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Phone Number"
        android:inputType="phone"/>

    <RadioGroup
        android:id="@+id/gender"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <RadioButton
            android:id="@+id/male"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Male"/>
        <RadioButton
            android:id="@+id/female"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Female"/>
    </RadioGroup>
```

```
<CheckBox
    android:id="@+id/terms"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="I accept the terms"/>
```

```
<Button
    android:id="@+id/submit"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Submit"/>
```

```
<Button
    android:id="@+id/reset"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Reset"/>
```

```
</LinearLayout>
```

Mainactivity.kt:

```
package com.example.test_android
```

```
import android.content.Intent
```

```
import android.os.Bundle
```

```
import android.widget.*
```

```
import androidx.appcompat.app.AppCompatActivity
```

```
class SixthQuestion : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_sixth_question)

        val nameInput: EditText = findViewById(R.id.name)
        val emailInput: EditText = findViewById(R.id.email)
        val phoneInput: EditText = findViewById(R.id.phone)
        val genderGroup: RadioGroup = findViewById(R.id.gender)
        val termsCheckBox: CheckBox = findViewById(R.id.terms)
        val submitButton: Button = findViewById(R.id.submit)
```

```

val resetButton: Button = findViewById(R.id.reset)

submitButton.setOnClickListener {
    val name = nameInput.text.toString()
    val email = emailInput.text.toString()
    val phone = phoneInput.text.toString()
    val genderId = genderGroup.checkedRadioButtonId
    val gender = if (genderId != -1) findViewById<RadioButton>(genderId).text.toString() else ""
    val termsAccepted = termsCheckBox.isChecked

    if (name.isEmpty() || email.isEmpty() || phone.isEmpty() || gender.isEmpty() || !termsAccepted) {
        Toast.makeText(this, "Fill all fields & accept terms", Toast.LENGTH_SHORT).show()
    } else {
        val intent = Intent(this, WelcomeActivity::class.java)
        intent.putExtra("name", name)
        startActivity(intent)
    }
}

resetButton.setOnClickListener {
    nameInput.text.clear()
    emailInput.text.clear()
    phoneInput.text.clear()
    genderGroup.clearCheck()
    termsCheckBox.isChecked = false
}
}
}

```

Activity_welcome.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=".WelcomeActivity">

    <TextView
        android:id="@+id/welcomeMessage"
        android:layout_width="wrap_content"

```



```
android:layout_height="wrap_content"
android:textSize="20sp"
android:textStyle="bold"/>
```

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

Welcomeactivity.kt:

```
package com.example.test_android
```

```
import android.os.Bundle
```

```
import android.widget.TextView
```

```
import androidx.appcompat.app.AppCompatActivity
```

```
class WelcomeActivity : AppCompatActivity() {
```

```
    override fun onCreate(savedInstanceState: Bundle?) {
```

```
        super.onCreate(savedInstanceState)
```

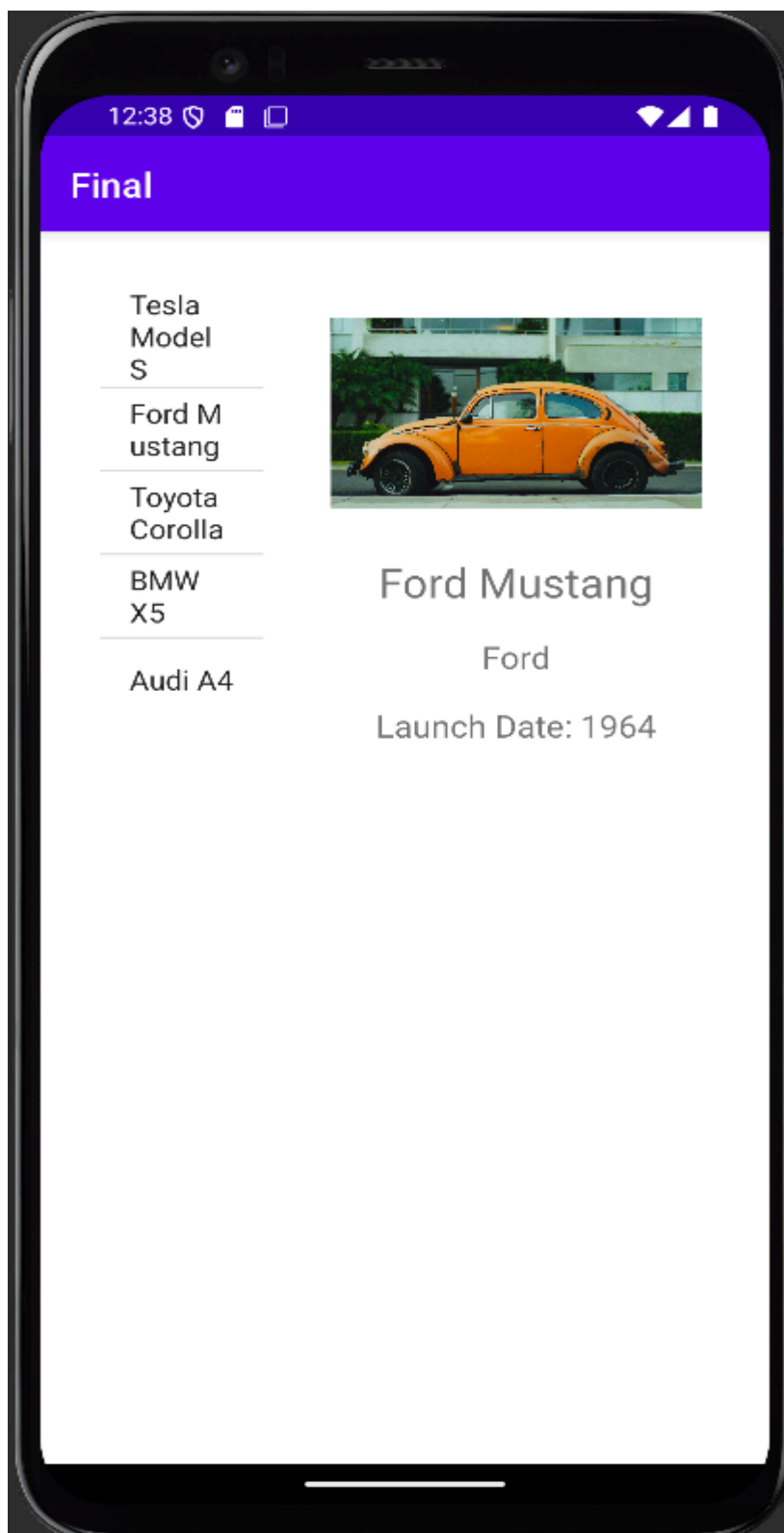
```
        setContentView(R.layout.activity_welcome)
```

```
        val welcomeMessage = findViewById<TextView>(R.id.welcomeMessage)
```

```
        val name = intent.getStringExtra("name")
```

```
        welcomeMessage.text = "Welcome, $name!"}}
```

