Android Application Development in Kotlin

Code: (MainAcitivity.kt)

```
 package com.example.vit_20bcd7121

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.layout.Arrangement
import androidx.compose.foundation.layout.Column
import androidx.compose.foundation.layout.Row
import androidx.compose.foundation.layout.fillMaxSize
import androidx.compose.foundation.layout.padding
import androidx.compose.foundation.layout.size
import androidx.compose.material3.Button
import androidx.compose.material3.ButtonDefaults
import androidx.compose.material3.Checkbox
import androidx.compose.material3.ExperimentalMaterial3Api
import androidx.compose.material3.OutlinedTextField
import androidx.compose.material3.Text
import androidx.compose.material3.TextField
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.res.painterResource
import androidx.compose.ui.unit.dp
import androidx.compose.ui.unit.sp


class MainActivity : ComponentActivity() {
    @OptIn(ExperimentalMaterial3Api::class)
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContent {
            Column(
                verticalArrangement = Arrangement.Top,
                horizontalAlignment = Alignment.Start,
                modifier = Modifier
                    .fillMaxSize()
                    .background(color = Color.White)
                    .padding(15.dp)
            ) {
                Image(painterResource(id = R.drawable.google), contentDescription
="",Modifier.size(200.dp) )
                var a by remember {
                    mutableStateOf("")
                }

                var b by remember {
                    mutableStateOf("")
                }

                var c by remember {
                    mutableStateOf("")
                }

                var x by remember {
                    mutableStateOf("")
                }

                var y by remember {
                    mutableStateOf("")
                }

                Row(){
                    OutlinedTextField(value = a, onValueChange = { a = it
},Modifier.size(170.dp,60.dp)
                    ,label={Text(text="First Name")})
                    Text(text="    ")
                    OutlinedTextField(value = b, onValueChange = { b = it
},Modifier.size(170.dp,60.dp)
                    ,label={Text(text="Last Name")})
                }

                Text(text="    ")

                OutlinedTextField(value = c, onValueChange = { a = it
},label={Text(text="Username")})

                Text(text="You can use letters,numbers & periods", fontSize = 10.sp,color=
Color.Gray)

                Text(text="    ")

                Text(text="Use my current email address instead",color= Color.Blue)

                Text(text="    ")

                Row(horizontalArrangement = Arrangement.Center) {
                    OutlinedTextField(value = x, onValueChange = { x = it
},Modifier.size(170.dp,57.dp),label={Text(text="Password")})
                    Text(text="    ")
                    OutlinedTextField(value = y, onValueChange = { y = it
},Modifier.size(170.dp,57.dp),label={Text(text="Confirm Password")})
                }

                Text(text="Use 8 or more characters with a mix of letters numbers and
symbols",fontSize = 10.sp,color= Color.Gray)
                Text(text=" ")
                var k=false
                Row(horizontalArrangement = Arrangement.Center){
                    Checkbox(checked = k, onCheckedChange={k=it})
                    Text(text="Show Password")
                }
                Text(text="    ")

                Row(horizontalArrangement = Arrangement.Center){
                    Text(text="Sign in instead",color=Color.Blue)
                    Text(text="                                    ")
                    Button(onClick = { Toast.makeText(applicationContext,"Signing
in",Toast.LENGTH_SHORT).show()},colors=ButtonDefaults.buttonColors(Color.Blue)) {
                    Text(text="Next")
                    }
                  }
                }
              }
            }
          }
```