SMARTBRIDGE EXTERNSHIP

ANDROID DEVELOPMENT WITH KOTLIN

ASSIGNMENT WEEK 1

GOOGLE ACCOUNT SIGN UP

https://github.com/Shrimad-Bhagwat/GoogleAccount_SignUp

```kotlin
package com.shrimadbhagwat.googleaccount

import android.content.Context
import android.os.Bundle
import android.widget.Toast
import androidx.activity.ComponentActivity
import androidx.activity.compose.setContent
import androidx.compose.foundation.Image
import androidx.compose.foundation.background
import androidx.compose.foundation.border
import androidx.compose.foundation.layout.Arrangement
import androidx.compose.foundation.layout.Column
import androidx.compose.foundation.layout.Row
import androidx.compose.foundation.layout.Spacer
import androidx.compose.foundation.layout.fillMaxHeight
import androidx.compose.foundation.layout.fillMaxSize
import androidx.compose.foundation.layout.fillMaxWidth
import androidx.compose.foundation.layout.height
import androidx.compose.foundation.layout.padding
import androidx.compose.foundation.layout.size
import androidx.compose.foundation.layout.width
import androidx.compose.foundation.shape.RoundedCornerShape
import androidx.compose.material3.AlertDialogDefaults.shape
import androidx.compose.material3.Button
import androidx.compose.material3.ButtonColors
import androidx.compose.material3.ButtonDefaults
import androidx.compose.material3.Checkbox
import androidx.compose.material3.Divider
import androidx.compose.material3.ExperimentalMaterial3Api
import androidx.compose.material3.Icon
import androidx.compose.material3.MaterialTheme
import androidx.compose.material3.Surface
import androidx.compose.material3.Text
import androidx.compose.material3.TextField
import androidx.compose.material3.TextFieldDefaults
import androidx.compose.runtime.Composable
import androidx.compose.runtime.getValue
import androidx.compose.runtime.mutableStateOf
import androidx.compose.runtime.remember
import androidx.compose.runtime.setValue
import androidx.compose.ui.Alignment
import androidx.compose.ui.Modifier
import androidx.compose.ui.graphics.Color
import androidx.compose.ui.platform.LocalContext
import androidx.compose.ui.res.painterResource
import androidx.compose.ui.text.font.FontFamily
import androidx.compose.ui.text.font.FontWeight
```

```kotlin
import androidx.compose.ui.tooling.preview.Preview
import androidx.compose.ui.unit.dp
import androidx.compose.ui.unit.sp
import com.shrimadbhagwat.googleaccount.ui.theme.GoogleAccountTheme


class MainActivity : ComponentActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContent {
            GoogleAccountTheme() {
                // A surface container using the 'background' color from the theme
                Surface(
                    modifier = Modifier.fillMaxSize(),
                    color = MaterialTheme.colorScheme.background
                ) {
                    App()
                }
            }
        }
    }
}

@OptIn(ExperimentalMaterial3Api::class)
@Composable
fun App(modifier: Modifier = Modifier) {
    var applicationContext = LocalContext.current
    val isChecked = remember { mutableStateOf(false) }
    var firstname by remember {
        mutableStateOf("")
    }
    var lastname by remember {
        mutableStateOf("")
    }
    var username by remember {
        mutableStateOf("")
    }
    var password by remember {
        mutableStateOf("")
    }
    var confirmpassword by remember {
        mutableStateOf("")
    }
    Column(
        verticalArrangement = Arrangement.Center,
        horizontalAlignment = Alignment.Start,
        modifier = Modifier
            .fillMaxSize()
            .background(Color.White)
            .padding(16.dp)
    ) {

        Spacer(modifier = Modifier.height(16.dp))
        // Logo
        Row() {
            Image(
                painter = painterResource(id = R.drawable.googlefull),
```

```kotlin
            contentDescription = "Instagram",
            modifier = Modifier.width(120.dp)
        )
    }
    Spacer(modifier = Modifier.height(16.dp))

    // TextLine
    Text(
        text = "Create your Google Account",
        fontSize = 24.sp,
        fontWeight = FontWeight.Bold,
        color = Color.Black,
        fontFamily = FontFamily.SansSerif

    )

    Spacer(modifier = Modifier.height(16.dp))

    Row {
        TextField(
            value = firstname,
            onValueChange = { firstname = it },
            colors = TextFieldDefaults.textFieldColors(containerColor = Color.White),

            placeholder = { Text("First name", color = Color.Gray) },
            modifier = Modifier
                .width(172.dp)
                .height(50.dp)
                .border(1.dp, Color.LightGray)

        )
        Spacer(modifier = Modifier.width(12.dp))
        TextField(
            value = lastname,
            onValueChange = { lastname = it },
            colors = TextFieldDefaults.textFieldColors(containerColor = Color.White),