8. Create an android application in Kotlin, to implement the Fragment. Divide the main activity UI into 2 parts. The 1st part should display a list of cars of your choice. On the click of any 1 item, display the car name, and car details like: name, launch date, company name, image related to it in the 2nd section. Provide a list of minimum 5 cars



MainActivity.kt

```
package com.example.afinal

import android.os.Bundle
import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity() {

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        supportFragmentManager.beginTransaction()
            .replace(R.id.fragment_container_list, CarListFragment())
            .commit()
    }
}
```

CarDetailFragment.kt

```
package com.example.afinal

import android.os.Bundle
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.ImageView
import android.widget.TextView
import androidx.fragment.app.Fragment

class CarDetailsFragment : Fragment() {

    private var carName: String? = null

    override fun onCreateView(
        inflater: LayoutInflater, container: ViewGroup?,
        savedInstanceState: Bundle?
    ): View? {
        val view = inflater.inflate(R.layout.fragment_car_details, container, false)

        carName = arguments?.getString("carName")

        // Update UI with car details
        val carImage: ImageView = view.findViewById(R.id.carImage)
```

```
val carNameTextView: TextView = view.findViewById(R.id.carName)
val carCompanyTextView: TextView = view.findViewById(R.id.carCompany)
val carLaunchDateTextView: TextView = view.findViewById(R.id.carLaunchDate)
```

```
when (carName) {
    "Tesla Model S" -> {
        carImage.setImageResource(R.drawable.car1)
        carNameTextView.text = "Tesla Model S"
        carCompanyTextView.text = "Tesla"
        carLaunchDateTextView.text = "Launch Date: 2012"
    }
    "Ford Mustang" -> {
        carImage.setImageResource(R.drawable.car2)
        carNameTextView.text = "Ford Mustang"
        carCompanyTextView.text = "Ford"
        carLaunchDateTextView.text = "Launch Date: 1964"
    }
    "Toyota Corolla" -> {
        carImage.setImageResource(R.drawable.car3)
        carNameTextView.text = "Toyota Corolla"
        carCompanyTextView.text = "Toyota"
        carLaunchDateTextView.text = "Launch Date: 1966"
    }
    "BMW X5" -> {
        carImage.setImageResource(R.drawable.car4)
        carNameTextView.text = "BMW X5"
        carCompanyTextView.text = "BMW"
        carLaunchDateTextView.text = "Launch Date: 1999"
    }
    "Audi A4" -> {
        carImage.setImageResource(R.drawable.car5)
        carNameTextView.text = "Audi A4"
        carCompanyTextView.text = "Audi"
        carLaunchDateTextView.text = "Launch Date: 1994"
    }
}
```

```
return view
```

```
}
```

```
companion object {
    fun newInstance(carName: String): CarDetailsFragment {
        val fragment = CarDetailsFragment()
        val args = Bundle()
        args.putString("carName", carName)
        fragment.arguments = args
        return fragment
    }
}
```

```
}  
}
```

CarListFragment.kt

```
package com.example.afinal
```

```
import android.os.Bundle  
import androidx.fragment.app.Fragment  
import android.view.LayoutInflater  
import android.view.View  
import android.view.ViewGroup  
import android.widget.AdapterView  
import android.widget.ArrayAdapter  
import android.widget.ListView
```

```
class CarListFragment : Fragment() {
```

```
    private lateinit var carListView: ListView  
    private val carList = arrayOf("Tesla Model S", "Ford Mustang", "Toyota Corolla", "BMW X5", "Audi A4")
```

```
    override fun onCreateView(  
        inflater: LayoutInflater, container: ViewGroup?,  
        savedInstanceState: Bundle?  
    ): View? {  
        val view = inflater.inflate(R.layout.fragment_car_list, container, false)  
        carListView = view.findViewById(R.id.carListView)
```

```
        val adapter = ArrayAdapter(requireContext(), android.R.layout.simple_list_item_1, carList)  
        carListView.adapter = adapter
```

```
        carListView.setOnItemClickListener = AdapterView.OnItemClickListener { _, _, position, _ ->  
            val carName = carList[position]  
            val detailsFragment = CarDetailsFragment.newInstance(carName)  
            parentFragmentManager.beginTransaction()  
                .replace(R.id.fragment_container_details, detailsFragment)  
                .commit()  
        }
```

```
        return view
```

```
    }  
}
```

ActivityMain.xml

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="horizontal"
    android:padding="16dp">

    <FrameLayout
        android:id="@+id/fragment_container_list"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1" />

    <FrameLayout
        android:id="@+id/fragment_container_details"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="2" />

