Vaishav Vijaykumar Dhepe
202206780
Mobile Application Development
Mid Term 1

Code :

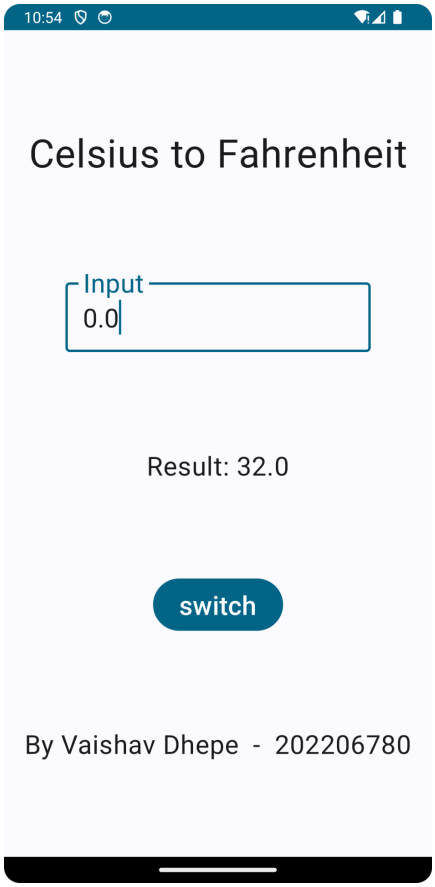
```
package com.example.quizprogram1

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.fillMaxSize
import androidx.compose.foundation.text.KeyboardOptions
import androidx.compose.material3.Button
import androidx.compose.material3.ExperimentalMaterial3Api
import androidx.compose.material3.OutlinedTextField
import androidx.compose.material3.Surface
import androidx.compose.material3.Text
import androidx.compose.runtime.Composable
import androidx.compose.runtime.getValue
import androidx.compose.runtime.mutableStateOf
import androidx.compose.runtime.remember
import androidx.compose.runtime.saveable.rememberSaveable
import androidx.compose.runtime.setValue
import androidx.compose.ui.Alignment
import androidx.compose.ui.Modifier
import androidx.compose.ui.text.TextStyle
import androidx.compose.ui.text.input.ImeAction
import androidx.compose.ui.text.input.KeyboardType
import androidx.compose.ui.unit.sp
import com.example.quizprogram1.ui.theme.QuizProgram1Theme
import java.util.Locale

class MainActivity : ComponentActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContent {
            QuizProgram1Theme {
```

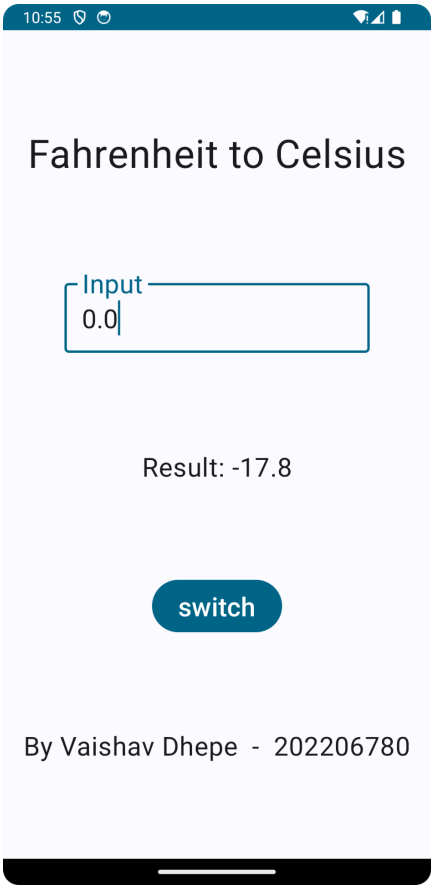
```
            // A surface container using the 'background' color from the theme
            Surface(
                modifier = Modifier.fillMaxSize()
            ) {
                //Task 2: Define state variables
                var isCelsius by rememberSaveable { mutableStateOf(true) }
                var currentInput by rememberSaveable { mutableStateOf("0.0") }

                //Task 3: Define lambda function for onValueChange of the TextField
                val onInputChange: (String) -> Unit = { currentInput = it }


                //Task 4: Define a lambda function for onClick callback for the "switch" Button
                val onSwitchClick = { isCelsius = !isCelsius }

                //Task 5: replace the call of TempConverter function with the define variables and
functions
                TempConverter(currentInput, isCelsius, onInputChange, onSwitchClick)
            }
        }
    }
}

@OptIn(ExperimentalMaterial3Api::class)
@Composable
fun TempConverter(
    input: String,
    c2f: Boolean,
    onInputChange: (String) -> Unit,
    onSwitchButton: () -> Unit
) {
    //Task 6: convert the "input" to double  and store it in a variable called "temperature" (could be
null!)
    val temperature: Double? = input.toDoubleOrNull()

    //Task 7: calculate the result, depending on the c2f state
    //note that variable "temperature" could be null
    val result = temperature?.let { temp ->
        if (c2f) {
            (temp * 9.0 / 5.0 + 32).toString()
        } else {
```

