

VIT VELLORE



NAME : TANISH BHATT

REG. NO. : 21BEC2122

COURSE : ANDROID APP DEVELOPMENT

TASK : ASSIGNMENT 1

```

package com.example.sample

import android.os.Bundle
import androidx.activity.ComponentActivity
import androidx.activity.compose.setContent
import androidx.compose.foundation.layout.Arrangement
import androidx.compose.foundation.layout.Column
import androidx.compose.foundation.layout.Spacer
import androidx.compose.foundation.layout.fillMaxSize
import androidx.compose.foundation.layout.padding
import androidx.compose.material3.Button
import androidx.compose.material3.ButtonDefaults
import androidx.compose.material3.ExperimentalMaterial3Api
import androidx.compose.material3.MaterialTheme
import androidx.compose.material3.Surface
import androidx.compose.material3.Text
import androidx.compose.material3.TextField
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.graphics.RectangleShape
import androidx.compose.ui.text.TextStyle
import androidx.compose.ui.tooling.preview.Preview
import androidx.compose.ui.unit.dp
import com.example.sample.ui.theme.SampleTheme

class MainActivity : ComponentActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContent {
            Column(modifier = Modifier.fillMaxSize(),
horizontalAlignment = Alignment.CenterHorizontally) {
                simpleText()

                simpleTextField()

                simpleTextFieldpswrd()
                Column(modifier =
Modifier.fillMaxSize(1f) ,verticalArrangement = Arrangement.Center,
horizontalAlignment = Alignment.CenterHorizontally) {
                    simpleButton()
                }
            }
        }
    }
}

@Composable
fun simpleText() {

```

```

        Text(text = "USER LOGIN", modifier = Modifier.padding(50.dp),
            style = TextStyle.Default
        )
    }

@OptIn(ExperimentalMaterial3Api::class)
@Composable
fun simpleTextField(){

    var text by remember { mutableStateOf("") }
    TextField(value = text , onValueChange = {it -> text = it },
        textStyle = TextStyle(color = Color.Black), label = { Text(text =
"USERNAME")} )
}

@OptIn(ExperimentalMaterial3Api::class)
@Composable
fun simpleTextFielddpswr(){

    var text by remember { mutableStateOf("") }
    TextField(value = text , onValueChange = {it -> text = it },
        textStyle = TextStyle(color = Color.Black), label = { Text(text =
"PASSWORD")} )
}

@Composable
fun simpleButton(){

    Button(modifier = Modifier.padding(16.dp), onClick = { /
*TODO*/ }, shape = RectangleShape) {
        Text(text = "LOGIN")
    }

}

@Preview
@Composable
fun simpleTextPreview(){
    Column(modifier = Modifier.fillMaxSize(), horizontalAlignment =
Alignment.CenterHorizontally) {
        simpleText()

        simpleTextField()

        simpleTextFielddpswr()
        Column(modifier =
Modifier.fillMaxSize(1f) ,verticalArrangement = Arrangement.Center,
horizontalAlignment = Alignment.CenterHorizontally) {
            simpleButton()
        }
    }
}

```

}
}
}

Output :