</LinearLayout>
```

FragmentCarDetail.xml

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp">

    <ImageView
        android:id="@+id/carImage"
        android:layout_width="200dp"
        android:layout_height="150dp"
        android:layout_gravity="center_horizontal"
        android:contentDescription="Car Image" />

    <TextView
        android:id="@+id/carName"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Car Name"
        android:textSize="24sp"
        android:layout_gravity="center_horizontal">
```

```
android:padding="8dp" />
```

```
<TextView
```

```
    android:id="@+id/carCompany"
```

```
    android:layout_width="wrap_content"
```

```
    android:layout_height="wrap_content"
```

```
    android:text="Company Name"
```

```
    android:textSize="18sp"
```

```
    android:layout_gravity="center_horizontal"
```

```
    android:padding="8dp" />
```

```
<TextView
```

```
    android:id="@+id/carLaunchDate"
```

```
    android:layout_width="wrap_content"
```

```
    android:layout_height="wrap_content"
```

```
    android:text="Launch Date"
```

```
    android:textSize="18sp"
```

```
    android:layout_gravity="center_horizontal"
```

```
    android:padding="8dp" />
```

```
</LinearLayout>
```

FragmentCarlist.xml

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
```

```
    android:layout_width="match_parent"
```

```
    android:layout_height="match_parent"
```

```
    android:orientation="vertical"
```

```
    android:padding="16dp">
```

```
<ListView
```

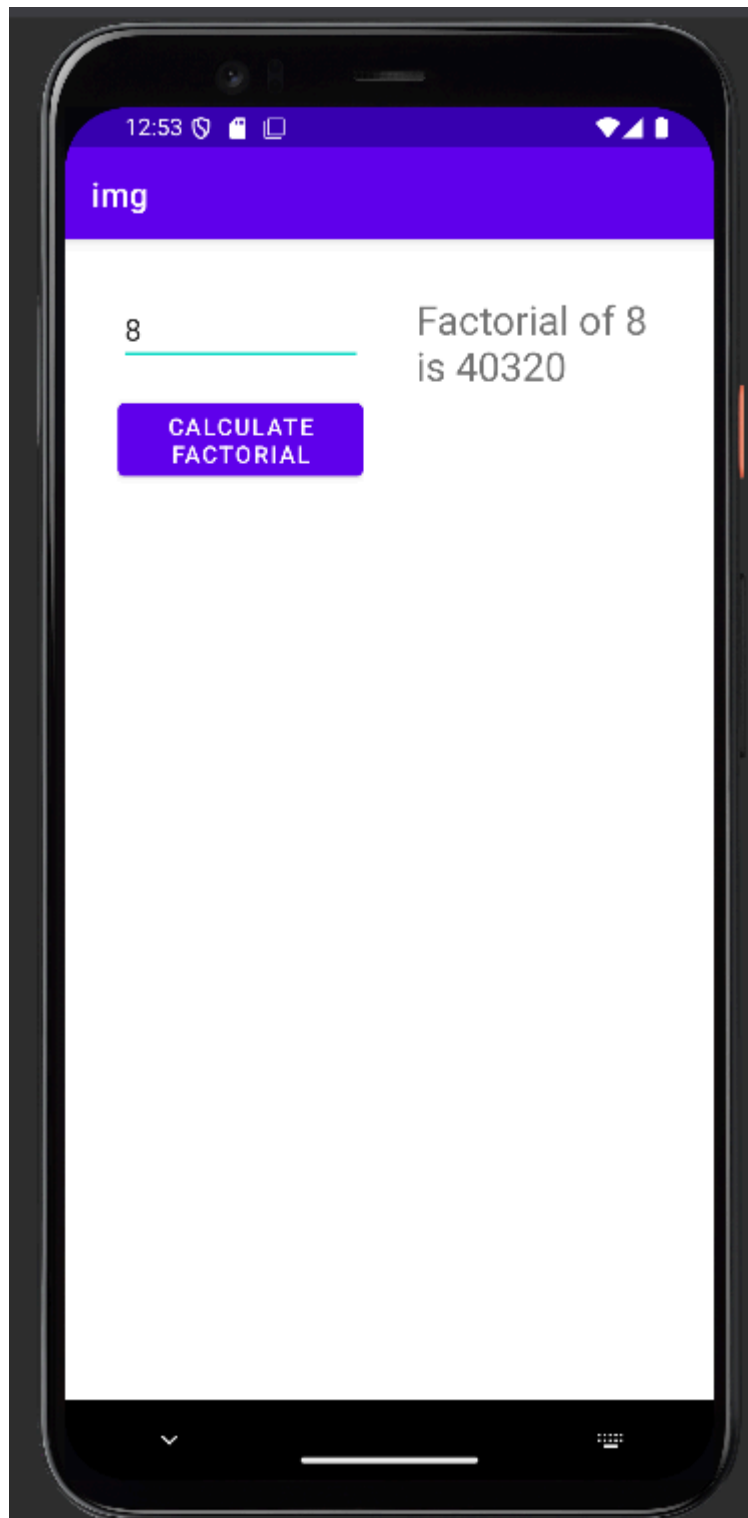
```
    android:id="@+id/carListView"
```

```
    android:layout_width="match_parent"
```

```
    android:layout_height="wrap_content" />
```

```
</LinearLayout>
```

9. Write a Kotlin program in Android, to implement Fragment Communication. Create an android application, which divides the main activity UI into 2 parts. The 1st part has an Edit Text & Button, which accepts a numeric value from the user and on the click of the button display the factorial of the number in the 2nd fragment.



MainActivity.kt

```
package com.example.afinal

import android.os.Bundle
import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity() {

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        supportFragmentManager.beginTransaction()
            .replace(R.id.fragment_container_input, InputFragment())
            .commit()

        supportFragmentManager.beginTransaction()
            .replace(R.id.fragment_container_result, ResultFragment())
            .commit()
    }
}
```

InputFragment.kt

```
package com.example.afinal

import android.os.Bundle
import androidx.fragment.app.Fragment
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.Button
import android.widget.EditText

class InputFragment : Fragment() {

    private lateinit var inputNumber: EditText
    private lateinit var calculateButton: Button

    override fun onCreateView(
        inflater: LayoutInflater, container: ViewGroup?,
        savedInstanceState: Bundle?
    ): View? {
```