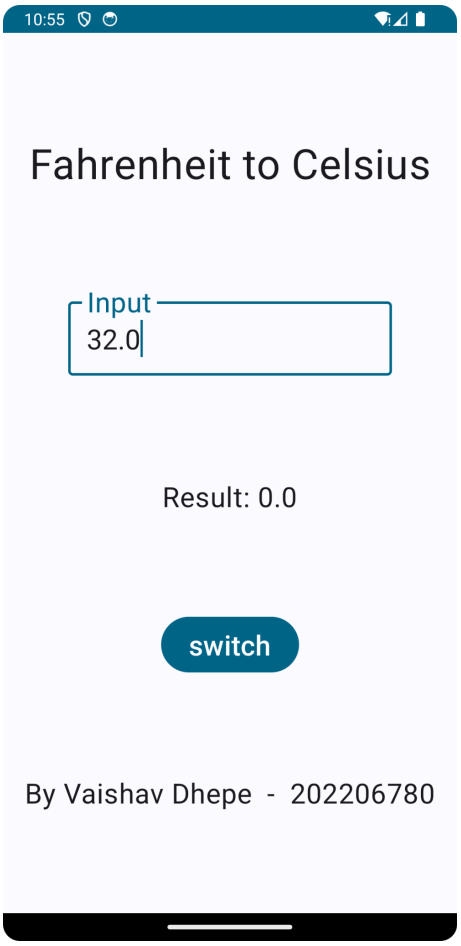
```
            ((temp - 32) * 5.0 / 9.0).toString()
        }
    } ?: ""

    val finalResult = if (result != "") {
        String.format(
            Locale.US, "%.1f", result.toDouble()
        )
    } else {
        ""
    }

    Column(
        horizontalAlignment = Alignment.CenterHorizontally,
        verticalArrangement = Arrangement.SpaceEvenly
    ) {
        //Task 8: change the following Text to show  "Celsius to Fahrenheit" or "Fahrenheit to
Celsius" depending on the c2f state
        if (c2f) {
            Text("Celsius to Fahrenheit", fontSize = 36.sp)
        } else {
            Text("Fahrenheit to Celsius", fontSize = 36.sp)
        }



        OutlinedTextField(
            value = input,
            label = { Text(text = "Input", fontSize = 24.sp) },
            textStyle = TextStyle.Default.copy(fontSize = 24.sp),
            keyboardOptions = KeyboardOptions.Default.copy(
                keyboardType = KeyboardType.Number,
                imeAction = ImeAction.Done
            ),
            onValueChange = onInputChange
        )

        Text(
            text = "Result: $finalResult", fontSize = 24.sp,
        )
        //switch Button
        Button(
            onClick = onSwitchButton,
```

```
            enabled = true
        ) {
            Text("switch", fontSize = 24.sp)
        }
        //Task 1: put your name and student ID in the following Text
        Text(text = "By Vaishav Dhepe  -  202206780", fontSize = 24.sp)
    }

}

//Task 9: make the "result" show only 1 digit after the decimal point

//Task 10: the input may get lost when you rotate your screen
//briefly describe a method to help solve this problem.

// To preserve the state when the user rotates the mobile we can use rememberSaveable() or ViewModel
// as we have done above. This wil maintain the state even if we rotate the mobile
```