            placeholder = { Text("Last name", color = Color.Gray) },
            modifier = Modifier
                .width(172.dp)
                .height(50.dp)
                .border(1.dp, Color.LightGray)

        )
    }

    Spacer(modifier = Modifier.height(32.dp))

    Row {
        TextField(
            value = username,
            onValueChange = { username = it },
            colors = TextFieldDefaults.textFieldColors(containerColor = Color.White),

            placeholder = { Text("Username", color = Color.Gray) },
            modifier = Modifier
                .width(360.dp)
```

```kotlin
                    .height(50.dp)
                    .border(1.dp, Color.LightGray)

            )
        }
        Spacer(modifier = Modifier.height(12.dp))
        Row() {
            Text(
                text = "  You can use letters, number & periods",
                color = Color.Gray,
                fontSize = 12.sp
            )
        }

        Spacer(modifier = Modifier.height(16.dp))
        Row() {
            Text(
                text = "Use my current email address instead",
                color = Color(0xFF1A70D5),
                fontSize = 16.sp,
                fontWeight = FontWeight.Bold
            )
        }
        Spacer(modifier = Modifier.height(48.dp))

        Row {
            TextField(
                value = password,
                onValueChange = { password = it },
                colors = TextFieldDefaults.textFieldColors(containerColor = Color.White),

                placeholder = { Text("Password", color = Color.Gray) },
                modifier = Modifier
                    .width(172.dp)
                    .height(50.dp)
                    .border(1.dp, Color.LightGray)

            )
            Spacer(modifier = Modifier.width(12.dp))
            TextField(
                value = confirmpassword,
                onValueChange = { confirmpassword = it },
                colors = TextFieldDefaults.textFieldColors(containerColor = Color.White),

                placeholder = { Text("Confirm", color = Color.Gray) },
                modifier = Modifier
                    .width(172.dp)
                    .height(50.dp)
                    .border(1.dp, Color.LightGray)

            )
        }
        Spacer(modifier = Modifier.height(12.dp))
        Row() {
            Text(
                text = "  Use 8 or more characters with a mix of letters, numbers &",
                color = Color.Gray,
```

```kotlin
            fontSize = 12.sp,

        )
}

Row() {
    Text(
        text = "  symbols",
        color = Color.Gray,
        fontSize = 12.sp,

    )
}

Spacer(modifier = Modifier.height(24.dp))
Row() {

    Column() {

        Checkbox(checked = isChecked.value, onCheckedChange = { isChecked.value = it })
    }
    Column {
        Spacer(modifier = Modifier.height(14.dp))
        Text(
            text = "  Show Password",
            color = Color.Black,
            fontSize = 16.sp,
            fontWeight = FontWeight.Bold
        )
    }

}

Spacer(modifier = Modifier.height(60.dp))
Row() {

    Button(
        onClick = { /*TODO*/ }, colors = ButtonDefaults.buttonColors(Color.White),
        shape = RoundedCornerShape(10.dp),
        modifier = Modifier
            .padding(all = 5.dp)
            .width(160.dp)
            .height(46.dp)
    ) {
        Text(
            text = "Sign in instead",
            color = Color(0xFF1A70D5),
            fontSize = 16.sp,
            fontWeight = FontWeight.Bold
        )
    }

    Spacer(modifier = Modifier.width(80.dp))
```

```
        Button(
            onClick = {
                    Toast.makeText(applicationContext,firstname + "\n" + lastname + "\n"  + username + "\n" +
password + "\n" + confirmpassword,Toast.LENGTH_LONG).show()
            }, colors = ButtonDefaults.buttonColors(Color(0xFF1A70D5)),
            shape = RoundedCornerShape(10.dp),
            modifier = Modifier
                .padding(all = 5.dp)
                .width(100.dp)
                .height(46.dp)
        ) {

            Text(
                text = "Next",
                fontWeight = FontWeight.Bold,
                color = Color.White,
                fontSize = 18.sp
            )
        }
    }

  }
}
```

OUTPUT

Google

Create your Google Account

First name	Last name

Username

You can use letters, number & periods

Use my current email address instead

Password	Confirm

Use 8 or more characters with a mix of letters, numbers & symbols

☐ **Show Password**

Sign in instead   **Next**

Google

Create your Google Account

Shrimad	Bhagwat

shrimad01

You can use letters, number & periods

Use my current email address instead

abc@123	abc@123

Use 8 or more characters with a mix of letters, numbers & symbols

☑ **Show Password**

Sign in instead

Shrimad
Bhagwat
shrimad01
abc@123
abc@123

Next