```

val view = inflater.inflate(R.layout.fragment_input, container, false)

inputNumber = view.findViewById(R.id.inputNumber)
calculateButton = view.findViewById(R.id.calculateButton)

calculateButton.setOnClickListener {
    val number = inputNumber.text.toString().toIntOrNull()
    if (number != null) {
        val factorial = calculateFactorial(number)
        // Communicate with ResultFragment
        val resultFragment =
parentFragmentManager.findFragmentById(R.id.fragment_container_result) as ResultFragment
        resultFragment.updateResult("Factorial of $number is $factorial")
    } else {
        inputNumber.error = "Please enter a valid number"
    }
}

return view
}

private fun calculateFactorial(n: Int): Long {
    return if (n == 0 || n == 1) 1 else n * calculateFactorial(n - 1)
}
}

```

ResultFragment.kt

```

package com.example.afinal

import android.os.Bundle
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.TextView
import androidx.fragment.app.Fragment

class ResultFragment : Fragment() {

    private lateinit var resultText: TextView

    override fun onCreateView(
        inflater: LayoutInflater, container: ViewGroup?,
        savedInstanceState: Bundle?
    ) {

```

```

): View? {
    val view = inflater.inflate(R.layout.fragment_result, container, false)
    resultText = view.findViewById(R.id.resultText)
    return view
}

fun updateResult(result: String) {
    resultText.text = result
}
}

```

ActivityMain.xml

```

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="horizontal"
    android:padding="16dp">
    <FrameLayout
        android:id="@+id/fragment_container_input"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1" />
    <FrameLayout
        android:id="@+id/fragment_container_result"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1" />

</LinearLayout>

```

FragmentInput.xml

```

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp">

    <EditText
        android:id="@+id/inputNumber"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter a number"
        android:inputType="number" />

```

```
<Button
    android:id="@+id/calculateButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Calculate Factorial"
    android:layout_gravity="center_horizontal"
    android:layout_marginTop="16dp" />
```

```
</LinearLayout>
```

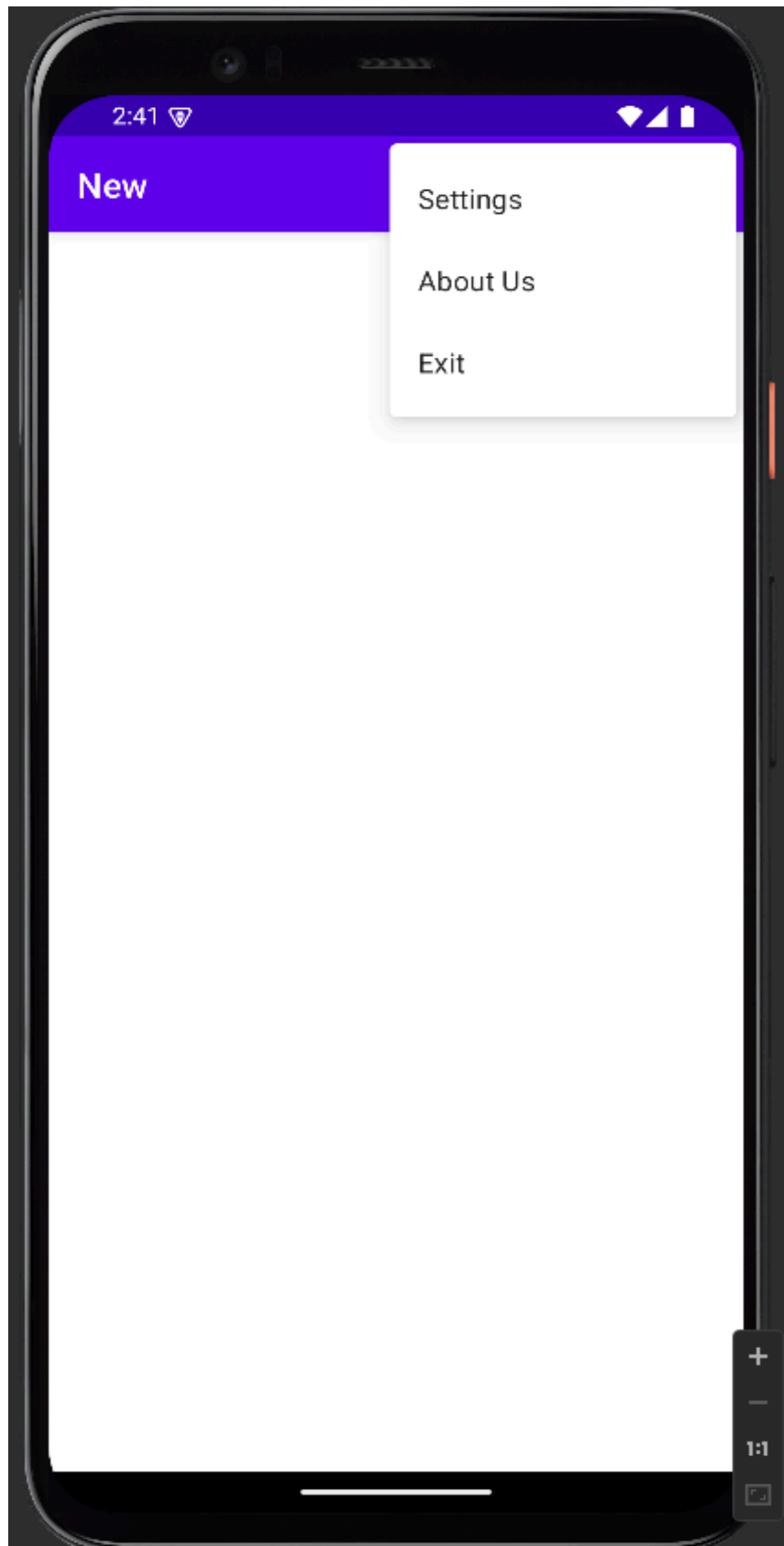
FragmentResult.xml

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp">
```

```
<TextView
    android:id="@+id/resultText"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Factorial Result"
    android:textSize="24sp"
    android:layout_gravity="center_horizontal" />
```

```
</LinearLayout>
```