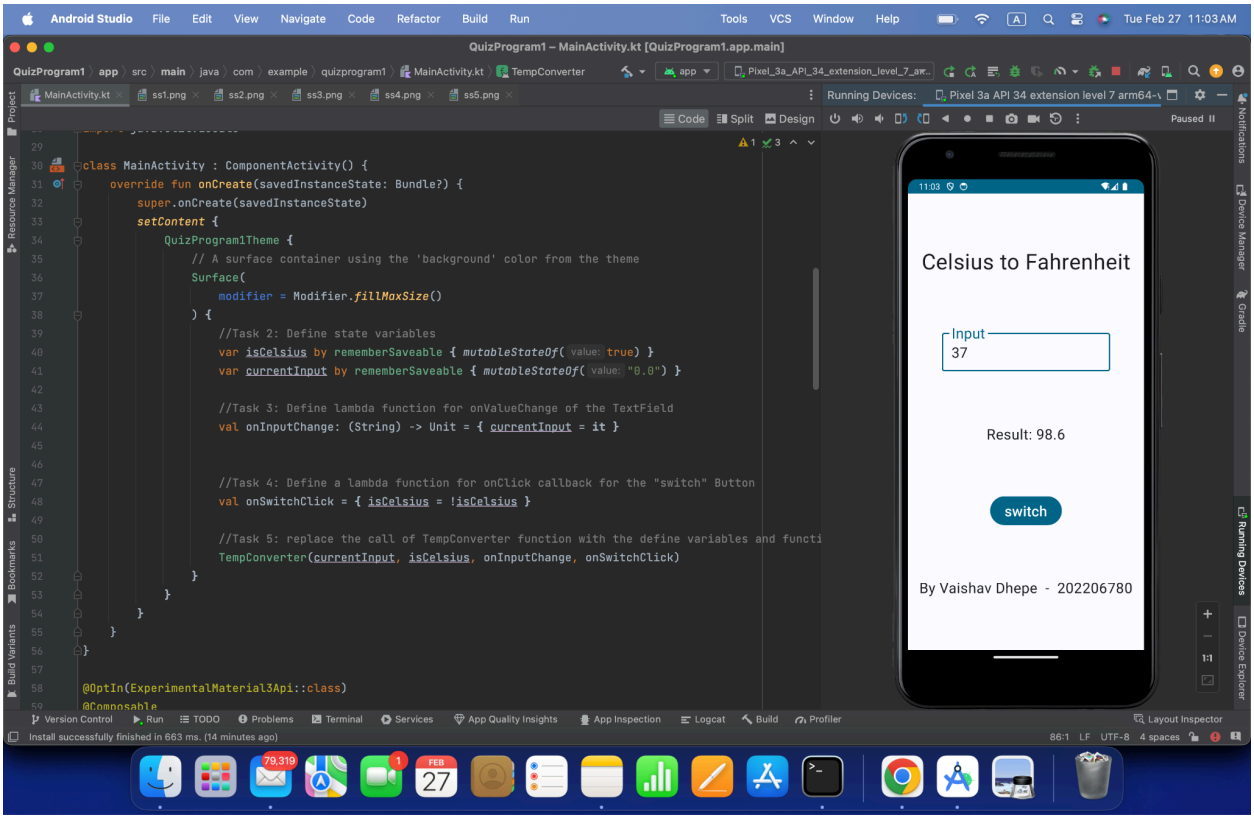
Screenshots:






10:55
Fahrenheit to Celsius
Input
32.0
Result: 0.0
switch
By Vaishav Dhepe - 202206780


10:56
Celsius to Fahrenheit
Input
Result:
switch
By Vaishav Dhepe - 202206780


Android Studio File Edit View Navigate Code Refactor Build Run Tools VCS Window Help Tue Feb 27 11:03 AM
QuizProgram1 – MainActivity.kt [QuizProgram1.app.main]
class MainActivity : ComponentActivity() {
override fun onCreate(savedInstanceState: Bundle?) {
super.onCreate(savedInstanceState)
setContent {
QuizProgram1Theme {
// A surface container using the 'background' color from the theme
Surface(
modifier = Modifier.fillMaxSize()
) {
//Task 2: Define state variables
var isCelsius by rememberSaveable { mutableStateOf(value: true) }
var currentInput by rememberSaveable { mutableStateOf(value: "0.0") }
//Task 3: Define lambda function for onValueChange of the TextField
val onInputChange: (String) -> Unit = { currentInput = it }
//Task 4: Define a lambda function for onClick callback for the "switch" Button
val onSwitchClick = { isCelsius = !isCelsius }
//Task 5: replace the call of TempConverter function with the define variables and functi
TempConverter(currentInput, isCelsius, onInputChange, onSwitchClick)
@OptIn(ExperimentalMaterial3Api::class)
@Composable
Celsius to Fahrenheit
Input
37
Result: 98.6
switch
By Vaishav Dhepe - 202206780