10. Write a Kotlin program in Android, to implement Fragment Communication. Create an android application, which divides the main activity UI into 2 parts. The 1st part has an EditText & Button, which accepts a numeric value from the user and on the click of the button display the factorial of the number in the 2nd fragment.



mainActivity.kt

```
package com.example.anew

import android.os.Bundle
import android.view.Menu
import android.view.MenuItem
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity

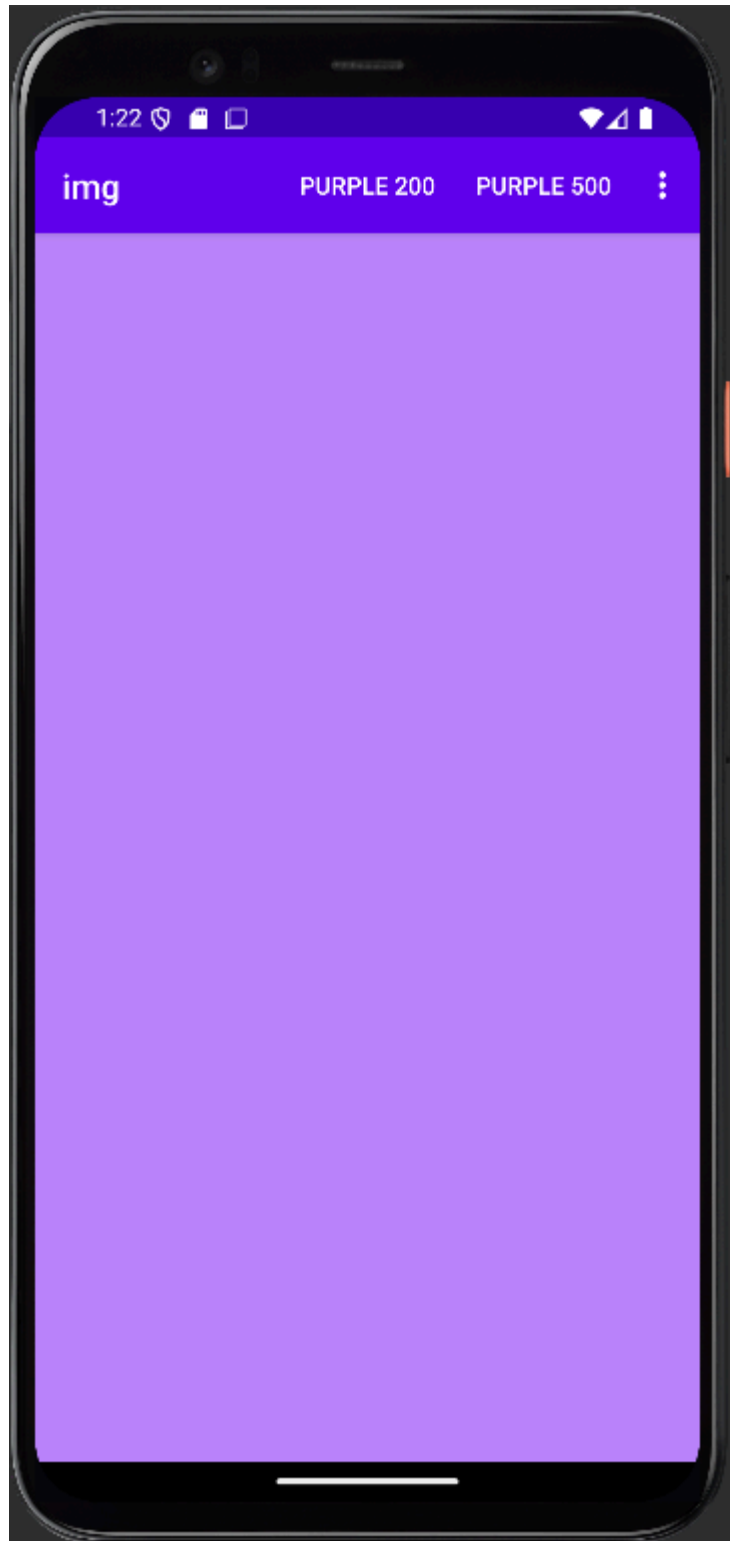
class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
    }
    override fun onCreateOptionsMenu(menu: Menu): Boolean {
        menuInflater.inflate(R.menu.options_menu, menu)
        return true
    }
    override fun onOptionsItemSelected(item: MenuItem): Boolean {
        return when (item.itemId) {
            R.id.settings -> {
                Toast.makeText(this, "Settings Selected", Toast.LENGTH_SHORT).show()
                true
            }
            R.id.about_us -> {
                Toast.makeText(this, "About Us Selected", Toast.LENGTH_SHORT).show()
                true
            }
            R.id.exit -> {
                Toast.makeText(this, "Exit Selected", Toast.LENGTH_SHORT).show()
                finish()
                true
            }
            else -> super.onOptionsItemSelected(item)
        }
    }
}
```

Optionmenu.xml

```
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item
        android:id="@+id/settings"
        android:icon="@drawable/ic_settings"
        android:title="Settings" />
```

```
<item
  android:id="@+id/about_us"
  android:icon="@drawable/ic_about"
  android:title="About Us" />
<item
  android:id="@+id/exit"
  android:icon="@drawable/ic_exit"
  android:title="Exit" />
</menu>
```

11. Create an android application in Kotlin, to create an options menu, to provide options for changing the background color of the layout. Also provide shortcuts for the menu items. Provide a minimum of 6 color options. Do not use basic colors, like red, blue, green etc...



MainActivity.kt

```
package com.example.afinal
import android.graphics.Color
import android.os.Bundle
import android.view.Menu
import android.view.MenuItem
import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity() {

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
    }
    override fun onCreateOptionsMenu(menu: Menu): Boolean {
        menuInflater.inflate(R.menu.options_menu, menu)
        return true
    }

    override fun onOptionsItemSelected(item: MenuItem): Boolean {
        when (item.itemId) {
            R.id.purple200 -> {
                window.decorView.setBackgroundColor(Color.parseColor("#FFBB86FC"))
                return true
            }
            R.id.purple500 -> {
                window.decorView.setBackgroundColor(Color.parseColor("#FF6200EE"))
                return true
            }
            R.id.purple700 -> {
                window.decorView.setBackgroundColor(Color.parseColor("#FF3700B3"))
                return true
            }
            R.id.teal200 -> {
                window.decorView.setBackgroundColor(Color.parseColor("#FF03DAC5"))
                return true
            }
            R.id.teal700 -> {
                window.decorView.setBackgroundColor(Color.parseColor("#FF018786"))
                return true
            }
            R.id.orange200 -> {
                window.decorView.setBackgroundColor(Color.parseColor("#FFFA500"))
                return true
            }
        }
        else -> return super.onOptionsItemSelected(item)
    }
}
```

```
}  
}  
}
```

Optionmenu.xml

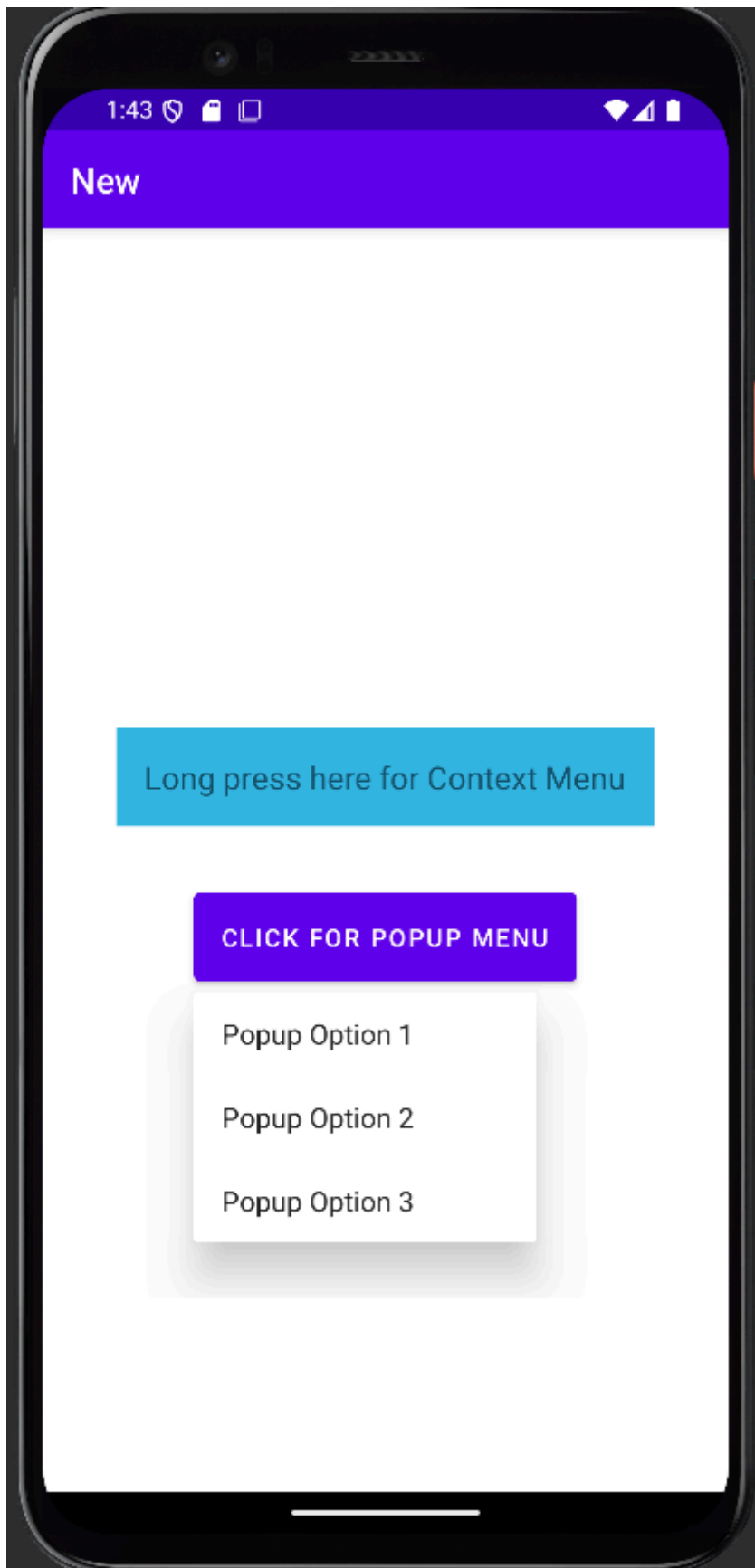
```
<menu xmlns:android="http://schemas.android.com/apk/res/android"  
      xmlns:app="http://schemas.android.com/apk/res-auto">  
  
  <item  
    android:id="@+id/purple200"  
    android:title="Purple 200"  
    app:showAsAction="ifRoom"  
    android:alphabeticShortcut="p" />  
  
  <item  
    android:id="@+id/purple500"  
    android:title="Purple 500"  
    app:showAsAction="ifRoom"  
    android:alphabeticShortcut="u" />  
  
  <item  
    android:id="@+id/purple700"  
    android:title="Purple 700"  
    app:showAsAction="ifRoom"  
    android:alphabeticShortcut="r" />  
  
  <item  
    android:id="@+id/teal200"  
    android:title="Teal 200"  
    app:showAsAction="ifRoom"  
    android:alphabeticShortcut="t" />  
  
  <item  
    android:id="@+id/teal700"  
    android:title="Teal 700"  
    app:showAsAction="ifRoom"  
    android:alphabeticShortcut="e" />  
  
  <item  
    android:id="@+id/orange200"  
    android:title="Orange 200"  
    app:showAsAction="ifRoom"  
    android:alphabeticShortcut="o" />  
  
</menu>
```

ActivityMain.xml

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="16dp">
```

```
</RelativeLayout>
```

12. Create an android application in Kotlin to demonstrate Context & Popup menu



MainActivity.kt

```
package com.example.anew

import android.os.Bundle
import android.view.ContextMenu
import android.view.MenuInflater
import android.view.MenuItem
import android.view.View
import android.widget.Button
import android.widget.PopupMenu
import android.widget.TextView
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity() {

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        val textView = findViewById<TextView>(R.id.textView)
        registerForContextMenu(textView)

        val popupButton = findViewById<Button>(R.id.popupButton)
        popupButton.setOnClickListener {
            showPopupMenu(it)
        }
    }

    override fun onCreateContextMenu(
        menu: ContextMenu?,
        v: View?,
        menuInfo: ContextMenu.ContextMenuInfo?
    ) {
        super.onCreateContextMenu(menu, v, menuInfo)
        menuInflater.inflate(R.menu.context_menu, menu)
    }

    override fun onContextItemSelected(item: MenuItem): Boolean {
        return when (item.itemId) {
            R.id.option1 -> {
                showToast("Context Option 1 selected")
                true
            }
            R.id.option2 -> {
                showToast("Context Option 2 selected")
            }
        }
    }
}
```

```

        true
    }
    R.id.option3 -> {
        showToast("Context Option 3 selected")
        true
    }
    else -> super.onContextItemSelected(item)
}
}

private fun showPopupMenu(view: View) {
    val popupMenu = PopupMenu(this, view)
    val inflater: MenuInflater = popupMenu.menuInflater
    inflater.inflate(R.menu.popup_menu, popupMenu.menu)

    popupMenu.setOnMenuItemClickListener { menuItem ->
        when (menuItem.itemId) {
            R.id.popupOption1 -> {
                showToast("Popup Option 1 selected")
                true
            }
            R.id.popupOption2 -> {
                showToast("Popup Option 2 selected")
                true
            }
            R.id.popupOption3 -> {
                showToast("Popup Option 3 selected")
                true
            }
            else -> false
        }
    }

    popupMenu.show()
}

private fun showToast(message: String) {
    Toast.makeText(this, message, Toast.LENGTH_SHORT).show()
}
}

```

Activitymain.xml

```

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp"

```

```
android:gravity="center">
<TextView
    android:id="@+id/textView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Long press here for Context Menu"
    android:textSize="18sp"
    android:padding="16dp"
    android:background="@android:color/holo_blue_light"
    android:gravity="center" />
<Button
    android:id="@+id/popupButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Click for Popup Menu"
    android:layout_marginTop="32dp"
    android:padding="16dp" />
</LinearLayout>
```

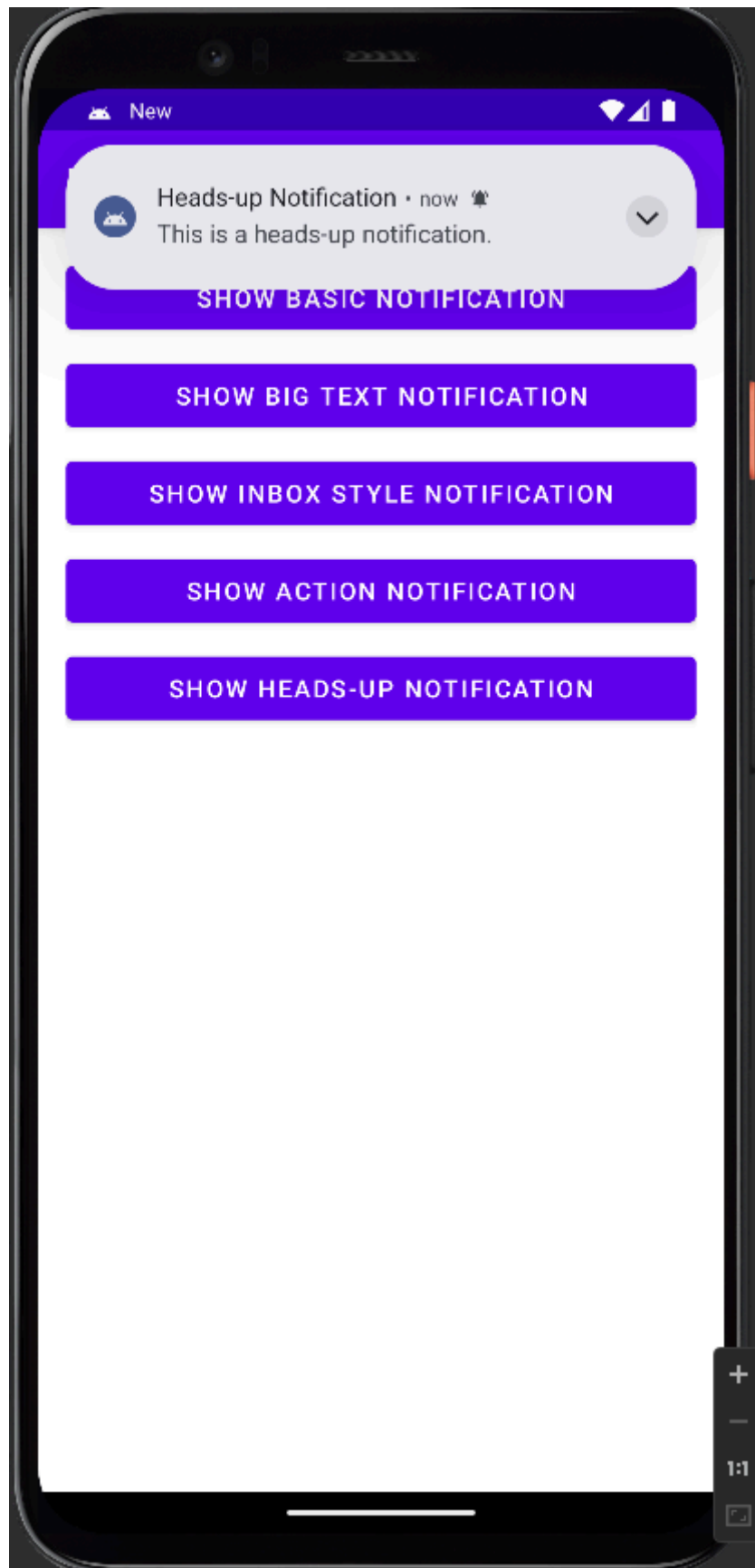
Contextmenu.xml

```
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item
        android:id="@+id/option1"
        android:title="Option 1" />
    <item
        android:id="@+id/option2"
        android:title="Option 2" />
    <item
        android:id="@+id/option3"
        android:title="Option 3" />
</menu>
```

Popupmenu.xml

```
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item
        android:id="@+id/popupOption1"
        android:title="Popup Option 1" />
    <item
        android:id="@+id/popupOption2"
        android:title="Popup Option 2" />
    <item
        android:id="@+id/popupOption3"
        android:title="Popup Option 3" />
</menu>
```

13. Create an android application in Kotlin to implement different types of Notifications.



Mainactivity.kt

```
package com.example.anew

import android.app.*
import android.content.Intent
import android.graphics.Color
import android.os.Build
import android.os.Bundle
import android.widget.Button
import androidx.annotation.RequiresApi
import androidx.appcompat.app.AppCompatActivity
import androidx.core.app.NotificationCompat
import androidx.core.app.NotificationManagerCompat

class MainActivity : AppCompatActivity() {

    private val CHANNEL_ID = "notification_channel"

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.O) {
            createNotificationChannel()
        }

        requestNotificationPermission()

        findViewById<Button>(R.id.basicNotification).setOnClickListener {
            showBasicNotification()
        }

        findViewById<Button>(R.id.bigTextNotification).setOnClickListener {
            showBigTextNotification()
        }

        findViewById<Button>(R.id.inboxStyleNotification).setOnClickListener {
            showInboxStyleNotification()
        }

        findViewById<Button>(R.id.actionNotification).setOnClickListener {
            showActionNotification()
        }

        findViewById<Button>(R.id.headsUpNotification).setOnClickListener {
            showHeadsUpNotification()
        }
    }
}
```

```

    }
}

@RequiresApi(Build.VERSION_CODES.O)
private fun createNotificationChannel() {
    val channel = NotificationChannel(
        CHANNEL_ID,
        "Notification Channel",
        NotificationManager.IMPORTANCE_HIGH
    ).apply {
        description = "This is a test notification channel"
        enableLights(true)
        lightColor = Color.RED
        enableVibration(true)
    }
    val manager = getSystemService(NotificationManager::class.java)
    manager.createNotificationChannel(channel)
}

private fun requestNotificationPermission() {
    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.TIRAMISU) {
        requestPermissions(arrayOf(android.Manifest.permission.POST_NOTIFICATIONS), 101)
    }
}

private fun showBasicNotification() {
    val builder = NotificationCompat.Builder(this, CHANNEL_ID)
        .setSmallIcon(R.drawable.ic_notification)
        .setContentTitle("Basic Notification")
        .setContentView("This is a basic notification.")
        .setPriority(NotificationCompat.PRIORITY_DEFAULT)
    NotificationManagerCompat.from(this).notify(1, builder.build())
}

private fun showBigTextNotification() {
    val builder = NotificationCompat.Builder(this, CHANNEL_ID)
        .setSmallIcon(R.drawable.ic_notification)
        .setContentTitle("Big Text Notification")
        .setStyle(NotificationCompat.BigTextStyle()
            .bigText("This is an example of a big text notification that contains more information."))
        .setPriority(NotificationCompat.PRIORITY_DEFAULT)
    NotificationManagerCompat.from(this).notify(2, builder.build())
}

private fun showInboxStyleNotification() {
    val builder = NotificationCompat.Builder(this, CHANNEL_ID)
        .setSmallIcon(R.drawable.ic_notification)
        .setContentTitle("Inbox Style Notification")

```

```

        .setStyle(NotificationCompat.InboxStyle()
            .addLine("Message 1")
            .addLine("Message 2")
            .addLine("Message 3")
            .setSummaryText("+3 more messages"))
        .setPriority(NotificationCompat.PRIORITY_DEFAULT)
        NotificationManagerCompat.from(this).notify(3, builder.build())
    }

    private fun showActionNotification() {
        val intent = Intent(this, MainActivity::class.java)
        val pendingIntent = PendingIntent.getActivity(this, 0, intent,
            PendingIntent.FLAG_UPDATE_CURRENT or PendingIntent.FLAG_IMMUTABLE)
        val builder = NotificationCompat.Builder(this, CHANNEL_ID)
            .setSmallIcon(R.drawable.ic_notification)
            .setContentTitle("Action Button Notification")
            .setContentText("Click to open the app.")
            .addAction(R.drawable.ic_launcher_foreground, "Open App", pendingIntent)
            .setPriority(NotificationCompat.PRIORITY_DEFAULT)
        NotificationManagerCompat.from(this).notify(4, builder.build())
    }

    private fun showHeadsUpNotification() {
        val builder = NotificationCompat.Builder(this, CHANNEL_ID)
            .setSmallIcon(R.drawable.ic_notification)
            .setContentTitle("Heads-up Notification")
            .setContentText("This is a heads-up notification.")
            .setPriority(NotificationCompat.PRIORITY_HIGH)
            .setDefaults(NotificationCompat.DEFAULT_ALL)
        NotificationManagerCompat.from(this).notify(5, builder.build())
    }
}

```

mainActivity.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp">

    <Button
        android:id="@+id/basicNotification"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"

```

```
android:text="Show Basic Notification" />
```

```
<Button
```

```
    android:id="@+id/bigTextNotification"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Show Big Text Notification"
    android:layout_marginTop="8dp"/>
```

```
<Button
```

```
    android:id="@+id/inboxStyleNotification"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Show Inbox Style Notification"
    android:layout_marginTop="8dp"/>
```

```
<Button
```

```
    android:id="@+id/actionNotification"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Show Action Notification"
    android:layout_marginTop="8dp"/>
```

```
<Button
```

```
    android:id="@+id/headsUpNotification"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Show Heads-Up Notification"
    android:layout_marginTop="8dp"/>
```

```
</LinearLayout>
```